Bienvenida del Comité Organizador

En el nombre del Comité Organizador, es un placer darles la bienvenida a la quinta edición de las Jornadas en Tecnología del Habla promovidas por la Red Temática en Tecnologías del Habla.

Esta quinta edición ha sido organizada por el laboratorio *Aholab-Signal Processing Laboratory* del departamento de Electrónica y Telecomunicaciones de la Universidad del País Vasco / Euskal Herriko Unibertsitatea, y con el apoyo de la ISCA (International Speech Communication Association). Las Jornadas han tenido el apoyo económico e institucional del Ministerio de Ciencia e Innovación, el Gobierno Vasco, la UPV/EHU, la BBK, así como patrocinios privados de las empresas Euskaltel S.A. y la empresa Bai & By.

Esta serie de Jornadas se han convertido en un punto de encuentro de los científicos y técnicos dedicados a la investigación y aplicación de las Tecnologías del Habla. Las Jornadas brindan una oportunidad excepcional para la presentación y divulgación de trabajos procedentes de diferentes campos de actividad relacionados, así como para el acercamiento del colectivo de investigadores, cada vez más numeroso, dedicados a esta disciplina. Al mismo tiempo, las Jornadas sirven de estímulo para la presentación de trabajos realizados por jóvenes investigadores que comienzan su actividad en el campo de las Tecnologías del Habla.

Al igual que en la edición anterior, y con el fin de fomentar y estimular la participación de jóvenes investigadores, la Red Temática en Tecnologías del Habla ha patrocinado 6 premios a los mejores artículos cuyo primer firmante sea un estudiante de una Universidad o Centro de Investigación Español. Además, se ha continuado con el desarrollo de campañas de evaluación de sistemas propuesto en la edición anterior, planteándose en esta edición las campañas en tres temáticas de alto nivel de interés. Las campañas han sido organizadas por 3 laboratorios de investigación de la UPV/EHU: la evaluación de sistemas de Conversión de Texto a Voz por el laboratorio organizador de las jornadas; la evaluación de sistemas de Verificación de la Lengua organizada por el laboratorio GTTH y la evaluación de sistemas de Traducción Automática organizadas por el grupo IXA. Las tres campañas han tenido un alto nivel de participación y se otorgarán sendos premios a los sistemas ganadores.

Además del programa de sesiones orales y pósters, se ha organizado una sesión especial de Proyectos y Demos, en la que se presentarán demostraciones y proyectos en marcha. Se han organizado también cuatro conferencias plenarias invitadas, que esperamos sean de gran interés para todos los participantes. El miércoles 12 de Noviembre Yannis Stylianou de la Universidad de Creta nos hablará sobre "Voice Conversion: State of the Art and Perspectives". El jueves 13 de Noviembre Björn Granstrom del Real Instituto de Tecnología de Estocolmo (KTH) nos dará una charla sobre "Embodied Conversational Agents in Verbal and Non-Verbal Communication" y Néstor Becerra Yoma de la Universidad de Chile sobre "Aplicaciones de las Tecnologías del Habla en Sistemas CALL y CAPT". Finalmente, el último día de las jornadas Giuseppe Riccardi de la Universidad de Trento hablará sobre "Third-Generation Conversational Interfaces"

Quisiera expresar mi agradecimiento en primer lugar a todos y cada uno de los miembros del Comité Organizador por su entusiasta colaboración en la organización de este evento. Agradezco igualmente su cooperación al Comité Científico y a las diferentes entidades que con su apoyo y colaboración han contribuido a una mejor organización de las Jornadas.

Finalmente, quiero desear a todos los participantes una estancia fructífera y agradable en Bilbao, disfrutando de las actividades lúdicas preparadas como complemento al programa técnico.

Bilbao, Noviembre de 2008
Inmaculada Hernáez Rioja
Presidenta del Comité Organizador JTH2008

ÍNDICE

SESIÓN ORAL 1: MODELADO ACÚSTICO

Generalized gaussians for continuous observation distributions in speech recognition
Antonio Miguel, Eduardo Lleida, Alfonso Ortega 3

Graphical models for discrete observation distributions in speech recognition
Antonio Miguel, Eduardo Lleida, Alfonso Ortega 7

On the use of augmented HMM models for overcoming time and parameter independence assumptions in ASR
Marta Casar, José A. R. Fonollosa 11

Support vector regression in NIST SRE 2008 multichannel core task
Ismael Mateos, Daniel Ramos, Ignacio López-Moreno, Joaquín González-Rodríguez 15

Training a robust command recognizer with the tecnovoz database
José Lopes, Cláudio Neves, Arlindo Veiga, Alexandre Maciel, Carla Lopes, Luís Sá, Fernando Perdigão 19

SESIÓN DE POSTER 1

Acoustic Event Recognition for Low Cost Language Identification
Danilo Spada, Ignacio López, Doroteo T. Toledano, Joaquín González-Rodríguez 25

Applying feature reduction analysis to a PPRLM-multiple gaussian language identification system
Juan Manuel Lucas Cuesta, Ricardo de Córdoba Herralde, Luis Fernando D'Haro Enríquez 29

Bio-inspired dynamic formant tracking for phonetic labelling
P. Gómez, J. M. Ferrández, V. Rodellar, R. Martínez, C. Muñoz, A. Álvarez, L. M. Mazaira 33

***COMUNICA* - Plataforma para el desarrollo, distribución y evaluación de herramientas logopédicas asistidas por ordenador**
Óscar Saz, W.-Ricardo Rodríguez, Eduardo Lleida, Carlos Vaquero, Antonio Escartín 37

Detección de cambios de toma con información de contenido visual y auditivo
Alejandro Abejón, Ismael Mateos 41

Feature selection *vs.* feature transformation in reducing dimensionality for speaker recognition
Maider Zamalloa, L. J. Rodríguez-Fuentes, Mikel Peñagarikano, Germán Bordel, Juan P. Uribe 45

n-gram Frequency Ranking with additional sources of information in a multiple-Gaussian classifier for Language Identification
Ricardo Córdoba, Luis F. D'Haro, Juan M. Lucas, Javier Zugasti 49

PRE-LINGUA - Una herramienta de apoyo para el pre-lenguaje
William Ricardo Rodríguez Dueñas, Eduardo Lleida Solano 53

Procedimiento para la medida y la modificación del jitter y del shimmer aplicado a la síntesis del habla expresiva
Carlos Monzo, Ignasi Iriondo, Elisa Martínez 58

T-NORM y desajuste léxico y acústico en reconocimiento de locutor dependiente de texto
Daniel Hernández López, Doroteo Torre Toledano, Cristina Esteve Elizalde, Joaquín González Rodríguez, Rubén Fernández Pozo, Luis Hernández Gómez 62

Turning Wikipedia into a resource for language research
Alberto Montero-Asenjo, Carlos A. Iglesias 66

Using pitch and formants for order adaptation of fractional fourier transform in speech signal processing
Hui Yin, Climent Nadeu, Volker Hohmann 71

iATROS: A speech and handwriting recognition system
Míriam Luján-Mares, Vicent Tamarit, Vicent Alabau, Carlos-D. Martínez-Hinarejos, Moisés Pastor, Alberto Sanchís, Alejandro Toselli 75

Valoración del cierre neo-glótico en voz esofágica
Roberto Fernández-Baíllo, Pedro Gómez, Bartolomé Scola, Carlos Ramírez 79

SESIÓN ESPECIAL 1 SOBRE LA EVALUACIÓN ALBAYZIN 08

Adaptación del CTH-URL para la competición ALBAYZIN 2008
Carlos Monzo, Lluís Formiga, Jordi Adell, Ignasi Iriondo, Francesc Alías, Joan Claudi Socoró 87

ATVS-UAM ALBAYZIN-VL08 system description
Doroteo T. Toledano, Ismael Mateos-García, Alejandro Abejón-González, Daniel Ramos, Juan Bonillo, Joaquín González-Rodríguez 91

Descripción de los sistemas presentados por IXA-EHU a la evaluación ALBAYCIN'08
Gorka Labaka, Arantza Díaz de Ilarraza, Kepa Sarasola 93

Descripción del conversor de texto a voz AHOTTS presentado a la evaluación ALBAYZIN TTS 2008
Iñaki Sainz, Inma Hernáez, Eva Navas, Jon Sánchez, Iker Luengo, Ibon Saratxaga, Igor Odriozola, Eneritz de Bilbao, Daniel Erro 96

Descripción del sintetizador de voz COTOVÍA para la evaluación ALBAYZIN TTS 2008
Eduardo R. Banga, Francisco Méndez, Francisco Campillo, Gonzalo Iglesias, Laura Docío 100

Descripción del Sistema I de Telefónica I+D presentado a la evaluación ALBAYZIN'08 para CTV
M. Á. Rodríguez, J. G. Escalada, A. Armenta 104

Descripción del Sistema II de Telefónica I+D presentado a la evaluación ALBAYZIN'08 para CTV
J. G. Escalada, A. Armenta, M. Á. Rodríguez 107

El sistema de identificación de la lengua de PRHLT
Míriam Luján-Mares, Vicent Tamarit, Roberto Paredes, Vicent Alabau, Carlos-D. Martínez-Hinarejos 110

Evaluación ALBAYZÍN-08 de sistemas de verificación de la lengua: sistema del grupo Softlab de la UC3M
M. J. Poza, B. Ruiz, L. Puente, D. Carrero 112

Generación de una voz sintética en castellano basada en HSMM para la evaluación ALBAYZÍN 2008: Conversión texto a voz
R. Barra-Chicote, J. Yamagishi, J. M. Montero, S. King, S. Lufti, J. Macías-Guarasa 115

Phrase segments obtained with stochastic inversion transduction grammars for spanish-basque translation
Germán Sanchís-Trilles, Joan Andreu Sánchez 119

The AVIVAVOZ phrase-based statistical machine translation system for ALBAYZIN 2008
Carlos A. Henríquez Q., Maxim Khalilov, José B. Mariño, Nerea Ezeiza 123

The CEREVOICE speech synthesiser
Juan María Garrido, Eva Bofias, Yesika Laplaza, Montserrat Marquina, Matthew Aylett, Chris Pidcock 126

The UPC TTS system description
Antonio Bonafonte, Pablo Daniel Agüero 130

The L^2F language verification systems for ALBAYZIN-08 evaluation
Alberto Abad, Isabel Trancoso 134

SESIÓN ORAL 2: SÍNTESIS DEL HABLA

Bayes Optimal Classification for Corpus-Based Unit Selection in TTS Synthesis
Hamurabi Gamboa Rosales, Oliver Jokisch, Ruediger Hoffmann 141

Flexible harmonic/stochasticmodeling for HMM-based speech synthesis
Eleftherios Banos, Daniel Erro, Antonio Bonafonte, Asunción Moreno 145

Further improvements to pronunciation by analogy
Tatyana Polyákova, Antonio Bonafonte 149

Modelo de síntesis de habla con disfluencias basado en modificaciones locales sobre frases constituyentes
Jordi Adell, David Escudero-Mancebo, Antonio Bonafonte 153

Nuevo módulo de análisis prosódico del conversor texto-voz multilingüe de Telefónica I+D
M. Á. Rodríguez, J. G. Escalada, A. Armenta, J. M. Garrido 157

SESIÓN DE POSTER 2

Adquisición de un corpus de diálogos para un dominio de reservas de instalaciones deportivas
E. Segarra, M. J. Castro, I. Galiano, F. García, J. A. Gómez, D. Griol, L. F. Hurtado, E. Sanchís, F. Torres, F. Zamora 163

Adquisición y evaluación de un corpus de diálogos mediante una técnica de generación automática de diálogos
David Griol, Lluís F. Hurtado, Emilio Sanchís, Encarna Segarra 167

Arquitectura distribuida para el desarrollo de sistemas de diálogo hablado, edecán
José Enrique García, Alfonso Ortega, Antonio Miguel, Eduardo Lleida 171

Arquitectura multimodal controlada por voz: revisión de metáforas de interacción
David Escudero-Mancebo, Héctor Olmedo-Rodríguez, Valentín Cardeñoso-Payo 175

Automatic word stress marker for portuguese TTS
Daniela Braga, Luis Coelho 179

Dialog ACT labeling in the DIHANA corpus using prosody information
Vicent Tamarit, Carlos-D. Martínez-Hinarejos 183

Evaluación de campo de un sistema de diálogo oral empleando relaciones estadísticas
Zoraida Callejas, Ramón López-Cózar 187

Evaluación subjetiva de una base de datos de habla emocional para euskera
Iñaki Sainz, Ibon Saratxaga, Eva Navas, Inmaculada Hernáez, Jon Sánchez, Iker Luengo, Igor Odriozola, Eneritz de Bilbao 191

Grabación de una base de datos bilingüe euskera/castellano para verificación de locutor
Iker Luengo, Eva Navas, Iñaki Sainz, Ibon Saratxaga, Jon Sánchez, Igor Odriozola, Juan José Igarza, Inmaculada Hernáez 195

Intelligibility of accented speech: the perception of word-final nasals by dutch and brazilians
Denise C. Kluge, Mara S. Reis, Denize Nobre-Oliveira, Andréia S. Rauber 199

Nueva técnica de post-corrección de errores de RAH para sistemas de diálogo oral
Ramón López-Cózar, Zoraida Callejas 203

Transcripción fonética en un entorno plurilingüe
Tatyana Polyákova, Antonio Bonafonte 207

Voice pleasantness: on the improvement of TTS voice quality
Luis Coelho, Daniela Braga, Carmen García-Mateo 211

SESIÓN PROYECTOS/DEMOS

A system architecture for multilingual spoken document retrieval
German Bordel, Arantza Casillas, Mikel Penagarikano, Luis Javier Rodríguez, Amparo Varona 217

AnHitz, development and integration of language, speech and visual technologies for Basque
Kutz Arrieta, Arantza Díaz de Ilarraza, Inma Hernáez, Urtza Iturraspe, Igor Leturia, Eva Navas, Kepa Sarasola 221

Arquitectura distribuida para un sistema de traducción del habla sobre la plataforma UIMA
Marc Poch, David Cuestas, José B. Mariño, Francisco Méndez, Iñaki Sainz 225

Computer-assisted handwritten text transcription using speech recognition
Antonio-L. Lagarda, Vicent Alabau, Carlos-D. Martínez-Hinarejos, Alejandro-H. Toselli, Verónica Romero, José-R. Navarro, Enrique Vidal 229

SAUTRELA: Un entorno de desarrollo versátil para las tecnologías del habla
Mikel Peñagarikano, Germán Bordel, Sonia Bilbao, Maider Zamalloa, Luis Javier Rodríguez 233

Servicio de Páginas Amarillas utilizando reconocimiento distribuido de voz
José A. González, Ángel M. Gómez, José L. Carmona, Antonio M. Peinado 236

Sistema de reconocimiento automático del habla distribuido aplicado a entornos logísticos
José Enrique García, Alfonso Ortega, Antonio Miguel, Eduardo Lleida 240

SESIÓN ORAL 3: RECONOCIMIENTO DEL HABLA

A novel two-level architecture plus confidence measures for a keyword spotting system
Javier Tejedor, Simon King, Joe Frankel, Dong Wang, José Colás, Javier Garrido 247

Cuantificación vectorial diferencial para la transmisión eficiente de parámetros acústicos en sistemas de reconocimiento automático del habla distribuido
José Enrique García, Alfonso Ortega, Antonio Miguel, Eduardo Lleida 251

Improved unsupervised speech recognition system using MLLR speaker adaptation and confidence measurement
Mukund Jha, Sourabh Sriom, Míriam Luján, Carlos D. Martínez-Hinarejo, Alberto Sanchís 255

New features for improving VAD when dealing with far-field and multi-speaker speech
Óscar Varela Serrano, Rubén San-Segundo Hernández, Luis Alfonso Hernández Gómez 259

SVM based posterior probabilities for syllable confidence annotation
Daniel Bolanos, Wayne Ward, Javier Tejedor 263

SESIÓN ORAL 4: TRADUCCIÓN AUTOMÁTICA

Deriving benefit from a generalized syntax-based reordering
Maxim Khalilov, José A. R. Fonollosa, Mark Dras 269

Incorporación de información sintáctico-semántica en la traducción de voz a lengua de signos
B. Gallo, R. San-Segundo, J. M. Lucas, R. Barra, F. Fernández, L. F. D'Haro 273

N-II: Traductor automático estadístico basado en ngramas
Marta R. Costa-Jussà, Mireia Farrús, Marc Poch, Adolfo Hernández, José B. Mariño 277

Statistical methods for speech technologies in basque language
M. Inés Torres, Víctor Guijarrubia, Raquel Justo, Alicia Pérez, Francisco Casacuberta 281

Técnicas estadísticas para el filtrado de un corpus bilingüe en traducción automática
Enrique Montolar, Marta R. Costa-Jussà, José A. R. Fonollosa 285

CONFERENCIAS INVITADAS

Voice conversion: State of the art and perspectives
Yannis Styilianou 291

Embodied conversational agents in verbal and non-verbal communication
Björn Granstrom 292

Aplicaciones de las tecnologías del habla en sistemas CALL y CAPT
Néstor Becerra Yoma 293

Third-generation conversational interfaces
Giuseppe Riccardi 295

SESIÓN ORAL 1
MODELADO ACÚSTICO

GENERALIZED GAUSSIANS FOR CONTINUOUS OBSERVATION DISTRIBUTIONS IN SPEECH RECOGNITION

Antonio Miguel, Eduardo Lleida, Alfonso Ortega

Communication Technologies Group (GTC), I3A, University of Zaragoza, Spain.

ABSTRACT

One of the most successful models for speech recognition has been the HMM with mixture of Gaussians in the states to generate/capture observations. In this work we show how the addition of a parameter to model higher order moment statistics, such us the kurtosis, can provide improvements to the system. The distributions in which this degree of freedom is integrated are the generalized Gaussians. It is shown a method to estimate the parameters of these distributions even if they are embedded in a HMM or mixture of distributions. Some experimental results are obtained with this method compared to baseline systems of full and diagonal covariance matrices.

1. INTRODUCTION

This paper offers a new approach to model more accurately the observation generation process in the states of the HMM. The working hypothesis for the approach of this paper is that there is information in the speech signal which is not accurately captured by standard models in the states of the HMMs, usually GMMs with diagonal covariance matrix.

In this paper we propose to increase the complexity of the pdfs which are the components of the mixture in the states of the HMM by adding a degree of freedom to control a higher order moment, the kurtosis. The basic idea is that the new pdfs should be able to capture or generate data with statistics beyond the normal distribution.

The goal of models in speech recognition is to keep the maximum of information that we think it is useful to recognize speech, not to synthesize a speech waveform. In this work we try to improve the quality of the statistics captured from the speech signal in order to capture the maximum of information. To do so, a modification in the nature of the Gaussian distribution is proposed. The proposed probability density function is the Generalized Gaussian distribution [1, 2], this distribution has an additional parameter over the normal distribution which controls the fourth order moment. For selected values of this parameter the distribution can adapt its shape to many symmetric distributions as the normal, the Laplacian, or even the uniform and Dirac's delta for extreme values. The generalized Gaussian distribution provides a richer mechanism to adapt to feature statistics.

Some authors have contributed to similar lines of research to enhance the models ability to generalize but the application of the generalized Gaussian distributions in the generation of observations is novel. The generalized Gaussian is an interesting distribution but to be useful in speech recognition, two additional mechanisms are proposed to complete those models. The first one is related to the fact that the generalized Gaussian, as the Gaussian, is not a multimodal distribution. In order to adapt to the complex statistics of the speech signal a hidden variable mechanism is needed to explain the observation in a more accurate way. This is achieved with the mixtures of generalized Gaussians. The parameter estimation will demand a modification in the standard EM algorithm based in the method of moments. The second one is to consider a method to reduce the amount of correlation in the features that we want to model. The features we want to model are usually vector valued. The simplest approach to face multivariate distributions is to use a Naive Bayes approach assuming independence between the features. The proposal to overcome this simplicity in order to find more complex dependences is to assume a linear transformation of the vector by means of a rotation which preserves the scale of the projected vectors.

This paper is organized as follows. In Section 1 there is an introduction. In Section 2 the generalized Gaussian is described. In Section 3 the parameter estimation is discussed. In Section 4 a rotation is included to model covariance.In Section 5, the estimation in hidden variable structures is presented. Experimental results are shown in Section 6 and finally conclusions are in Section 7.

This work has been supported by the national project TIN 2005-08660-C04-01.

2. GENERALIZED GAUSSIAN DISTRIBUTION

The Gaussian distribution is adequate for many problems in speech technologies but is still limited in the sense of modeling accurately distributions with a wide range of high order moments, greater to the second order. The Gaussian distribution has a fixed value of 3 for the kurtosis, which is related to the fourth order moment.

Along this paper we modify the Gaussian fundamental distribution to include an additional parameter which controls the kurtosis of the distribution, This distribution is called a generalized Gaussian (GG) [1, 2] and has the following definition: a continuous real valued random variable is assumed to follow a generalized Gaussian distribution if the probability density function takes the form:

$$x \sim GG(\mu, \sigma, \alpha), \quad x \in \mathbb{R} \tag{1}$$

$$f(x|\mu, \sigma, \alpha) = \frac{\beta(\alpha)}{2\Gamma\left(1+\frac{1}{\alpha}\right)\sigma} e^{-\left|\beta(\alpha)\frac{x-\mu}{\sigma}\right|^{\alpha}}, \tag{2}$$

where α is called the shape parameter and $\beta(\alpha)$ and $\Gamma(x)$ are defined as:

$$\beta(\alpha) = \sqrt{\frac{\Gamma\left(\frac{3}{\alpha}\right)}{\Gamma\left(\frac{1}{\alpha}\right)}}, \quad \Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt \tag{3}$$

The parameter α in (2) controls the kurtosis of the distribution. We can see that for some values of α the distribution equals some well known distributions. For $\alpha = 1$ the expression (2) equals the Laplacian distribution, for $\alpha = 2$ the Gaussian distribution and as α tends to infinity the distributions gets closer to the uniform distribution, $U(\mu - \sqrt{3}\sigma, \mu + \sqrt{3}\sigma)$ and if alpha tends to 0^+ the distribution is closer to a degenerated function, the dirac's delta. This is exemplified in Figure 1, where some examples of the pdf (2) are plot varying the value of the parameter α for a fixed value of $\mu = 0$ and $\sigma = 1$.

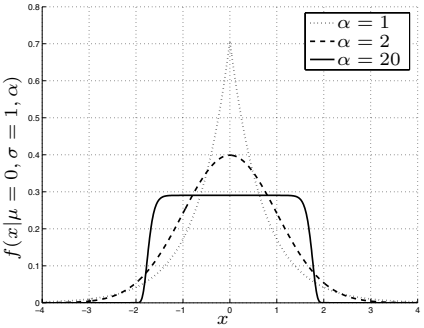


Figure 1. Examples of GG probability density functions for some values of the shape parameter $alpha$.

In [3] a general study about the GGs and its moments is performed. The even order moments of the distribution can be expressed in terms of the parameters . The following is a general expression of a centered moment of even orders r:

$$M_r^* = E[(x-\mu)^r] = \left(\frac{\sigma^2\Gamma\left(\frac{1}{\alpha}\right)}{\Gamma\left(\frac{3}{\alpha}\right)}\right)^{\frac{r}{2}} \frac{\Gamma\left(\frac{r+1}{\alpha}\right)}{\Gamma\left(\frac{1}{\alpha}\right)} \quad (4)$$

The central moments of odd r orders are equal to zero due to the symmetry of the pdf with respect to the mean.

From expression (12) we can see that if the order is $r = 2$:

$$M_2^* = E[(x-\mu)^2] = \sigma^2, \quad (5)$$

where it interesting to note that the variance only depends on the parameter σ in the model pdf, which makes expression (2) a very convenient parametrization. Also it will be of interest for the estimation process the moment of order 4 as will see in next section.

3. ESTIMATION OF PARAMETERS OF GG DISTRIBUTIONS

In [1] the estimation of moments method was proposed, this a simple alternative since in order to fix the three parameters we only have to propose a system of equations with three moments, the mean, the variance and the fourth order as will see. In [2] similar method was proposed to estimate the parameters μ and σ from the moment estimator and the shape parameter using a expression that related the variance, the mean of the absolute values and the shape parameter. In our work we are going to focus on the moment estimation method, we will argue some reasons for this selection due to the nature of the model, the HMM, that the pdf is going to be embedded into. Firstly we will compare the methods in [2] and [3] with the moment estimation method in [1].

In order to establish the notation, let us consider a random variable X with outcomes $x \in \mathbb{R}$, which is assumed to follow a GG distribution as expressed in (2). The training is set is defined as $\mathbf{X} = \{x_1, \ldots, x_n, \ldots, x_N\}$. The pdf is going to be estimated from a i.i.d.(identically distributed) sequence of samples from the variable X.

The parameters μ and σ in all those methods are estimated with equal expressions, the standard moments mean and variance. The expressions are:

$$\hat{\mu} = M_1 = \frac{1}{N}\sum_{n=1}^{N} x_n, \quad \hat{\sigma}^2 = M_2^* = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu})^2, \quad (6)$$

The shape parameter can be estimated using the moment estimation method. Since all odd order centered moments are zero, and the shape parameter only is present in the centered moments of order $r \geq 4$, then this method is based on the calculation of a moment in this range of orders and also the sample estimator for this moment.

$$M_r^* = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu})^r, \quad (7)$$

for any r even and $r \geq 4$.

Then, considering the simplest case $r = 4$, we can obtain the value of α from the following expression:

$$M_4^* = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{\mu})^4 = \left(\frac{\hat{\sigma}^2\Gamma\left(\frac{1}{\alpha}\right)}{\Gamma\left(\frac{3}{\alpha}\right)}\right)^2 \frac{\Gamma\left(\frac{5}{\alpha}\right)}{\Gamma\left(\frac{1}{\alpha}\right)}, \quad (8)$$

In order to reduce the sensitivity of the estimation compared to the calculation of the fourth order moment, an alternative method was proposed in [2]. It was demonstrated the following dependency of a function of the shape parameter with the mean of the absolute value of the random variable:

$$\frac{\hat{\sigma}^2}{(E[|x-\hat{\mu}|])^2} = \frac{\Gamma\left(\frac{1}{\alpha}\right)\Gamma\left(\frac{3}{\alpha}\right)}{\left(\Gamma\left(\frac{2}{\alpha}\right)\right)^2}. \quad (9)$$

which involves lower order moments computations and more estimation accuracy.

Now we discuss which is the best method to estimate the parameters of the generalized Gaussian distribution in order to be easily embedded in an HMM and as a component of a mixture of models. A first approach to the previously presented estimation methods shows that the method based on the absolute value mean is more robust and accurate than the method of moments, which is based on the estimation of moments of orders $r = 4$ and $r = 2$.

Nevertheless, there is an important argument in favour of the method of moments which is the implementation convenience. The estimators in the method of moments with orders $r = 4$ and $r = 2$ can be implemented in one pass over the training data. Therefore the integration in a multiple iteration training procedure as the EM algorithm as an HMM or a mixture of models will be more natural.

The estimators of the centered moments cannot be directly computed in one pass in the training process since while we have access to the samples x_n in the training process we have not calculated the mean for that iteration. In the following expression we perform some algebraic manipulations and the expressions are transformed to:

$$M_2^* = \frac{1}{N}S_2 - \frac{1}{N^2}(S_1)^2 \quad (10)$$

and

$$M_4^* = \frac{1}{N}S_4 - \frac{4}{N^2}S_1 \cdot S_3 + \frac{6}{N^3}(S_1)^2 \cdot S_2 - \frac{3}{N^4}(S_1)^4 \quad (11)$$

Therefore, the moment estimators can be implemented in one pass and the only operations needed during the iteration are the accumulation of powers of the samples. The accumulators are defined as $S_r = \sum_n x_n^r$ for orders $r = 1, 2, 3, 4$.

4. DIAGONALIZATION OF GG DISTRIBUTIONS

Usually, the first approach to a multivariate model is to use the independence assumption of the Naive Bayes approach. Let us suppose that we are modeling the probability density function of a D-dimensional feature vector, $\mathbf{x} = (x_1, \ldots, x_d, \ldots, x_D)$. We can assume that each individual component of the vector follows a GG.

The likelihood of that model can be expressed as:

$$\mathbf{x} \sim GG_D(\mu, \sigma, \alpha), \quad x \in \mathbb{R}^D \tag{12}$$

$$f(\mathbf{x}|\mu, \sigma, \alpha) = \prod_d^D f(x_d|\mu_d, \sigma_d, \alpha_d) \tag{13}$$

$$= \prod_d^D \frac{\beta(\alpha_d)}{2\Gamma\left(1+\frac{1}{\alpha_d}\right)\sigma_d} e^{-\left|\beta(\alpha_d)\frac{x_d-\mu_d}{\sigma_d}\right|^{\alpha_d}}, \tag{14}$$

where each component of the vector is modeled by an independent GG distribution, with parameters μ_d, σ_d and α_d. It is clear from the independence approximation that the estimation of the parameters can be performed with the method of moments for each component of the feature vector separately. The limitation of this technique is similar to the limitation of a Gaussian with diagonal covariance matrix, which is a special case of the previous model, where all the values $\alpha_d = 2$.

We propose the use of a linear transformation to reduce the correlation between the variables in the vector. The method consists in the estimation of a linear transformation matrix, $\mathbf{A}$, to transform the random vector $\mathbf{x}$ to a vector $\mathbf{y}$ as:

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{15}$$

where the objective is that the components of random vector $\mathbf{y}$ can be considered independent and its covariance matrix be diagonal. The problem of diagonalizing a covariance matrix is the classic problem of the principal component analysis (PCA).

The probability density function of the resulting random variable which is obtained applying a function $\mathbf{y} = g(\mathbf{x}) = \mathbf{A}\mathbf{x}$ over the existing variable $\mathbf{y}$ is:

$$f_y(\mathbf{y}) = f_x(g^{-1}(\mathbf{y}))\left|\frac{\delta g^{-1}(\mathbf{y})}{\delta \mathbf{y}}\right| = f_x(\mathbf{A}^{-1}\mathbf{y})\left|\mathbf{A}^{-1}\right| \tag{16}$$

One of the properties that it would be desiderable for the linear transformation is that $|\mathbf{A}^{-1}| = 1$, so that we can easily apply the Naive Bayes GG distribution to the transformed vectors $\mathbf{x}$. This is also desiderable since no further re-scaling of the likelihood is needed, this provides an important simplicity if the likelihood is going to be compared or operated with other likelihoods as in a mixture of models or in HMMs.

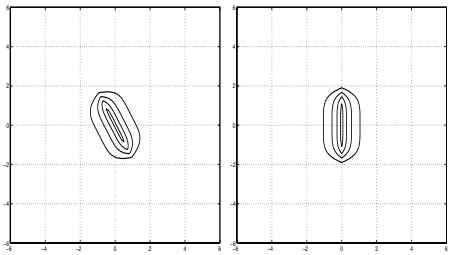
The question that remains is how to calculate the transformation $\mathbf{A}$. This problem can be solved considering two expressions: the relationship of the covariance matrices of variables in a linear transformation and the decomposition theorem. Given a random vector $\mathbf{x}$ with a full covariance matrix $\mathbf{\Sigma}_x$, then the covariance matrix of the random vector $\mathbf{y} = \mathbf{A}\mathbf{x}$ is $\mathbf{\Sigma}_y = \mathbf{A}\mathbf{\Sigma}_x\mathbf{A}^T$. The eigendecomposition a semidefinite positive matrix $\mathbf{V}$ can be obtained as $\mathbf{V} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where the matrix $\mathbf{\Lambda}$ is a diagonal matrix with the eigenvalues of $\mathbf{V}$ as the diagonal elements and the matrix $\mathbf{U}$ are the eigenvectors of $\mathbf{V}$ as columns.

If we use both results we can find the linear transformation of the variable $\mathbf{y}$ which makes the covariance matrix $\mathbf{\Sigma}_x$ diagonal as the eigen vectors of the matrix $\mathbf{\Sigma}_y$ by columns.

In the Figure 2 there is an example, we can see the pdf of some artificially generated data of a random vector of dimension $D = 2$. We can see that after the linear transformation, for the pdf in Figure 2b the main variation axis are the cartesian coordinate system which makes possible the application of a Naive Bayes GG distribution.

5. MIXTURES OF GG DISTRIBUTIONS

Similarly to the case of multivariate Gaussian distributions, an unimodal pdf is not an accurate model for the complexity of the speech signal. The mixture model is a natural solution to increase the modes of a pdf and to adapt to higher complexities in the data.



(a) (b)

Figure 2. Example of 2D GG distributions. a) A rotated space GG distributrion b) Naive Bayes multivariate GG

A mixture of GG distributions of C components is defined as a weighted sum of the pdfs of the components in the following way:

$$f(x) = \sum_{c=1}^{C} p_c \cdot \frac{\beta(\alpha_c)}{2\Gamma\left(1+\frac{1}{\alpha_c}\right)\sigma} e^{-\left|\beta(\alpha_c)\frac{x-\mu_c}{\sigma_c}\right|_c^{\alpha}}, \tag{17}$$

where for simplicity all the derivations in this section are expressed in terms of an univariate GG.

The previous expression can be considered the marginalization of hidden discrete variable Z that can take values $z \in \{1, \ldots, C\}$ which selects a from a pull of GG distributions as shown in [4]. This variable is assumed to follow a Multinomial distribution. The pdf of the variable Z is a Multinomial of size $z+ = 1$ and prototype vector $\mathbf{p} = (p_1, \ldots, p_c, \ldots, p_C)$,

$$Z \sim Mult(1, \mathbf{p}), \quad z \in \{1, \ldots, C\}, \quad p(Z = z) = \prod_{c=1}^{C} p_c^{\delta_{z,c}}, \tag{18}$$

The estimation of the parameters of a probability model in which there are hidden variables involved is usually solved with the EM algorithm [5] The E step auxiliary function is:

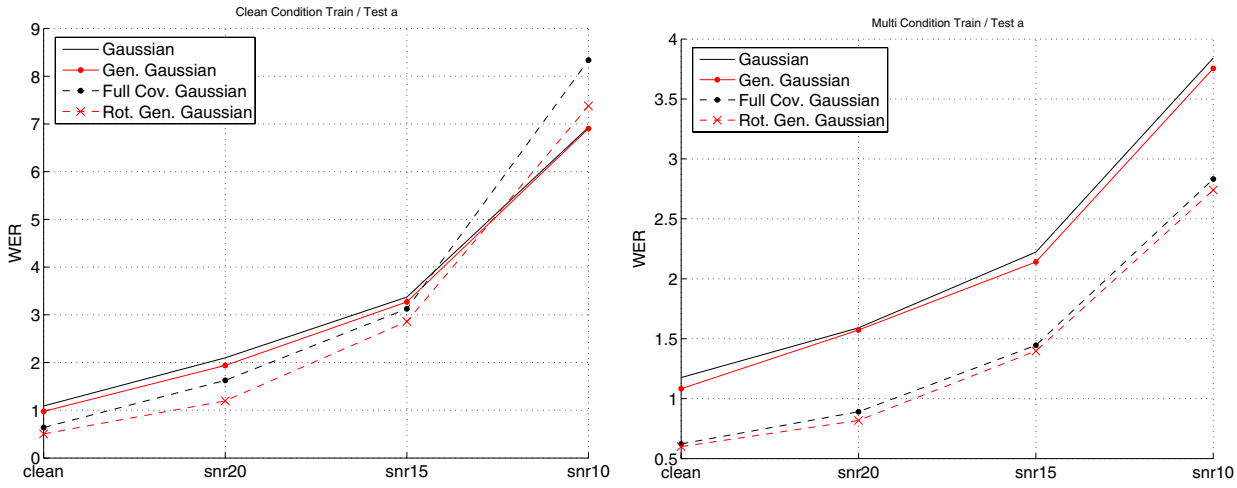
$$\begin{aligned} Q(\mathbf{\Theta}|\mathbf{\Theta}^{(k)}) &= E[\log p(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta})|\mathbf{X}, \mathbf{\Theta}^{(k)}] \\ &= \sum_n \sum_c \langle \delta_{Z,c} \rangle_{Z|_{x_n}}^{(k)} \cdot (\log p_c + \\ &\quad + \log\left(\beta(\alpha_c) 2\Gamma\left(1+\frac{1}{\alpha_c}\right)\sigma_c\right) - \left|\beta(\alpha_c)\frac{x_n-\mu_c}{\sigma_c}\right|^{\alpha_c}\Bigg) \end{aligned} \tag{19}$$

where $\langle \delta_{Z,c} \rangle_{Z|_{x_n}}^{(k)}$ is a short notation for the expected value of the function $\delta_{Z,c}$ of the variable Z conditioned to $X = x_n$:

$$\begin{aligned} \langle \delta_{Z,c} \rangle_{Z|_{x_n}}^{(k)} &= E_Z[\delta_{Z,c}|x_n, \mathbf{\Theta}^{(k)}] \\ &= \sum_{\forall z} \delta_{z,c} \cdot p(Z = z, |x_n, \mathbf{\Theta}^{(k)}) \\ &= p(Z = c|x_n, \mathbf{\Theta}^{(k)}) \\ &= \frac{p(Z = c|\mathbf{\Theta}^{(k)}) \cdot f(x_n|Z = c, \mathbf{\Theta}^{(k)})}{\sum_{c'} p(Z = c'|\mathbf{\Theta}^{(k)}) \cdot f(x_n|Z = c', \mathbf{\Theta}^{(k)})} \end{aligned} \tag{20}$$

where $f(x_n|Z = c, \mathbf{\Theta}^{(k)})$ is the component specific GG pdf.

It is possible to find an alternative to the EM direct estimation where the maximization step is substituted by a special moment estimation. This algorithm which is called expectation moment estimation (EME). It can be applied in general to any distribution for which we have defined moments of the distribution in terms of the parameters.



(a) (b)

Figure 3. Experimental WER results for Aurora2 database test set a, for moderate noisy conditions for the different techniques a) clean condition train, b) multi condition train.

The main result of the algorithm is the moment estimation ME step. The ME step, provides estimates for the moments of the distributions which are components of the mixture as a function of the expected values previously calculated $\langle \delta_{Z,c} \rangle^{(k)}_{Z|_{x_n}}$. The first order moment for the component c of the mixture is computed as:

$$M_{1,c}^{(k+1)} = \mu_c^{(k)} = \frac{\sum_{n=1}^{N} \langle \delta_{Z,c} \rangle^{(k)}_{Z|_{x_n}} \cdot x_n}{\sum_{n=1}^{N} \langle \delta_{Z,c} \rangle^{(k)}_{Z|_{x_n}}}, \qquad (21)$$

And the centered moments of order r for the component c of the mixture:

$$M^{*}{}_{r,c}^{(k+1)} = \frac{\sum_{n=1}^{N} \langle \delta_{Z,c} \rangle^{(k)}_{Z|_{x_n}} \left(x_n - \mu_c^{(k+1)} \right)^r}{\sum_{n=1}^{N} \langle \delta_{Z,c} \rangle^{(k)}_{Z|_{x_n}}}. \qquad (22)$$

Depending on the number of free parameters in the model a number of these EME expressions will be needed. For a generalized Gaussian distribution, the number of equations needed is three: $M_{1,c}^{(k)}$, $M^{*}{}_{2,c}^{(k)}$ and $M^{*}{}_{4,c}^{(k)}$.

The procedure to estimate in a single pass the EME $r = 2$ and $r = 4$ order moments can be written in a similar way to the previously shown for the moment estimation method.

6. EXPERIMENTS

The different proposals in this paper have been evaluated on the Aurora 2 task [6] which is a connected digit strings recognizing task in different noise environments. The feature set are the adv ETSI front-end features [7], and the baseline system has been trained with HMM word models of 14 states and 3 component Gaussian mixtures for the digits, a 1 state with 6 components model for the inter-word silence unit and a 3 state with 6 components model for the begin-end silence unit. The models were trained with 20 iterations of the EM algorithm.

In Figure 3 we can see experiments performed in Aurora 2 corpus. There are two baseline systems the observation distribution is a Gaussian mixture in both of them but in a case there are diagonal covariance matrices and in the other full covariance matrices. The results for the GG distributions are also shown where we can see that the error is below the corresponding baseline system in all the cases. We compare the mixture of Gaussians system with the mixture of GG system, and the full covariance Gaussian system with rotated GG, performed as shown in Section 4. The mean WER (word error rate) reduction for noise free condition training (clean) in clean test set a is 10.3% of GGs with respect to Gaussians and 21.1% of rotated GGs with respect to full covariance Gaussians.

7. CONCLUSIONS

In this work it has been shown a method to model high order moments with a distribution in which a parameter related with the kurtosis can be configured. Some methods are provided to estimate the parameters of these distributions, and also the solution to the estimation when the distributions are the outputs of a model with hidden variables. We have seen that the models with the additional degree of freedom perform better than the baseline system specially in noise free conditions.

8. REFERENCES

[1] M.K. Varanasi y B. Aazhang, "Parametric generalized gaussian density estimation," *J. Acoustical Society America*, vol. 86 (4), pp. 1404–1415, 1989.

[2] K. Sharifi y A. Leon-Garcia, "Estimation of shape parameter for generalized gaussian distribution in subband decomposition of video," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 5, no. 1, pp. 52–56, 1995.

[3] R. M. Rodríguez-Dagnino J. A. Domínguez-Molina, G. González-Farías, "A practical procedure to estimate the shape parameter in the generalized gaussian distribution," Tech. Rep., Centro de Investigación en Matemáticas, México, 2000.

[4] A. Juan y E. Vidal, "On the use of Bernoulli mixture models for text classification," *Pattern Recognition*, vol. 35, no. 12, pp. 2705–2710, December 2002.

[5] A. P. Dempster, N. M. Laird, y D. B. Rubin, "Maximum likelihood from incomplete data via the *EM* algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–21, 1977.

[6] H. G. Hirsch y D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ISCA ITRW ASR2000 "Automatic Speech Recognition: Challenges for the Next Millennium"*, Paris, France, September 2000, pp. 18–20.

[7] "ETSI ES 202 050 v1.1.1 Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms," July 2002.

GRAPHICAL MODELS FOR DISCRETE OBSERVATION DISTRIBUTIONS IN SPEECH RECOGNITION

Antonio Miguel, Eduardo Lleida, Alfonso Ortega

Communication Technologies Group (GTC), I3A, University of Zaragoza, Spain.

ABSTRACT

Speech recognition has been traditionally associated to continuous random variables. The most successful models have been the HMM with mixture of Gaussians in the states to generate/capture observations. In this work we show how the graphical models can be used to extract the joint information of more than two features, which is not modeled with full covariance matrices. This is possible if we previously quantize the speech features to a small number of levels and work with discrete random variables. It is shown a method to estimate a constrained number of parents subset of the directed acyclic graph based model framework. Some experimental results are obtained with this method compared to baseline systems of full and diagonal covariance matrices. Additionally, it is shown that it is possible to improve the information of the discrete random variable with qualitative features, such us voicing class or pitch information.

1. INTRODUCTION

The modeling of discrete variable probability distributions is a very interesting problem specially when dealing with high dimensional variables and all their complex interactions [1]. Nowadays, it is a very active field of research in the scope of pattern recognition, machine learning and intelligent systems.

In this paper we propose the use of the graphical model approach to discover underlying dependency structures in the process of generation of observations in the states of the HMM (Hidden Markov Model). Usually the joint distribution of the random feature vector which are the observations of a HMM are modeled to follow a GMM (Gaussian Mixture Model) with diagonal covariance matrix. When more accuracy is required and enough training data are available the covariance matrices are assumed to be not diagonal. In this case, the most complex relationship considered between components of the vector is a pair of components.

In this work it is proposed to quantize the components of the speech feature vector. The resulting quantized random vector can be described now with a disc rete random variable joint distribution. The techniques proposed in the paper try to approximate the joint distribution of all the components by taking approximations. It will be shown that for a certain level of complexity, a good approximation is given by a factorization described by a directed acyclic graph with a constrained number of parents per node.

Many authors have previously contributed to this line of research from different areas, [2], but the application of these tools in acoustic modeling is limited. There have been applications of graphical models or Bayesian networks applied as an alternative to the HMM independence assumption, to build language models or spoken automatic dialog systems. These have been more natural fields to develop techniques based on discrete probabilities since, stochastic approaches to language modeling and dialog systems use discrete variables referring to words or, dialog acts over limited size sets, such us vocabularies. These kind of applications suit perfectly the graphical model ability to exploit the potential of intricate hidden dependencies among large number of variables.

This paper is organized as follows. In Section 1 there is an introduction. In Section 2 the quantization of the speech is presented. In Section 3 the factorization is described. In Section 4, the parameter estimation is derived. Experimental results are shown in Section 5 and finally conclusions are in Section 6.

2. QUANTIZED SPEECH FEATURES

Most of today's systems for speech recognition are based on statistical approaches for modeling the process of emission of observations in the HMM. From the statistical point of view the speech signal can be considered a very complex process. Not only the non-stationary nature of the signal but also the variability of observations or measurements we can get have a wide range across speakers, environments, or even for a same individual. The exact modeling of all of these sources of uncertainty in a brute force approach is not affordable since it would require astronomic sizes of models, training data and computing time. It is important to build affordable systems to provide mechanisms able of making approximations and generalize the knowledge. In this work discrete variables are used to model the speech signal, so that we are able to learn joint distributions of the speech features.

In order to perform the quantization process, a very simple process is proposed for this work. First we take the complete training corpus and, after extracting all the feature vectors, evaluate some simple statistics as the histogram. Then, we find a number of areas with an approximate probability mass, same percentile. The limits between these areas will serve to build the quantizer. This process can be seen equivalent to construct a histogram equalization transformation function with an uniform target distribution and quantize uniformly. the objective is to build a quantizer so that each quantized level represents the same amount of mass of probability in the input signal.

We should note that the process here described for building the quantizer is simple and more optimum solutions can be proposed, since once it has been obtained from the training set it remains unaltered for all the experiments. There also exist the potential future possibility of incorporating a real time implementation of a histogram equalization system, which can be interesting under mismatch of signal and models because of different training and testing conditions. In that case the histogram could be estimated based on a temporal window around the current feature vector.

3. FACTORIZED PROBABILITY DENSITY FUNCTIONS

To define the distribution associated in general to a D dimensional random vector $\mathbf{x} = (x_1, \ldots, x_d, \ldots, x_D)$, where each component of the vector z_d is a discrete variable with outcomes $x_d \in \{1, \ldots, M\}$, where M is the number of levels after the quantization and D is the dimension of the feature space. The joint

This work has been supported by the national project TIN 2005-08660-C04-01.

distribution can be expressed mathematically in the following way, without loss of generality:

$$p(\mathbf{x}) = p(x_1) \prod_{d=2}^{D} p(x_d|x_1, ..., x_{d-1}), \quad (1)$$

where we apply the Bayes theorem recursively, and about the notation $p(\cdot)$ is used for density function and $P(\cdot)$ stands for probability of event.

When the size of the feature vectors grows, the joint model is intractable, since we would need to estimate $|\Theta| = M^D - 1$ parameters. The naive approximation consists on ignoring all the dependencies in the general term of the previous expression.

The objective in the graphical model approach is to find a way of describing the interactions and independencies between the variables of a probabilistic model, and represent as much useful information from data into models as possible. This information can be conveniently represented in the form of graphs [2, 1].

The pdf of interest for us is, as we have noted before, is the pdf of the generic probabilistic process in the state of a HMM. Also it can be a component in a mixture for each state, instead of a simple pdf. The exact pdf to model the observations or feature vectors is the joint pdf (1), but, as we have said, in most cases is intractable.

In [3] was proposed a method for storaging pdfs based on a convenient "factorization" of the exact pdf. The process consisted on the selection of an appropriate order for the index of the variables, the factorization as in (1) following this order and the approximation of the conditioned distributions by more simpler ones. In [3] the number of dependencies of each variable did not exceed one.

The combination of the order and the approximations to factorize a joint pdf can be explained with a DAG (Directed Acyclic Graph) [2, 1]. The probability structure described by the graph is also called Bayesian Network. To build a a factorization model, each variable in the model x_d is associated to a node in the graph v, therefore the size of the graph is $V = D$. The pdf given by a graph can be expressed as:

$$p(\mathbf{x}) \simeq \prod_{v=1}^{V} p(x_v|\boldsymbol{\pi}(v)), \quad (2)$$

where the expression $\boldsymbol{\pi}(v)$ denotes the dependencies associated to node v, which are the set of parent of the node v in the graph. The naive Bayes models are the case of $\boldsymbol{\pi}(\cdot) = \emptyset$ for all the variables.

3.1. Constrained order dependency models

For a given node, the number parent nodes defines the dependencies of the variable with respect to other variables in the graph. If this number of dependencies is high, the number of parameters in the model is large and more data are necessary to estimate accurately those parameters. The kind of model we propose is a subset in the factorizations provided by the directed acyclic graphs, where the complexity of the target factorization is controlled as a parameter of the model. The objective is to find the best graph so that the number of parameters remains low and the approximation to the joint distribution is good enough and keeps most of the information.

The proposed approach, the Constrained Directed Acyclic Graph (CDAG), is a DAG with a limited number of parents. The order r is the maximum number of parents in the graph.

As a first approach we can express the model pdf of a CDAG(r) as follows:

$$p(\mathbf{x}) \simeq \prod_{v=1}^{V} p(x_v|\boldsymbol{\pi}(v)) = \prod_{v=1}^{V} p(x_v|\pi_1(v), \ldots, \pi_r(v)), \quad (3)$$

where $\pi_1(v)$ is the first component of the parent set (of size r) for the node v. We have to note that if the order the model, r, is set to zero, then we have the naive Bayes model, CDAG(0), and if the order is set to one, then we have the [3] tree model, CDAG(1). For simplicity in the notation we are going to express the model probability and estimate the parameters for an order $r = 2$, but the method is also applicable to larger orders.

We propose the adjacency matrix of the graph, $\mathbf{A}$, to establish a more convenient notation. The term $p(x_v|\pi_1(v), \pi_2(v))$ for $r = 2$ in (3), can be expressed as:

$$p(x_v|\pi_1(v), \pi_2(v)) = \prod_{v'} \prod_{v''} [p(x_v|x_{v'}, x_{v''})]^{(a_{v',v} \cdot a_{v'',v})}, \quad (4)$$

where $a_{v',v}$ is a component of adjacency matrix which is equal to one if the node v' is a parent of node v.

3.2. CDAG model likelihood

In order to express the model probability using the adjacency matrix notation, we have to define an indicator vector $\mathbf{b}$, with only one element equal to one, which is the index of the node without parents. The value of the components of $\mathbf{b}$ can be expressed as:

$$b_v = \begin{cases} 1 & \text{if } \sum_{v'=1}^{V} a_{v',v} = 0, \\ 0 & \text{cc.} \end{cases} \quad (5)$$

The expression (3) can be written using the adjacency matrix, $\mathbf{A}$, and the vector $\mathbf{b}$ as follows:

$$p(\mathbf{x}) \simeq \prod_{v} [p(x_v)]^{b_v} \cdot \prod_{v'} \prod_{v''} [p(x_v|x_{v'}, x_{v''})]^{(a_{v',v} \cdot a_{v'',v})} \quad (6)$$

where if $a_{v',v}$ and $a_{v'',v}$ are equal to one, the pair of edges (v', v) and (v'', v) are in the graph, and the factor $p(x_v|x_{v'}, x_{v''})$ contributes to the product .

In order to achieve a more compact notation and simpler estimation derivation, we augment the information represented in the adjacency matrix to a matrix $\mathbf{R} = \mathbf{A} + \mathbf{b} \cdot \mathbf{I}$. Introducing the augmented matrix notation, the expression (6) can be now written as:

$$p(\mathbf{x}) \simeq \prod_{v} \prod_{v'} \prod_{v''} [p(x_v|x_{v'}, x_{v''})]^{(r_{v',v} \cdot r_{v'',v})}, \quad (7)$$

where this representation is also more compact because we consider the special cases:

$$p(x_v|x_{v'}, x_{v''}) = \begin{cases} p(x_v) & v = v', v = v'' \\ p(x_v|x_{v'}) & v = v'' \\ p(x_v|x_{v''}) & v = v' \\ p(x_v|x_{v'}, x_{v''}) & cc \end{cases}. \quad (8)$$

We can express the previous expression (7) as:

$$p(\mathbf{x}) \simeq \prod_{v,v',v''} \prod_{m',m''} [p(x_v|x_{v'}=m', x_{v''}=m'')]^{(r_{v',v} r_{v'',v})(\delta_{x_{v'},m'} \delta_{x_{v''},m''})}, \quad (9)$$

In the previous expression, we can identify the distributions in the factors as Multinomials:

$$x_v|_{x_{v'}=m', x_{v''}=m''} \sim Mult_M(1, \mathbf{p}_{v,v',v'',m',m''}). \quad (10)$$

where $\mathbf{p}_{v,v',v'',m',m''}$ is the prototype vector, i.e. the histogram which gives us the probability of the M possible values of the conditioned variable. The components of the prototype vector can be expressed as:

$$p_{v,v',v'',m,m',m''} = P(x_v = m|x_{v'} = m', x_{v''} = m'') \quad (11)$$

Then we can express the conditioned variable distribution as:

$$p(x_v|x_{v'}=m',x_{v''}=m'',\mathbf{P})=\prod_m[p_{v,v',v'',m,m',m''}]^{\delta_{x_v,m}}, \tag{12}$$

with the constraint $\sum_{m=1}^{M} p_{v,v',v'',m,m',m''}=1$ for all $v,v',v''=1,\ldots,V$ and $m',m''=1,\ldots,M$.

For a set of parameters $\mathbf{\Theta}=(\mathbf{P},\mathbf{R})$, the log likelihood function for a training set, $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ is:

$$L(\mathbf{\Theta};\mathbf{X})=\sum_n\sum_{v,v',v''}\sum_{m,m',m''}(r_{v',v}r_{v'',v})\times \\ \times\left(\delta_{x_{nv},m}\delta_{x_{nv'},m'}\delta_{x_{nv''},m''}\right)\cdot\log p_{v,v',v'',m,m',m''} \tag{13}$$

4. CDAG PARAMETER ESTIMATION

In order to estimate the optimum set of parameters to maximize the log likelihood function we have to solve the following optimization:

$$\{\hat{\mathbf{P}},\hat{\mathbf{R}}\}=\arg\max_{\mathbf{P},\mathbf{R}} L(\mathbf{P},\mathbf{R};\mathbf{X}), \tag{14}$$

subject to $\sum_{m=1}^{M} p_{v,v',v'',m,m',m''}$ for all $v,v',v''=1,\ldots,V$ and $m',m''=1,\ldots,M$, and subject to $\mathbf{R}\in$CDAG(2).

It is possible to show that the optimum parameter subset $\hat{\mathbf{P}}$, can be solved independently of the topology of the graph as:

$$\hat{p}_{v,v',v'',m,m',m''}=\frac{\sum_n\delta_{x_v,m}\delta_{x_{v'},m'}\delta_{x_{v''},m''}}{\sum_n\delta_{x_{v'},m'}\delta_{x_{v''},m''}}. \tag{15}$$

where in the numerator we have the number of feature vector examples in the training data whose v component is equal to the value m, the component v' is equal to m' and the component v'' is equal to m''. The numerator can be interpreted in this sense too, and together we can see that the parameter $\hat{p}_{v,v',v'',m,m',m''}$ estimates the probability $\hat{p}(x_v=m|x_{v'}=m',x_{v''}=m'')$.

The optimum set of parameters $\hat{\mathbf{R}}$, provides the edge set and the graph will be fully characterized. Once we have found the set of parameters $\hat{\mathbf{P}}$, we can obtain with a convenient manipulation an optimization similar to [3]:

$$\hat{\mathbf{R}}=\arg\max_{\mathbf{R}}\sum_{v,v',v''}(r_{v',v}r_{v'',v})\hat{I}(x_v||x_{v'},x_{v''}), \tag{16}$$

subject to $\mathbf{R}\in$CDAG(2). Where, with the notation $\hat{I}(x_v||x_{v'},x_{v''})$ we refer to the mutual information:

$$\hat{I}(x_v||x_{v'},x_{v''})= \\ \sum_{\forall x_v,x_{v'},x_{v''}}\hat{p}(x_v,x_{v'},x_{v''})\log\frac{\hat{p}(x_v,x_{v'},x_{v''})}{\hat{p}(x_v)\hat{p}(x_{v'},x_{v''})}, \tag{17}$$

which is also the Kullback-Leibler divergence between the distributions $\hat{p}(x_v,x_{v'},x_{v''})$ and $\hat{p}(x_v)\cdot\hat{p}(x_{v'},x_{v''})$, i.e. joint and approximated respectively.

It is interesting that as in [3] we obtain the same solutions (15) and (16) by minimizing the Kullback-Leibler divergence between the approximate and the exact model $D(p(\mathbf{x})||\hat{p}(\mathbf{x}))$.

4.1. Approximated algorithm for graph building

The exact algorithm to find the best graph from this expression is a hard problem, but a fast but approximate algorithm to estimate the best graph is proposed in this paper. The objective is to find an algorithm to obtain the best graph in terms of maximum likelihood,

Algorithm 1 Approximate optimum graph to obtain a CDAG(2)

Input: Random samples $\mathbf{X}$
Output: The graph $\hat{\mathbf{R}}$ of the class CDAG(2)
1. Initialization
Initiate graph matrix:
$\hat{\mathbf{R}}\leftarrow\mathbf{0}$
Estimate all $\hat{p}(x_v,x_{v'},x_{v''})$
Calculate all $\hat{I}(x_v||x_{v'},x_{v''})$
Initiate set of non assigned nodes, $\mathcal{N}$:
$\mathcal{N}\leftarrow\{x_1,\ldots,x_V\}$
Order decreasingly all $\hat{I}(x_v||x_{v'},x_{v''})$ so that:
$\hat{I}(x_{m_{1,1}}||x_{m_{1,2}},x_{m_{1,3}})>\hat{I}(x_{m_{2,1}}||x_{m_{2,2}},x_{m_{2,3}})>\ldots$
2. Search edges
$k\leftarrow 1$
while $|\mathcal{N}|>1$ **do**
  $v\leftarrow m_{k,1}$, $v'\leftarrow m_{k,2}$, $v''\leftarrow m_{k,3}$
  if $x_v\in\mathcal{N}$ **then**
    $\mathbf{R}'\leftarrow\hat{\mathbf{R}}$
    Add edges (v',v) and (v'',v) to $\mathbf{R}'$:
    $r'_{v',v}\leftarrow 1$, $r'_{v'',v}\leftarrow 1$
    if $|\mathbf{I}-\mathbf{R}'|\neq 0$ **then**
      $\hat{r}_{v',v}\leftarrow 1$, $\hat{r}_{v'',v}\leftarrow 1$
      $\mathcal{N}\leftarrow\mathcal{N}\setminus x_v$
    end
  end
  $k\leftarrow k+1$
end
Assign the last variable $x_v\in\mathcal{N}$:
$\hat{r}_{v,v}\leftarrow 1$

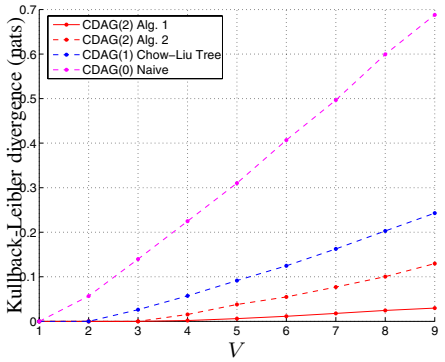
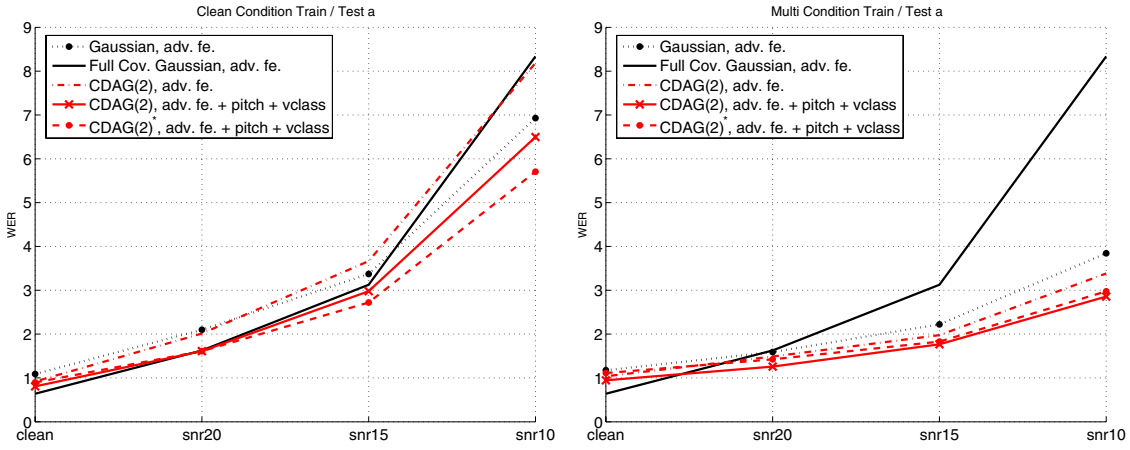


Figure 1. *DKL vs V for artificially generated data.*

which is equivalent to find the set of edges $\hat{\mathbf{R}}$ that maximize expression (16). This problem is trivial for order $r=0$,(the naive Bayes model), and can be solved exactly in polinomial time for order $r=1$, where we will obtain a special kind of graph, a tree, which is shown in [3]. For higher orders such us $r=2$, the problem becomes intractable, we can not subdivide the problem into smaller independent problems and the problem cannot be solved efficiently by dynamic programming approaches.

The approximate Algorithm 1, can be explained as follows. First the joint distributions $\hat{p}(x_v,x_{v'},x_{v''})$ and the Kullback Leibler divergences $\hat{I}(x_v||x_{v'},x_{v''})$ have to be calculated in a initializing phase. Then, the values of the Kullback-Leibler divergences are ordered and the indices of the variables v, v' and v'' are stored in the auxiliary variables $m(k,1)$, $m(k,2)$ and $m(k,3)$ respectively. The next step is the approximate search of the edges to construct the matrix $\hat{\mathbf{R}}$ with a maximum value of the sum of partial Kullback-Leibler divergences for all the edges in the graph, while keeping the graph acyclic. This is done in a loop by adding consecutively pairs of edges (v',v) and (v'',v) following



(a) (b)

Figure 2. Experimental results for Aurora2, test set a, for moderate noisy conditions. a) clean condition train, b) multi condition train.

the previous descending order. Before adding them to the solution, it is checked that the addition of both pairs does not form a cycle. In the last step we have only one variable x_v in $\mathcal{N}$. This variable has no parents, which is marked with $b_v = 1$ or $\hat{r}_{v,v} = 1$.

There is an operation in the search process which can be computationally expensive. It is the determinant calculation, to check if the new graph resulting after the addition of the current edges is acyclic. More efficient searches can be performed if this part is substituted by an incremental check of the acycliness.

The estimation process shown here can be incorporated to a hidden variable structure as a HMM or HMM with mixtures in the states. The results in the experimental section use an EM estimation of the parameters which can be derived as in [4], where is done for Bernouilli mixtures.

There exists an approach [5] to discover conditional independences or the I-map in data sets. In order to do so, there is a first step which involves the computation of terms $I(x_v||x_{v'})$ to construct a preliminary graph, which is an approximation with respect to the more exact mutual information measures the ones used in this work. In later passes of that algorithm CI (conditional independence tests) are performed. The CI test consists in the computation if the mutual information of two variables x_v and $x_{v'}$, given a cutset $\mathcal{C}$, is above certain threshold. The CI test step is carried without restrictions of the order of the joint pdfs involved. Another difference is that our approach is not intended to discover to true underlying graph with any number of parents in the nodes, but a constrained order graph.

5. EXPERIMENTS

A preliminary experiment is show in Figure 1. The performance of the Algorithm 1, compared to the naive Bayes approach, the Chow tree [3], or the approximations performed by algorithms similar to [5] (which is referred to Alg. 2) is shown. Since it is an experiment based on artificial data, we can compute the Kullback-Leibler divergence of the different models with the exact model, the joint distribution. We can see that the CDAG(2) model with Algorithm 1 behaves quite accurately in the experiment.

The proposal in this paper has also been evaluated on the Aurora 2 task [6] which is a connected digit strings recognizing task in different noise environments. The feature set are the extended adv ETSI front-end features [7], and the baseline system has been trained with HMM word models of 14 states and 3 component Gaussian mixtures for the digits, a 1 state with 6 components model for the inter-word silence unit and a 3 state with 6 components model for the begin-end silence unit. The models were trained with 20 iterations of the EM algorithm.

In Figure 2 we can see experiments with Aurora corpus. There are two baseline systems with Gaussian mixtures, but in a case there are diagonal covariance matrices and in the other full covariance matrices. Results for the discrete feature vector systems we obtained. The number of quantization levels was left to $M = 5$. We can see in Figure 2 the results obtained for all the discrete random variable approaches. We also can see that additional WER reduction can be obtained with the addition of more qualitative features such us the voicing class or the pith given by the extended ETSI front-end. The mean WER (word error rate) reduction for the best case in the test set a for the clean, snr20 to snr05 conditions (moderate noise) is a 13.5%.

6. CONCLUSIONS

In this work it has been shown a method to model high dimensional discrete distributions which is based on the assumption of a model of dependencies with a limited number of them. The generalization ability of these factorizations had been previously shown in previous works. We have adapted a previous solution for a constrained order of one to larger orders. The result has been a very interesting class of model with a good accuracy, specially in noise conditions and benefits such us a low transmission rate for the features which we will continue to enhance in future works.

7. REFERENCES

[1] J. Pearl, *Causality: Models, Reasoning, and Inference.*, Cambridge Univ. Press., 2000.

[2] S. Lauritzen, *Graphical Models*, Oxford Univ. Press, 1996.

[3] C. K. Chow y C. N. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Trans. on Information Theory*, vol. 14 (3), pp. 462–467, 1968.

[4] A. Juan y E. Vidal, "On the use of Bernoulli mixture models for text classification," *Pattern Recognition*, vol. 35, no. 12, pp. 2705–2710, December 2002.

[5] J. Cheng, R. Greiner, J. Kelly, D. A. Bell, y W. Liu, "Learning bayesian networks from data: an information-theory based approach," *Artificial Intelligence*, vol. 137, pp. 43–90.

[6] H. G. Hirsch y D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ISCA ITRW ASR2000*, Paris, France, September 2000, pp. 18–20.

[7] "ETSI ES 202 050 v1.1.1 distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," July 2002.

ON THE USE OF AUGMENTED HMM MODELS FOR OVERCOMING TIME AND PARAMETER INDEPENDENCE ASSUMPTIONS IN ASR

Marta Casar and José A. R. Fonollosa

Departament de Teoria del Senyal i Comunicacions,
Universitat Politècnica de Catalunya (UPC)

ABSTRACT

There is significant interest in developing new acoustic models for speech recognition that overcome traditional HMM restrictions. In this work, we propose to use N-gram based augmented HMMs. Two approaches are presented. The first one consists on overcoming the parameter independence assumption. This is achieved by modeling the dependence between the different acoustic parameters, using N-gram modeling. Then, the input signal is mapped to the new probability space. The second proposal tries to overcome the time independence assumption, by modeling temporal dependencies of each acoustic feature. Different configurations have been tested for connected digit and continuous speech recognition, results showing that adding long span information is beneficial for ASR performance.

1. INTRODUCTION

For modeling temporal dependencies or multi-modal distributions of 'real-world' tasks, Hidden Markov Models (HMM) are one of the most commonly used statistical models. Because of this, HMMs have become the standard solution for modeling acoustic information in the speech signal and thus for most current speech recognition systems. When putting HMMs into practice, however, there are some assumptions that, even if effective, are known to be poor [1], degrading classification performance. Adding dependencies through expert knowledge and hand tuning can improve models, but it is often not clear which dependencies to include. Therefore, the development of new acoustic models that overcome traditional HMM restrictions is an active field of research in Automatic Speech Recognition (ASR).

In order to overcome HMM limitations, many extensions have been proposed. One interesting approach for allowing complex dependencies to be represented are augmented statistical models [2], which are used in this paper in a new framework for dealing with temporal and parameter dependencies while still working with regular HMMs.

This work has been partially supported by the TECNOPARLA project, granted by the Catalan Government

2. MODELING TIME AND PARAMETER DEPENDENCES

In HMMs there are some assumptions that make evaluation, learning and decoding feasible. Among them, the Markov assumption for the Markov chain [1] states that the probability of a state s_t depends only on the previous state s_{t-1}. Also, when working with different parameters to represent the speech signal, we rely on the parameter independence assumption. It states that the acoustical parameters modeled by HMMs are independent, and so are the output symbol probabilities emitted.

However, in many cases, the independence and conditional-independence assumptions encoded in these latent-variable models are not correct, potentially degrading classification performance. For modeling dependencies between features, Gaussian mixture distribution-based techniques are very common. The parametric modeling of cepstral features with full covariance Gaussians using the ML principle is well-known and has led to good performance. However, these techniques are expensive with real-time and/or low resource applications.

For modeling time-domain dependencies, several approaches have focused on studying the temporal evolution of the speech signal to optimally change the duration and temporal structure of words, known as duration modeling [3]. However, incorporating explicit duration models into the HMM structure also breaks some of conventional Markov assumptions: when the HMM geometric distribution is replaced with an explicitly defined one, Baum-Welch and Viterbi algorithms are no longer directly applicable. In another approach to overcome the temporal limitations of the standard HMM framework, alternative trajectory modeling [4] has been proposed, taking advantage of frame correlation. The models obtained can improve speech recognition performance, but they require a demoralizing increase in model parameters and computational complexity. A smooth speech trajectory can be generated by HMMs through maximization of the model's output probability under the constraints between static and dynamic feature, modeling the temporal evolution of the acoustic models [5].

Therefore, a natural next step, given this previous research, is to work on a framework for dealing with temporal and parameter dependencies while still working with

regular HMMs, which can be done by using augmented HMMs. Augmented statistical models have been proposed previously as a systematic technique for modeling additional dependencies in HMMs, allowing the representation of highly complex distributions. Additional dependencies are thus incorporated in a systematic fashion. However, the price for flexibility is high, even when working with more computationally-friendly purposes [2].

The approach presented in this chapter consists of creating an augmented set of models, modeling temporal and inter-parameter dependence.

3. N-GRAM MODELING

To better analyze the influence of temporal and parameter dependencies in recognition performance, both dependencies can be modeled in an independent fashion. Thus, a new set of acoustic models will be built for each case without losing the scope of regular HMMs. For both cases, the most frequent combinations of features from the MFCC-based parameterized signal will be selected following either temporal or parameter dependence criteria. Language modeling techniques should be used for performing this selection. In this way, a new probability space can be defined, to which the input signal will be mapped, defining a new set of features.

In standard semi-continuous HMMs (SCHMMs), the density function $b_i(x_t)$ for the output of a feature vector x_t by state i at time t is computed as a sum over all codebook classes $m \in M$ (see [1]):

$$b_i(x_t) = \sum_m c_{i,m} \cdot p(x_t|m,i) \approx \sum_m c_{i,m} \cdot p(x_t|m) \quad (1)$$

Now new weights should be estimated as there are more features (inter-parameter dependencies or temporal dependencies) to cover the new probability space. Also, the posterior probabilities $p(x_t|m)$ will be modified as some independencies will no longer apply.

From this new set of features, regular SCHMM-based training will be performed, leading to a new set of augmented statistical models.

3.1. Modelling inter-parameter dependence

Let us assume that we work with four MFCC features: cepstrum (f_0), its first and second derivatives (f_1,f_2) and the first derivative of the energy (f_3). We can express the joint output probability of these four features applying Bayes' rule:

$$P(f_0,f_1,f_2,f_3) = P(f_0)P(f_1|f_0)P(f_2|f_1,f_0)P(f_3|f_2,f_1,f_0) \quad (2)$$

where f_i corresponds to each of the acoustic features used to characterize the speech signal.

Assuming parameter independence, HMM theory expresses equation 2 as:

$$P(f_0,f_1,f_2,f_3) = P(f_0)P(f_1)P(f_2)P(f_3) \quad (3)$$

To overcome parameter independence, some middle ground has to be found between equations 2 and 3. Thus, instead of using all dependencies to express the joint output probability, only the most relevant dependence relations between features are kept. For the spectral features, we take into account the implicit temporal relations between features. For the energy, experimental results show in a more relevant dependence on the first spectral derivative than to the rest. Thus, equation 2 is expressed as:

$$P(f_0,f_1,f_2,f_3) = P(f_0)P(f_1|f_0)P(f_2|f_1,f_0)P(f_3|f_1)$$

In practice, not all the combinations of parameters will be used for modeling each parameter dependence for each $P(f_i)$, but only the most frequent ones. Taking into account the parameter dependence restrictions proposed, a basic N-gram analysis of the dependences in the training corpus is performed, defining those most frequent combinations of acoustic parameterization labels for each spectral feature. That is, we will consider dependence between the most frequent parameter combinations for each feature (considering 3-grams and 2-grams), and assume independence for the rest.

The input signal will be mapped to the new probability space. Recalling equation 1, we can redefine the output probability of state i at time t for each of the features used as $P_i(f_k)$, where f_k corresponds to each of the acoustic feature used to characterize the speech signal. Then, the new output probability is defined as a sum over all codebook classes $m \in M$ of the new posterior probability function weighted by the new weights (taking advantage of 2-grams and 3-grams):

$$\begin{aligned} P_i(f_0) &= \sum_m c^0_{i,m} \cdot p(f_0|m) \\ P_i(f_1) &= \sum_m c^1_{i,m,\hat{m}_0} \cdot p(f_1|m) \\ P_i(f_2) &= \sum_m c^2_{i,m,\hat{m}_0,\hat{m}_1} \cdot p(f_2|m) \\ P_i(f_3) &= \sum_m c^3_{i,m,\hat{m}_1} \cdot p(f_3|m) \end{aligned}$$

where $$\hat{m}_k = \operatorname*{argmax}_m p(f_k|m)$$

is the likeliest class for parameter f_k at state i and time t. The new weights are defined according to N-gram based feature combinations:

- $c^1_{i,m,j} = c^1_{i,m}$ if the 2-gram "j,m" is not defined
- $c^2_{i,m,j,k} = c^2_{i,m,j}$ when the 3-gram "k,j,m" is not defined, but it is defined the 2-gram "j,m", and $c^2_{i,m,j,k} = c^2_{i,m}$ when neither the 3-gram not the 2-gram are defined
- $c^3_{i,m,j} = c^3_{i,m}$ when the 2-gram "j,m" is not defined

From these new output probabilities, a new set of HMMs can be obtained, using a Baum-Welch training, and used for decoding following the traditional scheme.

3.2. Modelling temporal dependencies

Next, we study the Markov assumption for the Markov chain. It is generally expressed as:

$$P(s_t|s_1^{t-1}) = P(s_t|s_{t-1}) \quad (4)$$

where s_1^{t-1} represents the state sequence $s_1, s_2, ..., s_{t-1}$.

Considering temporal dependences, equation 4 should be reformulated. But, for simplicity, not all of the sequence of observations is taken into account, but only the two previous ones for each observation s_t, working with the 3-gram s_{t-2}, s_{t-1}, s_t. Then, equation 4 can be expressed as:

$$P(s_t|s_1^{t-1}) = P(s_t|s_{t-2}, s_{t-1})$$

Applying independence among features (recall equation 3), the output proability of each HMM feature will be expressed as:

$$P(f_i) = P(f_i|f_{i_{t-2}}, f_{i_{t-1}})$$

Again, the most frequent combinations of acoustic parametrization labels can be defined, and a set of augmented acoustic models can be trained. The output probability (from equation 1) of state i at time t for each feature k will be rewritten as:

$$P_i(f_k) = \sum_m c^k_{i,m,\hat{m}_{k,t-1},\hat{m}_{k,t-2}} \cdot p(f_k|m) \qquad (5)$$

with $$\hat{m}_{k,t-i} = \underset{m}{\operatorname{argmax}}\, p(f_k|m, t-i)$$

Notice that if the 3-gram "$\hat{m}_{k,t-2}, \hat{m}_{k,t-1}, m$" does not exist, the 2-gram or 1-gram case will be used.

4. EXPERIMENTS AND RESULTS

4.1. Methods and tools

For the experiments performed to test these approaches, the semi-continuous [6] HMM-based speech recognition system RAMSES [7] was used as reference ASR scheme, and it is also used in this chapter as baseline for comparison purposes.

When working with connected digit recognition, 40 semidigit models were trained for the first set of acoustic models, with the addition of one noisy model for each digit, each modeled with 10 states. Silence and filler models were also used, each modeled with 8 states. When working with continuous speech recognition, demiphones models were used. For the first set of acoustic models, each phonetic unit was modeled by several 4-state left-to-right models, each of them modeling different contexts. In the augmented set of HMMs, each phonetic unit was modeled by several models that modeled different temporal dependencies, also using 4-state left-to-right models.

Connected digits recognition was used as the first working task for testing speech recognition performance, as it is still a useful practical application. Next, a restricted large vocabulary task was tested in order to evaluate the utility of the approach for today's commercial systems. Different databases were used: the Spanish corpus of the SpeechDat and SpeechDatII projects and an independent database obtained from a real telephone voice recognition application, known as DigitVox, were used for the experiments related to connected digits recognition. The Spanish Parliament dataset (PARL) of the TC-STAR project[1] was used for testing the performance of the models for continuous speech recognition.

[1]TC-STAR: Technology and corpora for speech to speech translation, www.tc-star.org

4.2. Results modeling parameter dependencies

In the first set of experiments we modeled parameter dependencies. The different configurations used are defined by the number of N-grams used for modeling the dependencies between parameters for each new feature. In the present case, no dependencies are considered for the cepstral feature, 2-grams are considered for the first cepstral derivative and for the energy, and 2 and 3-grams for the second cepstral derivative. As explained in section 3, as we cannot estimate all the theoretical acoustic parameter combinations, we define those **N** most frequent combinations of parameterization labels for each spectral feature. A low **N** means that only some combinations were modeled, maintaining a low dimension signal space for quantization. On the other hand, increasing **N** more dependencies will be modeled at the risk of working with an excessive number of centroids to map the speech signal.

Different configurations were tested. Each configuration is represented by a 4-digit string with the different values of **N** used for each feature. The total number of codewords to represent each feature is the original acoustic codebook dimension corresponding to this feature plus the number of N-grams used. The different combinations that result in the configurations chosen were selected after several series of experiments, defined to either optimize recognition results or to simplify the number of N-grams used.

database	configuration	SRR	WER
SpeechDat	baseline	90.51	2.65
	−/2000/2000,2000/2000	91.04	2.52
DigitVox	baseline	93.30	1.27
	−/2000/2000,2000/2000	93.71	1.17

Tabla 1. *Connected digit recognition rates modeling inter-parameter dependencies*

In table 1 we present the best results obtained for connected digit recognition experiments. Results are expressed according to SRR (Sentence Recognition Rate) and WER (Word Error Rate) to measure the performance. We can see an important improvement in speech recognition for this task using the SpeechDat dataset, with a relative WER decrease of nearly a 5%. When using the DigitVox dataset this improvement is slightly higher, with a relative WER decrease of 7.8%. Because both datasets are independent from the training datasets, we didn't expect adaptation of the solution to the training corpus.

4.3. Results modeling temporal dependencies

When modeling temporal dependencies, each new HMM feature models the dependencies of the original acoustic features. Again, the different configurations are represented by a 4-digit string with the number of N-grams

used in equation 5 for modeling each acoustic parameter. In contrast to inter-parameter dependence modeling, a wider range of **N** leads to an increase in recognition accuracy. Thus, this is a more flexible solution, where we can chose between optimizing the accuracy and working with reasonable codebook size (close to the state-of-the art codebooks when working with standard implementations) while still improving the recognition performance.

A first set of experiments using connected digit recognition was used to analyze the evolution of recognition performance regarding **N**, and also to study the differences in performance when testing the system with the SpeechDat database or an independent database (DigitVox). Results obtained with the SpeechDat dataset show that by modeling time dependencies, we can achieve a great improvement in recognition, outperforming the inter-parameter dependencies modeling approach with a relative WER reduction of around 26% compared to baseline results. However, the improvement when using the DigitVox dataset was slightly lower, with a relative WER reduction of 10.2%. Thus, this solution seems more likely to be adapted to the training corpus for connected digit recognition.

To test whether time dependencies modeling works better using a bigger (and wider) training corpus, continuous speech recognition was used, with new sets of acoustic models based on demiphones, using the PARL dataset. The results, presented in table 2 show a WER reduction between 14.2% and 24.3%. We observe some saturation in WER improvement when **N** is increased over certain values: after reaching optimum values, WER improvement becomes slower, and we should evaluate if the extra improvements really do justify the computational cost of working with such large values of **N** (which means working with high codebook sizes). Afterwards, additional WER improvement tends to zero, so no extra benefit is obtained by working with a very high number of N-grams. Thus a compromise between the increase in codebook size and the improvement in recognition accuracy is made when deciding upon the best configuration.

configuration	**WER**	**WER**$_{var}$
baseline	28.62	-
3240/2939/2132/6015	24.56	14.19%
7395/6089/4341/8784	27.73	24.07%
20967/18495/17055/15074	21.66	24.32%

Tabla 2. *Continuous speech recognition rates modeling time dependencies with TC-Star database*

5. CONCLUSIONS

In this paper we present two approaches for using N-gram based augmented HMMs. The first solution consists of modeling the dependence between the different acoustic parameters, thus overcoming the parameter independence assumption. The second approach relies on modeling the temporal evolution of the regular frequency-based features, trying to break the time independence assumption.

Experiments on connected digit recognition and continuous speech recognition have been performed. The results presented show an improvement in recognition accuracy especially for the time dependencies modeling based proposal. Therefore, it seems that time-independence is a restriction for an accurate ASR system. Also, temporal evolution seems to need to be modeled in a more detailed way than the mere use of the spectral parameter's derivatives.

A more relevant improvement is achieved for continuous speech recognition than for connected digit recognition. For both tasks, independent testing datasets were used in last instance. Hence, this improvement does not seem to be related to an adaptation of the solution to the training corpus, but to better modeling of the dependencies for demiphone-based models. Thus, more general augmented models were obtained when using demiphones as HMM acoustic models.

Further work will be needed to extend this method to more complex units and tasks, i.e. using other state-of-the-art acoustic units and addressing very large vocabulary ASR or even unrestricted vocabulary tasks.

6. BIBLIOGRAPHY

[1] X. Huang, A. Acero and H.W., *Spoken Language Processing*, Prentice Hall PTR, 1st edition, 2001.

[2] M.T. Layton and M.J.F. Gales, "Augmented statistical models for speech recognition," *Proceedings of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2006.

[3] J. Pylkkönen and M. Kurimo, "Duration modeling techniques for continuous speech recognition," *Proceedings of European Conf. on Speech Technology (EUROSPEECH)*, 2003.

[4] S. Takahashi, "Phoneme HMMs constrained by frame correlations," *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 1993.

[5] M. Casar and J.A.R. Fonollosa, "Analysis of hmm temporal evolution for automatic speech recognition and utterance verification," *Proc. of IEEE Int. Conf. on Spoken Language Processing (ICSLP)*, 2006.

[6] X.D. Huang and M.A. Jack, "Unified techniques for vector quantisation and hidden markov modeling using semi-continuous models," *Proceedings of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 1989.

[7] A. Bonafonte et alter, "Ramses: el sistema de reconocimiento del habla continua y gran vocabulario desarrollado por la UPC," *VIII Jornadas de Telecom I+D*, 1998.

SUPPORT VECTOR REGRESSION IN NIST SRE 2008 MULTICHANNEL CORE TASK

Ismael Mateos, Daniel Ramos, Ignacio Lopez-Moreno and Joaquin Gonzalez-Rodriguez[1]

ATVS (Biometric Recognition Group), C/ Francisco Tomas y Valiente 11,
Universidad Autonoma de Madrid, E28049 Madrid, Spain
{ismael.mateos, daniel.ramos, ignacio.lopez, joaquin.gonzalez}@uam.es

ABSTRACT

This paper explores two alternatives for speaker verification using Generalized Linear Discriminant Sequence (GLDS) kernel: classical Support Vector Classification (SVC), and Support Vector Regression (SVR), recently proposed by the authors as a more robust approach for telephone speech. In this work we address a more challenging environment, the NIST SRE 2008 multichannel core task, where strong mismatch is introduced by the use of different microphones and recordings from interviews. Channel compensation based in Nuisance Attribute Projection (NAP) has also been investigated in order to analyze its impact for both approaches. Experiments show that, although both techniques show a significant improvement over SVC-GLDS when NAP is used, SVR is also robust to channel mismatch even when channel compensation is not used. This avoids the need of a considerable set of training data adapted to the operational scenario, whose availability is not frequent in general. Results show a similar performance for SVR-GLDS without NAP and SVC-GLDS with NAP. Moreover, SVR-GLDS results are promising, since other configurations and methods for channel compensation can further improve performance.

Index Terms: speaker verification, GLDS, SVM classification, SVM regression, inter-session variability compensation, robustness.

1. INTRODUCTION

Speaker verification aims at determining whether a given speech material of unknown source belongs to a claimed individual's identity or not. The state-of-the-art in speaker verification has been dominated in the last years by systems working at the spectral level. Techniques like Gaussian Mixture Models (GMM) [1], Support Vector Machines (SVM) [2, 3], or hybrid approaches such as GMM-SVM [3] have demonstrated higher performance for this task.

Among this kind of systems working at the spectral level, SVM Classification (SVC) using Generalized Linear Discriminant Sequence (GLDS) kernel has been used in the past [2]. This technique first maps the parameter vectors extracted from the speech to a high-dimensional space via a GLDS kernel function, where a SVM classifier is used to classify such speech as belonging to the claimed identity or to an impostor. Thus, this task is essentially a binary classification problem.

Another essential factor for the improvement of state-of-the-art performance of the technology in the last years has been the use of session variability compensation schemes. Techniques like Factor Analysis [4] or Nuisance Attribute Projection (NAP) [5] have been critical for the robustness of systems under variation in the conditions of the speech. However, their ability to reduce inter-session variability effects is conditioned to the availability and correct use of appropriate databases similar to the data that the system will face in operational conditions. Such databases may be hard to obtain in many applications.

During the last years there have not been significant improvements in SVC-GLDS, so its performance is lower than other approaches at the spectral level such as GMM or GMM-SVM. Nevertheless, Support Vector Regression (SVR) has been recently proposed, showing a significant performance improvement over classical SVC for GLDS kernel in a telephone scenario [6]. Thus, it is necessary to study the performance of SVR-GLDS when facing more challenging environments in terms of session variability.

In this paper we show experiments illustrating the robustness of the SVR-GLDS approach under strong session variability. For this purpose we have used the NIST SRE 2008 evaluation protocol where speech from different microphones and telephone networks is present with different languages and speaking styles. This paper is organized as follows, SVM classification and regression is introduced in Section 2. Section 3 presents the proposed SVR-GLDS system. In Section 4, experiments showing the performance of SVC-GLDS and SVR-GLDS with and without session variability compensation are presented. Finally, conclusions are drawn in Section 5.

2. SUPPORT VECTOR MACHINE CLASSIFICATION AND REGRESSION

SVM have been largely used for a wide range of different pattern recognition and machine learning tasks. One of the main advantages of this technique is its good generalization capabilities to unseen data. This fact joined to its computational efficiency establish SVM as a good candidate for tasks like speaker recognition, as has been demonstrated in [2, 3].

The approach for speaker verification using SVM in the past has been mainly based on SVC, where the classes are defined as *the claimed speaker being the author of the test speech segment of unknown origin* (target speaker hypothesis) or *another individual being the author* (non-target speaker hypothesis). Recently, the authors proposed the use of SVR, a more general approach.

[1] This work has been supported by the Spanish Ministry of Education under project TEC2006-13170-C02-01.

2.1. Support Vector Machine Classification (SVC)

The goal of classification using support vector machines consists in finding and optimal decision hyperplane, represented by its normal vector *w*. This is performed in a so-called expanded feature space, where the MFCC *(Mel Frequency Cepstral Coefficients)* feature vectors are mapped in order to be easily separable [2]. The hyperplane *w* divides the high-dimensional space in two regions. In our particular speaker verification problem one of these regions will correspond to the target speaker hypothesis and the other to the non-target speaker hypothesis. The scoring function is then defined as the distance of each vector to the hyperplane:

$$f(x_i) = \langle w, x_i \rangle + b \tag{1}$$

where *b* is a learned offset parameter. To explain the SVM algorithm in more detail let consider the linearly separable case. Suppose we have a data set labelled $D = \{(x_1, y_1), (x_2, y_2) \cdots (x_l, y_l)\}$ where x_i represent the vector and y_i the label. For example $y_i = 1$ if x_i belong to the target speaker and $y_i = -1$ in the rest of cases. The objective hyperplane in this case will be the one that maximize the margin between classes:

$$w = \min\left(\frac{1}{2} w^T \cdot w\right) \quad \text{subject to: } y_i f(x_i) - 1 \geq 0 \tag{2}$$

Unfortunately, in real applications there are many effects, e.g. noise, channel effects, intra- and inter- class variability, etc., which can cause the restriction in (2) to be violated. In such case, the problem will not be linearly separable. This new problem can be solved by considering two different criteria for finding *w*: *i)* maximising the margin between classes and *ii)* minimising a loss function proportional to misclassified vectors. A weighting factor *C* controls the relevance of one criterion against the other.

$$w = \min\left(\frac{1}{2} w^T \cdot w + C \frac{1}{m} \sum \xi_{c,i}\right) \quad \text{subject to: } 0 \leq \xi_{c,i} \leq 1 - y_i f(x_i) \tag{3}$$

$\xi_{c,i}$ is a penalty associated to the vectors that do not satisfy the restriction in (2). Thus, for classification problems the loss function is defined as:

$$f_{loss}(x_i) = \max\{0, 1 - y_i \cdot f(x_i)\} \tag{4}$$

If a non-linear classification boundary is desired, an elegant method consists in mapping each vector to a higher-dimension feature space. For this purpose a map function, $\phi(x_i)$, is used. It can be demonstrated that we can obtain a transformation $\phi(x_i)$ where the vectors are linearly separable. Furthermore, the SVM algorithm only requires inner products of the vectors in the expanded space, $\langle \phi(x_i), \phi(x_j) \rangle$, where the *kernel* function is defined as:

$$k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{5}$$

The possibility of computing the inner products without explicitly mapping each vector into the high dimensional space is known as *kernel trick.*

2.2. Support Vector Machine Regression (SVR)

As shown before, the objective of SVC was to find an optimal hyperplane witch separates the target and nontarget data. In the SVR case the goal is more general: learning a n-dimensional function based on the data.

The vector labels, y_i, are seen as a function of x_i, $g_n(x_i) = y_i$. SVR will try to find a function $f(\cdot) \simeq g_n(\cdot)$. The degree of approximation to the function $g_n(\cdot)$ is controlled through the parameter *C*.

The main difference between SVC and SVR is the loss function. SVC penalizes the situation where $f(\cdot) < g_n(\cdot)$, but as SVR aims at estimating a function, it also penalizes $f(\cdot) > g_n(\cdot)$. The loss function should consider such effect, and there are different options in the literature. A popular choice is the ε-insensitive loss function [7], where vectors are penalized when $|f(\cdot) - g_n(\cdot)| > \varepsilon$. The objective hyperplane in the SVR case will then be:

$$w = \min\left(\frac{1}{2} w^T \cdot w + C \frac{1}{m} \sum \xi_{c,i} + \xi'_{c,i}\right) \quad \text{subject to: } \begin{cases} 0 \leq f(x_i) - y_i \leq \xi_{c,i} + \varepsilon \\ 0 \leq y_i - f(x_i) \leq \xi'_{c,i} + \varepsilon \end{cases} \tag{6}$$

If we compare these criteria with SVC in Equation (3), we observe some differences. We have the SVC penalty variable, $\xi_{c,i}$, for those vectors for which $f(x_i) > g_n(x_i) + \varepsilon$, and a new variable $\xi'_{c,i}$ for those ones for which $f(x_i) < g_n(x_i) - \varepsilon$. The loss function is then defined as:

$$f'_{loss}(x_i) = \max\{0, |y_i \cdot f(x_i)| - \varepsilon\} \tag{7}$$

The differences between $f'_{loss}(x_i)$ (SVR) and $f_{loss}(x_i)$ (SVC) are shown in Figure 1. The loss functions are centered at $f(x_i) = y_i$ for SVC and at $f(x_i) = g_n(x_i)$ for SVR.

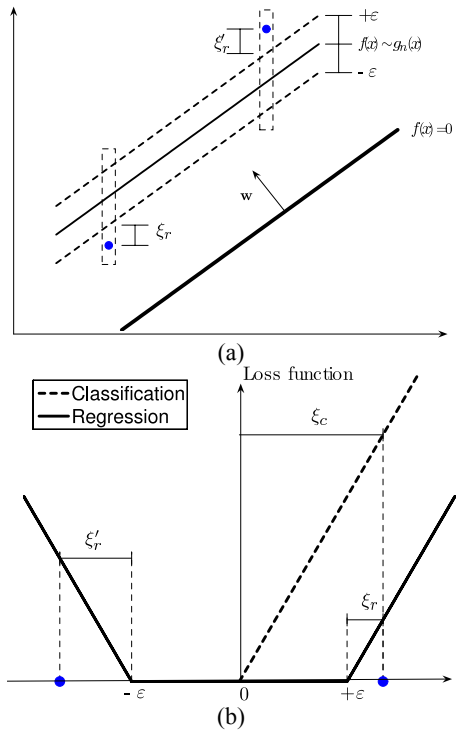


Figure 1. *SVR vs. SVC. Boundaries (a) and loss functions (b).*

3. SVR-GLDS SPEAKER VERIFICATION

We propose to use SVR with a ε-insensitive loss function for the speaker verification task. Recently, the authors showed the performance of this novel approach over the core task of NIST SRE 2006 [6], a telephone scenario, obtaining good results in comparison with SVC.

One of the main consequences of using the SVR approach in the GLDS space relates to the use of support vectors for SVM training. SVC uses support vectors which are near the frontier between classes, where the vectors use to be scarce. SVR selects support vectors from areas where there is a higher concentration of vectors. Thus, the SVC hyperplane may be more sensitive than SVR to outliers, noisy vectors, etc. In this sense, SVR can present a more robust performance than SVC against outlier support vectors due to extreme conditions in some speech utterances.

Another advantage of the SVR approach relies on the use of the ε parameter. There are some works in the literature [8] that relate the ε parameter to the noise or variability of the function estimate. Following such assumptions, tuning ε allows us to adapt the SVR training process to the variability in the expanded feature space.

4. EXPERIMENTS

4.1. SVM-GLDS systems

Both ATVS SVC-GLDS and SVR-GLDS systems are based on a GLDS kernel as described in [2]. Feature extraction is performed based on audio files processed with Wiener filtering (an implementation is available at http://www.icsi.berkeley.edu/ftp/global/pub/speech/papers/qio). 19 MFCC plus deltas are then extracted. In order to avoid channel mismatch effects, CMN *(Cepstral Mean Normalization)*, RASTA filtering and feature warping are performed. A GLDS kernel expansion is performed on the whole observation sequence, and a separating hyperplane is computed between the training speaker features and the background model. The system uses a polynomial expansion of degree three prior to the application of the GLDS kernel.

In order to face the problem of session variability, speaker vectors obtained after calculating the expanded feature vector, were channel compensated. The compensation was performed by projecting out its expanded values into a trained channel subspace, which is known as NAP [5]. The score computation is based on the distance of the expanded features to the separating hyperplane, as shown in (1). Finally, the T-Norm [9] score normalization technique is applied. We have used the LibSVM library (http://www.csie.ntu.edu.tw/~cjlin/libsvm) for training both SVM classification and regression algorithms.

4.2. Databases and experimental protocol

Experiments have been performed using the NIST Speaker Recognition Evaluation (SRE) 2008 [10]. These evaluations are the main forum for the improvement of the technology performance of speaker recognition systems. Each new NIST evaluation involves a new challenge to the science community, contributing to increase the efforts and research works in the speaker verification field, and fostering common testing and comparison protocols.

The main difference of NIST SRE 2008 with previous evaluations consists in including in the training and test conditions for the core task not only conversational telephone speech data but also conversational speech data recorded over a microphone channel involving an interview scenario, and additionally, for the test condition, conversational telephone speech recorded over a microphone channel. The evaluation protocol defines the following training conditions: 10 seconds, 1 *(short2)*, 3 and 8 conversation sides and long conversation; and the following test condition: 10 seconds, 1 (*short3)* conversation side and long conversation. Each "short" conversation, either recorded over a telephone or a microphone, has an average duration of 5 minutes, with 2.5 minutes of speech on average after silence removal. Interview segments contain about 3 minutes of conversational speech recorded by a microphone, most of the speech generally spoken by the target speaker. Although there are speakers of both genders in the corpus, no cross-gender trials are defined. In our case the experiments followed the core task, namely short2 training conditions, and short3 test condition (short2-short3).

Taking into account the test and train channel types, the evaluation protocol can be divided in 4 conditions: *tlf-tlf* (37050 trials), *tlf-mic* (15771 trials), *mic-mic* (34046 trials) and *mic-tlf* (11741 trials). NIST made available for the participants the type of channel (microphone or telephone) for each speech segment.

The background set for system tuning is a subset of databases from previous NIST SRE, including telephone and microphone channels. The T-Norm cohorts were extracted from the NIST SRE 2005 target models, 100 telephone models and 240 microphone models. NAP channel compensation was trained using recordings belonging to NIST SRE 2005 speakers which are present in both telephone and microphone data.

4.3. Results

The performance of SVC-GLDS is first evaluated with two different configurations: *i)* without including any compensation technique, and *ii)* including a NAP compensation scheme. Table 1 shows the performance of the system detailed per condition. Results are presented both as Equal Error Rate *(EER)* and DCF_{min} as defined by NIST [10]. It is observed that the performance of the system significantly improves when NAP is added to the system, both for EER and DCF values. The improvement is bigger when strong channel mismatch occurs (*tlf-mic* or *mic-tlf* conditions).

	SVC-GLDS		SVC-GLDS + NAP	
	EER (%)	DCF_{min}	EER (%)	DCF_{min}
tlf-tlf	13.8	0.054	**10.2**	**0.047**
tlf-mic	24.1	0.075	**13.9**	**0.053**
mic-mic	17.4	0.075	**13.0**	**0.057**
mic-tlf	23.5	0.078	**15.3**	**0.059**

Table 1. *EER and DCF_{min} in NIST SRE 2008 short2-short3, for SVC-GLDS.*

In order to use the proposed SVR-GLDS system, tuning the ε parameter is firstly required, and the variation of its performance with respect to such parameter is presented here. As we saw in [6] the system performance significantly changes as a function of this parameter. In that case $\varepsilon = 0.1$ was the optimal value. Tables 2 and 3 show the performance for different values of ε.

	0.05	0.1	0.2	0.4	0.8
tlf-tlf	**9.9**	10.0	10.9	13.5	13.9
tlf-mic	16.9	**15.1**	16.6	23.8	24.0
mic-mic	15.7	**15.4**	15.9	16.8	17.4
mic-tlf	17.0	**16.4**	18.8	22.8	23.6

Table 2. *EER in NIST SRE 2008 sort2-sort3, for different values of ε in SVR-GLDS.*

	0.05	0.1	0.2	0.4	0.8
tlf-tlf	0.046	**0.045**	0.047	0.052	0.054
tlf-mic	0.059	**0.055**	0.063	0.074	0.075
mic-mic	**0.064**	0.065	0.067	0.074	0.075
mic-tlf	**0.063**	0.064	0.066	0.078	0.078

Table 3. *DCF_{min} in NIST SRE 2008 sort2-sort3, for different values of ε in SVR-GLDS.*

In most part of the cases $\varepsilon = 0.1$ significantly improves the system performance. Thus we just have to tune the system one time and not for each one of the four conditions. For the rest of experiments $\varepsilon = 0.1$ will be used for SVR.

Finally, we have evaluated the performance of SVR-GLDS + NAP versus the systems mentioned above: SVC-GLDS, SVC-GLDS + NAP and SVR-GLDS. Table 4 shows the comparison in EER and DCF values for each condition and Figure 2 shows the global DET curves of the systems.

		tlf-tlf	tlf-mic	mic-mic	mic-tlf
SVC	EER	13.8	24.1	17.4	23.5
	DCF_{min}	0.054	0.075	0.075	0.078
SVC+ NAP	EER	10.2	**13.9**	**13.0**	15.3
	DCF_{min}	0.047	**0.053**	**0.057**	**0.059**
SVR	EER	10.0	15.1	15.4	16.4
	DCF_{min}	0.045	0.055	0.065	0.064
SVR+ NAP	EER	**9.6**	14.3	13.8	**15.0**
	DCF_{min}	**0.045**	0.053	0.060	0.062

Table 4. *EER (%) and DCF_{min}. performance of SVC, SVC + NAP, SVR and SVR + NAP systems in NIST SRE 2008 short2-short3 task.*

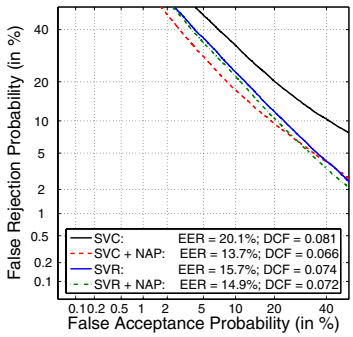


Figure 2. *DET curve of SVC, SVC + NAP, SVR and SVR + NAP systems in NIST SRE 2008 short2-short3 task.*

The system with the best performance is SVC-GLDS + NAP, obtaining a relative improvement in EER of 31% and 19% in DCF value. The proposed system, SVR-GLDS, presents a similar performance before and after channel compensation. This has the advantage that there is no need of using NAP to obtain similar performance as SVC-GLDS + NAP. If a suitable database is available, NAP may significantly improve the performance of the system, but if such database is not available or the representative data is scarce, SVR-GLDS seems a convenient option for obtaining robustness. The latter may be the case in many applications.

Moreover, SVR-GLDS + NAP provides a slight improvement, in both EER and DCF values, with respect to SVR-GLDS. This result is promising, as no special tuning of the ε parameter has been performed. As the NAP transformation changes the properties of the expanded space, a finer determination of ε may possibly lead to a further increase in performance.

5. CONCLUSIONS

In this paper we have explored the performance of SVR-GLDS for speaker verification, recently proposed by the authors, over the NIST SRE 2008 core multichannel task. This technique is a more general and robust approach than the widely-used SVC-GLDS. Results show that the performance of the SVR-GLDS approach without channel compensation is comparable to SVC-GLDS with NAP. Therefore, if a suitable database is available, NAP may significantly improve the performance of the system, but if such database is not available or the representative data is scarce, SVR-GLDS seems a more convenient option for obtaining robustness. Moreover, since channel compensation requires a sizeable amount of data, in many real applications SVR may seem an attractive option for robustness. Furthermore, it is possible to combine SVR-GLDS with channel compensation, which further improves SVR-GLDS performance showing promising results.

Future work includes the use of different SVR approaches for the GLDS space, such as μ-SVR, non-linear loss functions and different kernels. Also, the combination of SVR with NAP, tuning the ε parameter and its effects on the system performance will be investigated in depth.

6. REFERENCES

[1] D. A. Reynolds, et al., "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19-41, 2000.

[2] W. M. Campbell, et al., "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, no. 2-3, pp. 210-229, 2006.

[3] W. M. Campbell, et al., "Support vector machines using GMM supervectors for speaker verification," *Signal Processing Letters*, vol. 13(5), pp. 308-311, 2006.

[4] P. Kenny and P. Dumouchel, "Disentangling speaker and channel effects in speaker verification," in *Proc. Of ICASSP*, vol. 1, pp. 37-40, 2004.

[5] A. Solomonoff, et al., "Advances in channel compensation for SVM speaker recognition," in *Proc. Of ICASSP*, pp. 629-632, 2005.

[6] I. Lopez-Moreno, et al., "Support Vector Regression for Speaker Verification", in *Proc. Of Interspeech,* pp. 306-309, Antwerp, Belgium, 2007.

[7] K. Muller, et al., "Predicting time series with support vector machines," in *Proc. of the 7th International Conference on Artificial Neural Networks*, vol. 1327 of *Lecture Notes In Computer Science*, pp. 999-1004, 1997.

[8] A. J. Smola and B. Schoelkopf, "A tutorial on support vector regression," Tech. Rep. NeuroCOLT2 Technical Report NC2-TR-1998-030, Royal Holloway College, University of London, UK, 1998.

[9] R. Auckenthaler, et al., "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42-54, 2000.

[10] NIST, "2008 speaker recognition evaluation plan: http://www.nist.gov/speech/tests/sre/2008/index.html," 2008.

TRAINING A ROBUST COMMAND RECOGNIZER WITH THE TECNOVOZ DATABASE

José Lopes[1,3], Cláudio Neves[1], Arlindo Veiga[1], Alexandre Maciel[1], Carla Lopes[1], Luís Sá[1,2], Fernando Perdigão[1,2]

[1] Instituto de Telecomunicações – Pólo de Coimbra, 3030-290 Coimbra, Portugal
[2] Dep. Eng. Electrotécnica e de Computadores, FCTUC, 3030-290 Coimbra, Portugal
[3] L2F – Spoken Language Systems Lab INESC-ID, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal

ABSTRACT

This paper describes the development of a command-based robust speech recognition system for the Portuguese language. Due to an efficient noise reduction algorithm the system can be operated in adverse noise environments such as in vehicles or factories. The acquisition of a Portuguese database in the scope on the Tecnovoz project is addressed in this paper. The paper also describes a new noise-robust front-end and some experiments regarding the best acoustic model to use for a command-based speech recognizer. Results with whole-word, monophone and triphone models are presented and discussed.

1. INTRODUCTION

Tecnovoz [1] is a shared-cost project funded by the Portuguese government which aims to create a body of knowledge on voice technologies, particularly to the Portuguese language. This will materialize in a series of products for the market. The authors were responsible in the framework of the project for the development of a speech independent connected word recognizer which operates under noise adverse conditions, such as factories and vehicles. Therefore it has to incorporate advanced noise reduction techniques. Finally, the recognizer has to be computationally efficient in order to operate on small footprint embedded hardware platforms.

The speech database was collected in the scope of the Tecnovoz project. It has been designed regarding typical application demands, in terms of vocabulary and acoustic environments. The acoustic models are based on Hidden Markov Models (HMMs).

In order to deal with noise adverse conditions, a noise reduction front-end was designed, based on the Advanced Front-End (AFE) ETSI Standard [2]. Some modifications were made from the standard to enhance the performance and speed of the speech recognizer.

In order to improve the robustness of the speech recognizer, several experiments with different acoustic models were carried out using either whole-word HMM models or smaller unit HMM models, such as monophones and triphones.

The paper is organized as follows. In section 2 the database is described. Section 3 describes the front-end implementation. Section 4 refers to the approaches to the acoustic modelling. Finally, in section 5, results obtained with different acoustic models and front-end configurations, are presented.

2. SPEECH DATABASE

Three acoustical environments were considered during the database acquisition, namely: clean (TVFL), vehicle (TVV) and factory (TVF) environments. The collected speech database includes about 250 commands and several phonetically rich sentences. About 30 minutes of spoken content were recorded by each of the 368 speakers, which turn into about 184 hours of speech content and a total of 232,000 files. Table 1 and Table 2 show the distribution of the database according to gender and file types, respectively.

Gender	TVFL	TVF	TVV
Female	103	20	9
Male	197	16	23

Table 1: *Gender distribution.*

Content	TVFL	TVF	TVV
Words	141,992	30,090	19,648
Sentences	40,458	–	384

Table 2*: Speech file distribution.*

3. FEATURE EXTRACTION

The feature extraction system is based on the AFE standard, which incorporates a two-stage Wiener filtering system. In this standard, the Wiener filter is estimated in the linear frequency domain and is implemented by a time domain convolution. Li et al, [3], proposed a new algorithm where both filter estimation

and operation are carried out in the Mel frequency domain. In our implementation some changes were made to Li et al approach in order to improve the front-end efficiency, as depicted in Figure 1 [4].

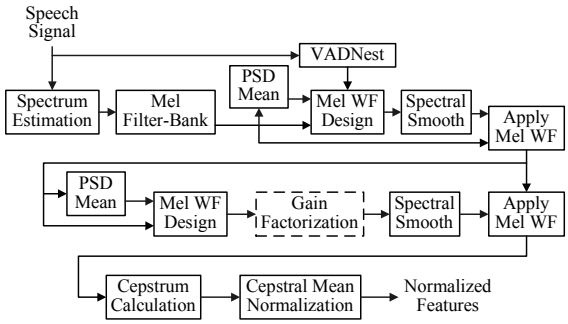


Figure 1: *Block diagram of the feature extraction system.*

It can be seen that the speech signal is processed by a two-stage Wiener filter as in the ETSI standard. The estimated signal spectrum is applied to a Mel filter-bank and the frames are then classified as "noise only" or "speech with noise" by the VADNest block. The Wiener filter design depends on this classification in order to estimate the noise spectrum. The "Spectral Smooth" block presents some modifications: the operations involved in the smoothing of the Wiener filter coefficients were reduced to a single matrix multiplication [4]. Apart from the gain factorization block, that were not found valuable for the final system performance, the second Wiener filter stage is similar to the one proposed in [3].

The de-noised frames are then converted to cepstral coefficients by a discrete cosine transform (DCT) and their means are normalized by a real-time algorithm, resulting on a feature vector with 39 components, comprising 12 cepstral coefficients plus log energy and their first and second time derivatives. The feature extraction algorithm is described in detail in [4].

4. MODEL TRAINING

Acoustic models were built using the Tecnovoz speech database and only files corresponding to command utterances with Signal-to-Noise Ratio above 15 dB were considered. There were a total of 137,860 files (120,459 from TVFL, 8,760 from TVF and 8,641 from TVV). From these files 75 % were picked for training, 20 % for test and 5 % for development. From the first trained models a recognition test was performed on the training database. The results allow us to detect transcription errors, and consequently, some annotation files had their marks re-adjusted, others were deleted and wrong labels were changed according to the word effectively pronounced. From these procedures the total number of files was reduced to 137,237 (119.975 from TVFL, 8,633 from TVF and 8,629 from TVV).

The model training was carried out using the HTK toolkit [5]. During the training three approaches were explored for the acoustic models: word-level, context-free phones and context-dependent triphones. The word-level approach tries to create HMM models for the whole-word, whereas context-free monophone models split the words into the corresponding monophone transcription to provide data for monophone training. Finally, triphone training tries to profit from left and right contexts of each phone, which naturally influence the acoustic realization of each phone, to create a new model. The advantages and disadvantages of each method will be discussed in the next sub-sections.

4.1. Word-level training

For word-level training, each of the 254 words is represented by an HMM with left-to-right topology. The number of states of the HMM depends on the word length in terms of phones. For example for the command "stop", the transcription is /s t O p/ (in SAMPA), which results in a 12-state HMM for this word, using 3 states for each phone.

The models "ruido" and "sil" are used to model noise and silence, respectively. They are represented with 3-state HMM's with left-to-right topology with an extra transition from the first to the last emitting states and vice-versa.

The model initialization was done with the HTK tools HInit and HRest. Afterwards, the training was carried out with the embedded re-estimation HTK tool HERest. Word-level models were trained with mixture increment, up to 10 Gaussian mixtures for each state.

4.2. Monophone training

The first step consisted in defining the phone set for the Portuguese language. A list of 40 phones was taken, including models for silence and pause. All phone HMMs have 3 states with a left-to-right topology and were initialized with the "flat start" method [5]. Multiple pronunciations were considered for some words, which permitted to realign the training data after 5 iterations of embedded re-estimation. The number of mixture components was then incremented up to 16 Gaussians, as depicted in Figure 2.

4.3. Triphone training

Triphones depends on the two adjacent phones, which gives considerable robustness to variations in pronunciations in specific contexts [6].

Since there is no annotated speech data at phone level, monophone models were used (initialized with the flat start procedure) to develop the intra-word triphone models. As triphones are phones with context, it was used a straightforward procedure to convert from one notation to another (e.g.: "dez" ("ten") → /d E S/ → /d+E d-E+S E-S/).

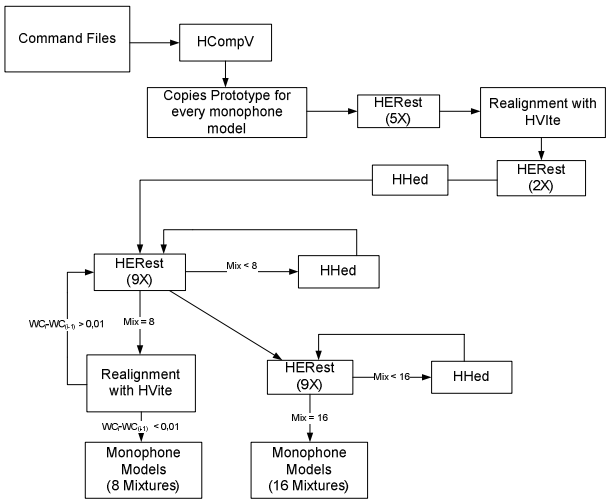


Figure 2: *Monophone training procedure.*

The resulting number of triphone models is 872 for the command vocabulary. This results in much less training material for each triphone compared with monophones. To overcome this problem and taking into consideration that there are many similar triphones in the model list, some models can be tied in order to reduce the total number of physical models. For this purpose two methods were considered: data-driven clustering (DDC) and tree-based clustering (TBC). The data-driven clustering uses a similarity measure between HMM states, while tree-based clustering builds a binary decision tree. This tree attempts to find those contexts which make the largest difference to the acoustics and which should therefore distinguish clusters. The latter method has the advantage of accommodating the construction of systems which have used unseen triphones [5]. Different likelihood thresholds in TBC and distance thresholds in DDC were taken into account as sources of variability in training. These thresholds have a strong influence on the number of physical models that need to be trained, and consequently in the total number of Gaussians, which is a major concern as recognizer will be working over low performance hardware. The number of Gaussians was incremented up to 16, as depicted in Figure 3.

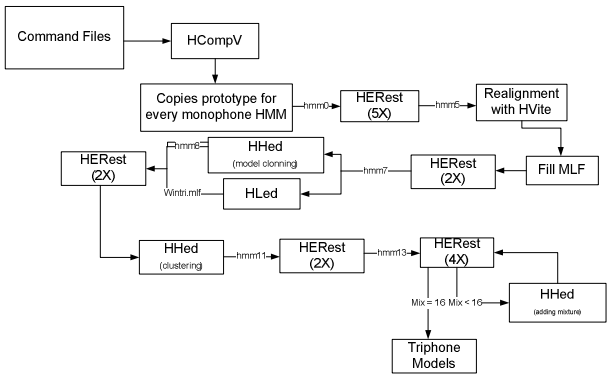


Figure 3: *Triphone training procedure.*

5. RESULTS

In this section, the results obtained with each acoustic modeling approach are presented. Tests were carried out using the HTK decoder tool HVite.

To perform the experiments, a task grammar must be defined in order to provide information about the sequence of events that can be found in the test utterances. The used grammar consisted in taking all the command words in parallel, with an optional silence before and after a command, as shown in Figure 4.

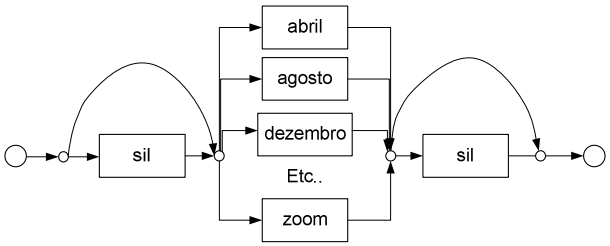


Figure 4: *Task grammar.*

Table 3 shows the achieved recognition rates of both the original and proposed front-ends in terms of the *Word Correctness* rate. The improvements made at the front-end level resulted in a system' performance improvement of about 2% (absolute points). This result suggests that the ETSI's AFE may be biased towards the database used for the evaluation of the algorithm (the Aurora 2 database).

Front-End	Word Correctness
Original ETSI AFE	94.88 %
Efficient Front-end (E-AFE)	96.88 %

Table 3: *Comparison between AFE and E-AFE.*

Three different versions of the whole-word models were tested. The first one corresponds to the models created with the first alignment of the training database (V1). The models' second version resulted from the new label files with re-adjusted marks (V2). The third one was created with a modification in the front-end, that consists in removing the gain factorization block, indicated in Figure 1 (V3). Results for 8 Gaussian mixtures are presented in Table 4. As expected, the consecutive modifications made on training procedure, label files and front-end improved the whole-word models.

Version	Word Correctness
V1	95.92 %
V2	96.55 %
V3	96.76 %

Table 4: *Whole-word models results.*

As referred to in section 4.2, the label files were automatically aligned several times, in order to improve robustness. Table 5 shows the word correctness for monophone models, with 8 Gaussian mixtures, for several realignment iterations. Besides the low rates obtained with the monophone models, an improvement of 6 % was observed by realigning the training data 3 times.

Number of Realignments	Word Correctness
0	83.46 %
1	87.57 %
2	87.54 %
3	89.28 %

Table 5: *Monophone models results.*

To evaluate triphone model performance, experiments were carried out with no clustering and with both clustering methods. Results obtained with 8 Gaussian mixture models are presented in Table 6.

Clustering Method	Threshold	Word Correctness
TBC	7500.0	96.06 %
TBC	1000.0	97.03 %
TBC	300.0	97.06 %
DDC	0.3	96.81 %
No clustering	–	97.03 %

Table 6: *Triphone models results.*

Results indicate that the lower the thresholds in TBC, the better are the results. This is due to the number of physical models resulting from the cluster which is higher when the likelihood threshold is lower. With about the same number of physical models, the DDC clustered models presents a slightly lower score. Nevertheless, the clustering method seems to be useless, since with no clustering a very similar recognition rate is achieved.

In order to compare the performance of the three acoustic model types, the best score from each approach is presented in the same table as well as the number of Gaussians that the ensemble of models have. According to Table 7, the triphone models have the best performance, comparing to whole-word or monophone models. As the recognizer should work on low performant hardware, a trade-off between computational load (dependent on the number of Gaussians), and the recognition rate should be made. The triphone models not only have less computational load when compared to whole-word models, as achieve higher recognition rate. As a result, most commands in the vocabulary are represented by triphone models in our recognition engine. Only smaller commands, where whole-word models seem to be more accurate, use this kind of models.

Acoustic Model	Word Correctness	Total Number of Gaussians
Whole-word	96.76 %	37,344
Monophone	89.28 %	952
Triphone	97.03 %	16,204

Table 7: *Comparison of acoustic models.*

6. CONCLUSIONS AND DISCUSSION

In this paper some modifications are proposed to the ETSI's AFE regarding noise robustness of a command-based speech recognition system for the Portuguese language. The new proposal outperformed the ETSI standard in about 2%.

Three different acoustic models (whole-word, monophone and triphone models) were also tested and compared. Results show that triphone models achieved the best performance.

Another interesting conclusion is that new word models can be easily built using the monophone models. The user just need to add the sequence of phones that compose a new command in order to be accepted by the recognizer engine. With triphones it is not that simple, because only a small set of triphones are available. An algorithm that associates to an unseen triphone the better one that is already on the initial triphone list is currently being developed. This tying takes into account acoustic and phonetic similarities between triphones. With this association the recognizer will be prepared to recognize any command.

7. REFERENCES

[1] Tecnovoz website (2007), http://www.tecnovoz.pt/web/home_english.asp.

[2] ETSI ES 202 050 v1.1.3, "Speech Processing Transmission and Quality Aspects (STQ); Distributed Speech Recognition; Advanced Front-End Feature Extraction Algorithm; Compression Algorithms", Technical Report ETSI ES 202 050, November 2003.

[3] J.-Y. Li, B. Liu, R.-H. Wang, and L.-R. Dai, "A complexity reduction of ETSI standard advanced Front-End for DSR", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 1, pp. 61-64, Montreal, Canada, May 2004.

[4] C. Neves, A. Veiga, L. Sá, and F. Perdigão, "Efficient Noise-Robust Speech Recognition Front-End Based on the ETSI Standard", IEEE 9th International Conference on Signal Processing (ICSP), Beijing, China, October 2008.

[5] S. Young, G. Everman, et al, "The HTK Book (For Version 3.4)", University of Cambridge, England, 2006.

[6] S. Abate, and W. Menzel, "Automatic Speech Recognition for an Under Resourced Languaged – Amharic", Proc. Interspeech, pp. 1541-1544, Antwerp, Belgium, August 2007.

SESIÓN DE POSTER 1

Acoustic Event Recognition for Low Cost Language Identification

Danilo Spada, Ignacio Lopez, Doroteo T. Toledano, Joaquín González-Rodríguez

Biometric Recognition Group – ATVS
Escuela Politécnica Superior
Universidad Autónoma de Madrid, Spain

{danilo.spada, doroteo.torre, ignacio.lopez, joaquin.gonzalez}@uam.es

Abstract

One of the most popular approaches to Automatic Language Identification (LID) is Parallel Phone Recognition followed by Language Modeling (Parallel PRLM or PPRLM). This approach has proved to be very successful in LID. However, it has two mayor drawbacks: its high computational cost due to the need to run several phone recognizers on the same test segment; and the need to train the phone recognizers on manually transcribed data that may not match closely the type of speech on which the system will work. In this paper we present a novel approach for LID that tries to solve these two problems. It is based on substituting the phonetic recognizers by an Acoustic Event Recognizer (AER) that can be trained on untranscribed data and is much faster than the phone recognizers. Results show that this method, which we call AERLM, can be much faster than PRLM, although at the cost of reduced LID precision, and therefore suitable for low-cost LID.

1. Introduction

Automatic Language Identification is the task of recognizing the language spoken in a sample of speech. This automation can be very useful in multicultural environments like airports, congresses or international meetings. It can act as integrating part of all those services, both fully automated or not, that are able to act in different languages, adapting themselves to the user's spoken language.

Nowadays it is possible to distinguish two main groups of techniques for automatic language recognition: a high level approach (which uses acoustic features and linguistic units) and an acoustic approach (the algorithms which only use acoustic features). We can classify the most popular systems as the following:

acoustic level techniques:
- GMM, Gaussian Mixture Model classification;
- SVM-GLDS, Support Vector Machines with General Lineal Discriminant Sequence kernel;

high level techniques:
- PPR, parallel phone recognition;
- PRLM, phone recognition followed by language modeling;
- PPRLM, parallel PRLM;
- Improvements on PPRLM (lattices and SVM).

Most commonly used high level techniques can be grouped together as phonotactic techniques because they try to recognize languages based on the phones and sequences of most frequent phones in a language. This approach has two major drawbacks. Firstly, all these approaches are based on the concept of phoneme, a knowledge-based linguistic concept that is language-dependent and in many cases difficult to deal with in speech processing. An example of this difficulty is that in order to train a phoneme recognizer it is considered a requirement to have a manually phonetically transcribed database. Secondly, all these phonotactic approaches rely on phoneme recognizers that are costly in terms of computation, particularly when several recognizers in different languages are run in parallel as in PPRLM.

The first drawback is becoming less important with the increasingly number of transcribed speech corpora in different languages, as well as with techniques such as PRLM that don't require transcribed speech from a language in order to recognize it. However, the second drawback is becoming more and more important, particularly when emphasis is starting to be put not only in obtaining low-error systems, but also in obtaining low-cost systems [1].

In this paper, we present a modification of PRLM systems in which the Phonetic Recognizer (PR) is substituted by a data-driven Acoustic Event Recognizer (AER) to create what we call an AERLM system (Acoustic Event Recognizer followed by Language Modeling). In this way, we eliminate the need for phonetically transcribed data for training, which allows training the AER on data as close as possible to the testing (or working) data. Perhaps more importantly, AER is much faster than a phonetic recognizer, which makes it a good alternative to PRLM for limited-resource systems like embedded systems, as well as for a fast-matching stage prior to a more detailed matching.

The rest of the paper is organized as follows. Section 2 gives a panoramic view of our AERLM system. Section 3 gives more details about the Acoustic Event Segmentation. Section 4 describes our Acoustic Event Clustering. In section 5 we explain the language modeling and in section 6 our experiments. Finally, section 7 presents some conclusions as well as future work.

2. AERLM system

The AERLM technique is based on the same idea of PRLM: modeling and identifying languages based upon token sequences detected by a tokenizer. In AERLM, however, the tokenizer is not a phone recognizer. Our idea is to obtain transcriptions by an Acoustic Event Segmentation followed by an Acoustic Event Clustering.

The Acoustic Event Segmentation tries to approximate a phonetic segmentation. Segments are obtained using the variations in the spectrogram of speech, while the silence

segments are removed. A similar technique has been used by Glass and Zue [2] for speech recognition, and also by Chollet for language independent speaker recognition [3]. This last technique was successfully applied in the context of speaker recognition [4]. This work tries to apply a similar (but even simpler) technique to the domain of automatic language identification.

The Acoustic Events obtained are parameterized using 13 MFCC and each segment is represented by a vector obtained averaging all the vectors of the segment. The parameterized segments are then used to train a clustering algorithm and to obtain the transcriptions, given by the sequences of recognized cluster numbers. In a second step, we model and recognize each language using statistical information about the obtained transcriptions, like in the PRLM architecture. In the literature, we found that Heck and Sankar built a cluster for speech segmentation [5], and, more recently, clustering has been used in language recognition to improve the modeling of co-articulation behavior [6].

3. Acoustic Event Segmentation

The first step in the process is the segmentation of the utterance into acoustically stable segments that intend to represent phonemes or stable parts of phonemes. This segmentation is based on a Spectral Variation Function (SVF) based on the Euclidean distance between the static MFCCs to the left and right of the current frame. After this SVF has been computed the utterance is segmented initially into segments divided by the maxima of the SVF. After this initial segmentation is applied, a Voice Activity Detector (VAD) working on a segmental basis is applied. This VAD is based on the average energy on the segment and maximum and minimum durations of speech and silence pulses. After the VAD is applied all contiguous silence segments are unified into a single segment and are not considered for the rest of the processing. For the segments corresponding to speech we compute the average of the static MFCCs as well as deltas and double deltas within each segment. Given that the segmentation procedure is intended to produce spectrally stable segments it makes sense to summarize the spectral content of the segment by a single vector (although more testing would be required to determine whether this option is the best or it would be better to take the central vector, for instance). In this way, we expect to reduce the computational complexity of the system without compromising its discriminative power.

4. Acoustic Event Clustering

Clusters have been largely used in pattern analysis to compress data [7]. By using clusters, the feature space is divided into subspaces, each one represented by a centroid. The clustering algorithm goal is to substitute each data vector for its nearest centroid. Data compression is achieved by then replacing the centroid by its associated token.

Somehow in a similar way, phonetic transcriptors replace a sequence of data vectors by its associated phoneme. This work proposes a modification of the phonotactic language identification problem by replacing the HMM phoneme models of a phonetic trancriptor with an acoustic event segmentation followed by a clustering of the segments.

The use of clusters in modeling acoustic events has several advantages: (i) Reduced computational cost; (ii) Phonetically labeled data is not needed to train the cluster; and (iii) It is a complementary approach to the phonotactic language identification approach, therefore fusion of standard systems and our system is expected to improve performance.

Popular algorithms to solve this problem are k-means or binary splitting. Nevertheless, during the last years many other have been introduced [8].

In this paper we propose a data driven algorithm based on GMM modeling. After the acoustic event segmentation and the averaging of the MFCCs over each segment, the resulting average MFCCs from the training corpus are modeled using a GMM. This GMM is then used to cluster together all the average MFCC vectors that produce the maximum likelihood for the same Gaussian of the GMM. In this way, the whole segment producing the average MFCC vector is substituted by the Gaussian number. With this approach the number of tokens produced is the same as the number of Gaussians in the GMM. For our experiments we used 64, 128 and 512 Gaussians/clusters. .

Depending on the GMM training data, the cluster can be used to model different acoustic events. We can train the cluster on a single language (single-language clustering) or train it using data from different languages (multi-language clustering).

5. Language modeling

Independent language models are created by obtaining a transcription of a different training set by means of a nearest-neighbor with the previously trained cluster. The stored language model is formed by the frequencies of all the unigrams, bigrams and trigrams for the given codebook. A Universal Background Model (UBM) with information of several languages is also trained following the same algorithm.

The independent language models are adapted from the UBM in order to obtain adapted language models by means of the following formula:

$$P(c \mid \lambda A) = \alpha P(c \mid \lambda L) + (1-\alpha)(c \mid \lambda UBM) \quad (1)$$

where c is a given n-gram, λA is the adapted language model, λL the independent language model, λUBM the Universal Model, and α a given constant in the range [0-1].

Finally, test scores are computed for all the modeled languages using the adapted language model and the UBM .

6. Experiments

In this section we present language recognition results using as training material the LDC CallFriend database [9] and as testing database the evaluation data from the 2005 NIST Language Recognition Evaluation [10]. It is worth noting that we haven't made use of any phonetically transcribed data for these experiments.

This section is organized into 4 sections. The first one presents the experimental set-up, the second one presents results with a single clustering system trained on a single language. Then we build a system similar to a PPRLM system in which we train a clustering system for each language and then fuse all of them. In section 6.4 we introduce results

obtained training the cluster on more than one language and finally, section 6.5 compares our results with those obtained by our group in NIST 2005 Language Recognition Evaluation [10], where a more standard PPRLM system with 6 or 12 phonetic decoders were used.

6.1. Experimental set-up

Our base system has a front-end module that outputs 13 MFCC, Δ and ΔΔ parameters (39), but we only used the first 13 (static) MFCCs in the experiments described in this paper. Three different GMM base clusters, with 64, 128 and 512 mixtures have been trained for each of the following ten languages: english, mandarin, spanish, arabic, farsi, german, hindi, japanese, korean, tamil. For this task we used the complete LDC CallFriend database [9]. For each clustering, we processed this corpus to obtain transcriptions of all 12 languages. In a second step we used the transcriptions to train the UBM and to adapt the independent language models. We used unigrams, bigrams and trigrams. The testing material consisted of the samples of 30 seconds of the NIST LRE 2005 test corpus [10]. The selected target languages are: English, Hindi, Japanese, Korean, Mandarin, Spanish and Tamil. English, Mandarin and Spanish appear with two dialects. We applied TNorm to all experiments presented.

6.2. Results for a single AERLM system

For the systems using single language trained clustering, we obtained an EER around 36%.
In Figure 1 we present the results obtained processing the target languages with one of the system that seems to work better: the one using the clustering trained on Japanese.

Is it possible to imagine that the average computed during the acustic event segmentation and clustering reduces the amount of information available for discriminating among languages. An interesting aspect to explore in the future consists in removing this limitation by not averaging features over an acoustic event segment.

6.3. Parallel AERLM

The distribution of phonemes across languages may be different depending of the languages involved. In some cases, phonemes which are typical in one language may be rare in another. Therefore a clustering trained on a single language can introduce a loss of information.

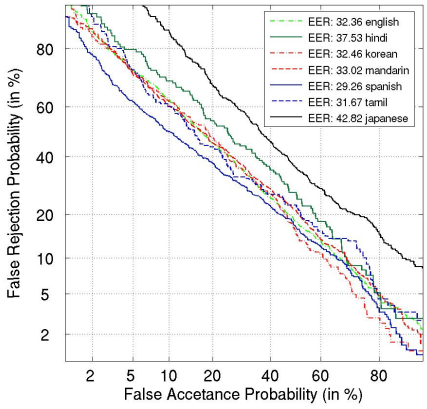


Figure 1: *Results on NIST LRE 2005 per Language for an AERLM system with AER trained on Japanese*.

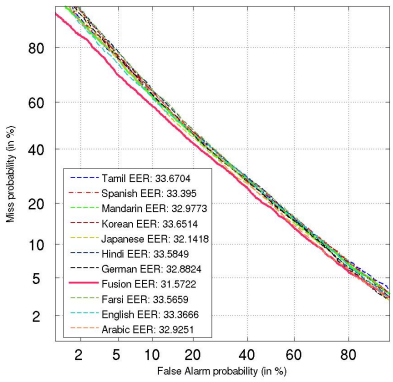


Figure 2: *Results on NIST LRE 2005, all languages. Parallel AERLM with AERs using 128 clusters trained for 10 languages.*

For PRLM language recognition systems Hazen proposed to train a phoneme recognizer on more than one language [11]. An alternative possibility is to run concurrently many systems and then fuse results: this is the base idea of PPRLM. The fusion of the different PRLM systems is typically obtained by using first a score normalization algorithm, for example T-Normalization [12], due to the intrinsic difference between the subsystems.

In our approach we explored both ideas: running many systems and fusing results (Parallel-AERLM) and training a cluster on many languages (Multi-language-Cluster-AERLM).The results obtained with Parallel AERLM are presented in Figure 2.

By fusing all AERLM systems we have obtained an absolute performance improvement of 1,5 points in EER relative to the single AERLM global behaviour.

6.4. Multi-Language AERLM system

A multi-language cluster, which models acoustic events from all languages, can be trained by using a language independent GMM. The main advantage of Multi-Language AERLM systems is that we do not need several language-dependent AERLM systems, as in the Parallel AERLM approach. So, we would obtain a computational cost of a tenth of the P-AERLM.

We explored the possibility to train a clusterig on more than one language using 64, 128 and 512 clusters. Results shows that the multilanguage system performance is similar to any single language AERLM system. Probably because, although the tokenization is richer, tokens are not discriminative enough.

6.5. Comparison with PPRLM systems

The main goals of the AERLM system were to improve the PPRLM system in two ways: firstly by avoiding the need to use phonetically transcribed speech to train the phonetic decoders, and secondly, by making language recognition much faster. In this sense the AERLM system achieved these goals since we don't need phonetically transcribed material any more and the computational cost for performing the NIST LRE 2005 test is only 80 hours with the 10 AERLMs in parallel, much less than the 500 hours that would require a system with 10 PRLM systems in parallel. Moreover, performance for any single-language AERLM system or the multi-language AERLM system is only slightly worse than

the 10 AERLM systems in parallel, and they consume 10 times less processing time. This last option is particularly adequate for resource-limite systems and also for a fast-matching module prior to a more detailed matching. In all cases experiments were run on a Pentium IV system at 2.4 GHz. with 1Gb RAM.

But of course the goal was to do this without significant degradation in performance. Unfortunately this last condition has not yet been met. Figure 4 presents results achieved by our group in 2005 NIST Language Recognition Evaluation (NIST LRE 2005). In there ATVS1 was a PPRLM system using 12 phonetic decoders trained on OGI Multi-Language Telephone Speech Corpus and ATVS2 was similar but with only 6 phonetic decoders. For 30s test segments results were between 20 and 22% in terms of EER. Our best AERLM system so far is still far from this result (31.5% EER), as presented in Figure 3. For future work we need to fine tune some parameters of the AERLM systems to try to get better performance while keeping the computational cost low.

7. Conclusions

We have analyzed the substitution of the phone recognizers in a PPRLM system by Acoustic Event Recognizers (AER) that are much faster and can be trained on untranscribed data. With this substitution, we can build AERLM and Parallel AERLM systems that are much faster than the corresponding PRLM and PPRLM systems and have the additional advantage that they can be trained on untranscribed data, thus increasing the amount of available training data and allowing for a better fit between the characteristics of the training and test (or working) data.

Given the reduced computational cost of the systems proposed, particularly for a single multi-language AERLM system, which performs almost as well as the 10-langauge Parallel AERLM system with 10 times less computational cost, we can envisage these type of systems as a very useful alternative to more computational complex systems for embedded devices or even as a fast-matching stage prior to a more detailed (and complex) matching where required.

8. References

[1] W. M. Campbell, D. E. Sturim, W. Shen, D. A. Reynolds, J. Navratil, "The MIT-LL/IBM 2006 Speaker Recognition System:High-Performance Reduced-Complexity Recognition", in IEEE ICASSP 2007, pp. 217-220.

[2] J.R Glass, V.W.Zue, "Multi-level acoustic segmentation of continuous speech", in Acoustics, Speech, and Signal Processing, ICASSP-88, New York, USA, 1988.

[3] G. Chollet, J. ˇ Cernock´y, A. Constantinescu, S. Deligne, and F. Bimbot, "Towards ALISP: a proposal for Automatic Language Independent Speech Processing," In Keith Ponting, editor, NATO ASI: Computational models of speech pattern processing Springer Verlag, 1999.

[4] Asmaa El Hannani1, Doroteo T. Toledano, Dijana Petrovska-Delacretaz, Alberto Montero-Asenjo and Jean Hennebert, "Using Data-driven and Phonetic Units for Speaker Verification", In Proceedings IEEE Odyssey 2006, San Juan, Puerto Rico.

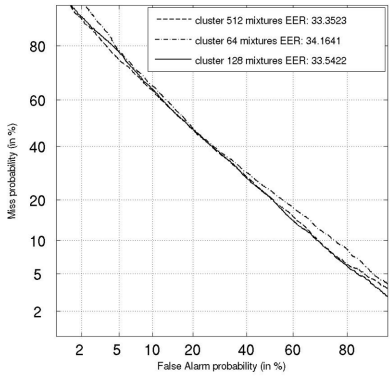


Figure 3: *Results on NIST LRE 2005, all languages. Multi-Language Cluster AER systems with 64, 128 and 512 clusters.*

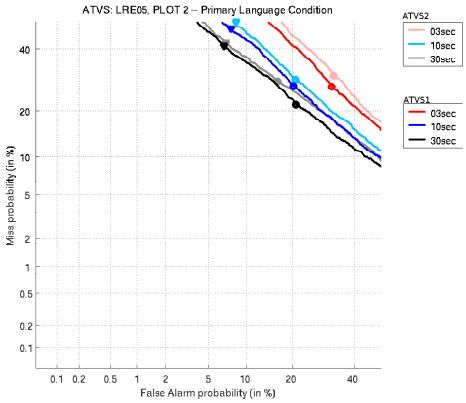


Figure 4: *Results achieved by ATVS on NIST 2005 LRE data using a standard PPRLM system trained on OGI Multi-Language Telephone Speech data.*

[5] L Heck, A Sankar "Acoustic Clustering and Adaptation for Robust Speech Recognition"- Proceedings of EUROSPEECH, Rhodes, Greece 1997

[6] Chien-Lin Huang, Chung-Hsien Wu, "Phone set generation based on acoustic and contextual analysis for multilingual speech recognition" Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '07, Hawaii 2007

[7] S. Theodoridis, and K. Koutroumbas., "Pattern Recognition" Second ed. Amsterdam, Elsevier Academic Press, 2003

[8] RO Duda, PE Hart, DG Stork "Pattern Classification", New York, Wiley-Interscience, 2000

[9] Linguistic Data Consortium, http://www.ldc.upenn.edu/

[10] "Speaker recognition evaluations", http://www.nist.gov/speech/

[11] TJ Hazen, VW Zue, "Recent improvements in an approach to segment-based automatic language identification", Proc. ICSLP '94, Yokohama, Japan, pp 1883-1886, 1994

[12] R Auckenthaler, M Carey, H Lloyd-Thomas, "Score Normalization for Text-Independent Speaker verification system", Digital Signal Processing, vol 10 2000, pp 42-54, 2000

APPLYING FEATURE REDUCTION ANALYSIS TO A PPRLM-MULTIPLE GAUSSIAN LANGUAGE IDENTIFICATION SYSTEM

Juan Manuel Lucas Cuesta, Ricardo de Córdoba Herralde, Luis Fernando D'Haro Enríquez

Grupo de Tecnología del Habla
Departamento de Ingeniería Electrónica
E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid
Ciudad Universitaria s/n. 28040. Madrid

ABSTRACT

This paper presents the application of a feature selection technique such as LDA to a language identification (LID) system. The baseline system consists of a PPRLM module followed by a multiple-Gaussian classifier. This classifier makes use of acoustic scores and duration features of each input utterance. We applied a dimension reduction of the feature space in order to achieve a faster and easier-trainable system. We imputed missing values of our vectors before projecting them on the new space. Our experiments show a very low performance reduction due to the dimension reduction approach. Using a single dimension projection the error rates we have obtained are about 8.73% taking into account the 22 most significant features.

1. INTRODUCTION

Automatic language identification (LID) has become a cornerstone task in multilingual environments. For an automatic customer care system which could be used for users that speak in different languages, a language-specific speech recognition module has to be used. So, determining the language in what the user speaks is a need in order to adapt further steps of a dialogue system.

The most widespread LID approach consists of using several phoneme recognizers in parallel. At the output of those recognizers, a phoneme language model is applied for each language to be identified. This technique is known as *Parallel Phone Recognition followed by Language Modeling* (PPRLM). Examples of this approach can be seen on [1] or [2].

Each of the phonemes of a given language can be estimated by using Gaussian Mixture Models (GMM, [3]) or Hidden Markov Models (HMM). A GMM-based LID system can be improved with a *clustering* algorithm that groups the feature vectors on an unsupervised approach, according to a distance criterion ([4]).

As an alternative to these probabilistic approaches, [5] or [6] develop neural network-based LID systems that lead to identification rates comparables to the obtained with PPRLM-based systems.

This work continues the presented in [1], [7] and [8], which present a LID system based on PPRLM. The performance of the baseline system is improved with the implementation of a multiple-Gaussian classifier. This subsystem takes its decisions using as input vectors the acoustic score of each phoneme within the input utterance, or the duration of those phonemes.

The number of features of each input vector is high, so the training and the evaluation of the Gaussian models takes an large fraction of processing time. In order to tackle this drawback, a feature selection algorithm such as LDA is proposed.

Since a given phoneme can or cannot appear on an utterance, several features may be missing on an input vector. This fact can cause a reduction of the system performance. To avoid this weakness we have analyzed several missing data imputation algorithms.

The rest of the paper is organized as follows. Section 2 presents a brief description of the dimensionality reduction approach that has been employed. The different imputation methods we have implemented are shown in Section 3. Our baseline LID system is then presented in section 4. Section 5 summarizes the different experiments we have carried out. Finally, Section 6 presents several conclusions of our work.

2. FEATURE SELECTION

The Gaussian classifiers we use as a second classification stage make use of 68-dimensional feature vectors. These 68 features are the acoustic scores of the phonemes of each target language (English and Spanish, 34 features each).

We propose a feature selection technique to reduce processing time and resources. Our system can choose the most representative features according to the following criterion:

$$\frac{\mu_1 - \mu_2}{\sigma_1^2\ \sigma_2^2} \tag{1}$$

being μ_1 and μ_2 the arithmetic means of a given feature considering each language, and σ_1^2 and σ_2^2, their corresponding variances. Higher values of 1 for a given feature

imply a better separation between the classes.

Despite the goodness of this approach, which can lead to better results using just 22 features instead of the whole feature vector, we want to analyze the behaviour of our system when a more restrictive reduction is applied. The chosen approach is Linear Discriminant Analysis (LDA), which is explained in [9].

We have chosen LDA because it is oriented to labeled samples, that is, the algorithm makes use of the language of each training utterance. Furthermore LDA tries to increase the separability between different class data, so it can be efficiently used for class discrimination.

LDA consists of applying a linear transform over a data set of *d* dimensions which project on a *d'*-dimensional subspace (*d'* < *d*), with *d'* equal to the number of languages minus 1 (in our case, *d'* = 1). The transformation is made in such a way that the between-class variance is maximized, while the within-class variance is minimized. This can lead to an improvement of the separability between the classes.

3. MISSING DATA TREATMENT

Our feature vectors consist of the acoustic score for each phoneme that has been observed on each input utterance. This fact implies that a given feature may not appear in a given vector. This could happen if the speaker has not used that phoneme or the system has not recognized it.

This lack of information is a drawback when dimension reduction is applied, because those missing values usually lead to a biased estimation of the optimal transformation vector. So the implementation of an imputation technique ([10]) that can fill those missing features is a must.

We have chosen two different imputation techniques: a substitution based on the arithmetic mean of the non-missing features, and a modification of the imputation approach proposed in [11].

The *mean imputation* procedure consists of evaluating the arithmetic mean of each feature using the non-missing values on the training data. Let $x_1, \cdots, x_n$ be a set of *n* *d*-dimensional feature vectors, which could have several missing values. The arithmetic mean of each feature *j* can be obtained as

$$\overline{X_j} = \frac{1}{n_j} \sum_{i=1}^{n} x_{ij} \tag{2}$$

where $n_j < n$ is the number of vectors for which the feature *j* is not missing.

This method is very simple and is accurate for classification purposes if a supervised training is carried out. However, mean substitution does not take into account the variance of each feature, so it can cause a bias in the estimation.

The imputation procedure proposed in [11] (henceforth referred to as *Bingham imputation*) computes the cross-case mean among the different non-missing features of a given vector *i*,

$$\overline{I_i} = \frac{1}{k} \sum_{j\ non-missing} \left(x_{ij} - \overline{X_j}\right) \tag{3}$$

where *k* is the number of non-missing features in vector *i* and x_{ij} is the value of the non-missing feature *j* of vector *i*.

The imputed value $\tilde{E_{ij}}$ of the missed feature *j* of vector *i* is computed as follows:

$$\tilde{E}_{ij} = \overline{X_j} + \overline{I_i} \tag{4}$$

So, the objective of this imputation is to include an offset in the imputation value that reflects the tendency that acoustic scores exhibit for the non-missing features in the vector.

This imputation method is especially effective when the different features are very correlated, because it weights each feature mean with the rest of the features in the vector. Nevertheless, this method does not take into account the feature variance. To tackle this lack we have modified the former definitions of cross-case mean and final imputation value:

$$\tilde{I}_i = \frac{1}{k} \sum_{jnon-missing} \frac{\left(x_{ij} - \overline{X_j}\right)}{\sigma_j} \tag{5}$$

$$\tilde{E}_{ij} = \overline{X_j} + \sigma_j\ \tilde{I}_i \tag{6}$$

where σ_j is the variance of feature *j*.

The inclusion of the variance provides a normalization of this imputed value, so that the values are more stable, avoiding the presence of outliers.

4. BASELINE SYSTEM

4.1. Database

Our database consists of a set of continuously spoken sentences extracted from conversations between airplane pilots and air traffic controllers. All speakers were native Spanish.

We have used 2929 Spanish sentences and 1053 English sentences. By applying a leave-one-out technique we have used each sentence for both training and evaluating the system, but obviously in separate sets. This way we expand the size of the test set. We have not considered those sentences whose duration is less than 0.5 seconds.

Each phoneme recognizer makes use of context-independent continuous hidden Markov models (HMM). We have considered 49 different phonemes for Spanish and 61 for English. However, we have grouped the less representative phonetic variations and built phoneme vectors of 68 features, 34 for each language.

4.2. PPRLM-based LID system

The PPRLM identification system makes use of a phoneme recognizer for each target language. A language model module scores the probability that the sequence of phonemes corresponds to a given language.

We have used smoothed *n*-gram language models to approximate the *n*-gram distribution as the weighted sum of the probabilities of the *n*-grams considered.

We improved the PPRLM approach by taking into account silence models, defining and using a smoothing function in the evaluation of the *n*-gram score, and removing bias in the classifier. The baseline error rate is about 3.7% using only PPRLM.

4.3. Gaussian classifier

We have used a second identification system that includes acoustic information of each phoneme. We built a feature vector with the phonemes that the PPRLM system has recognized. We computed an average score for each phoneme appearing in the sentence. Instead of using absolute scores for each phoneme, our previous work ([1]) demonstrated that we can achieve better identification rates by using differential scores obtained by the LM. We then applied equation (1) to get the most representative features. The best results that we have achieved showed an error rate of 7.9% when we use the acoustic score of each phoneme and keep 30 features in the reduced space. If we use the phoneme duration instead, error rate takes a value of 24.7%. This implies that phoneme duration is a much less discriminative feature, at least the way we have implemented it. These results will be our baseline.

5. EXPERIMENTS

5.1. LDA with mean substitution

The first imputation procedure we have implemented consists of substituting each missing value with the arithmetic mean of the corresponding feature and applying LDA. As the original feature space, we have used the 68-dimensional feature vectors as well as the selection of the most representative features, according to equation (1). The error rates are shown in Table 1 together with the relative improvement over the system without LDA (7.9% error rate) and the average percentage of missing values on the original feature space.

No of features	Error rate (%)	Improve (%)	Miss feat (%)
68	12.48	-58.0	31.63
30	9.46	-19.7	30.64
22	8.76	-10.9	26.41
20	8.94	-13.2	26.08

Table 1. *Error rates with LDA and mean substitution.*

The former average is similar for all setups with different number of features (close to 30%). This means that the most discriminant features also present a high number of missing values. So, the imputation of those missing values is still crucial.

We can also see how a pre-selection of the most representative features leads to a more accurate LDA projection. Nevertheless, the use of a low space dimension implies an information loss.

5.2. LDA with original Bingham imputation

Our second test makes use of the imputation algorithm presented in [11]. The following table summarizes the results we have obtained as well as the improvement in relation to the previous experiment.

No of features	Error rate (%)	Improve (%)
68	11.00	11.9
30	9.36	1.1
22	8.82	-0.7
20	9.03	-1.0

Table 2. *Error rates with LDA and Bingham substitution.*

If we compare these results with the previous ones we can see that Bingham imputation yields lower error rates when considering 68 and 30 features.

5.3. LDA with weighted Bingham imputation

We next weighted the cross-case mean of Bingham imputation by the variance of the corresponding feature, following equation (5). The different error rates for each input feature space are shown in Table 3, together with the relative improvement over the mean substitution-based experiments.

No of features	Error rate (%)	Improve (%)
68	12.24	1.9
30	9.21	2.6
22	8.73	0.3
20	9.02	-0.9

Table 3. *Error rates with LDA and weighted Bingham imputation.*

This results are slightly better than those obtained with mean substitution, except for the case of 20 features.

All the previous results are shown in Figure 1.

5.4. LDA applied to phoneme duration

If we consider the duration of each phoneme and apply both missing value imputation approaches we obtain the following results.

We can obtain a relevant improvement over the original error rate (24.7%). Despite the error rates are clearly higher than those obtained with acoustic scores, when we

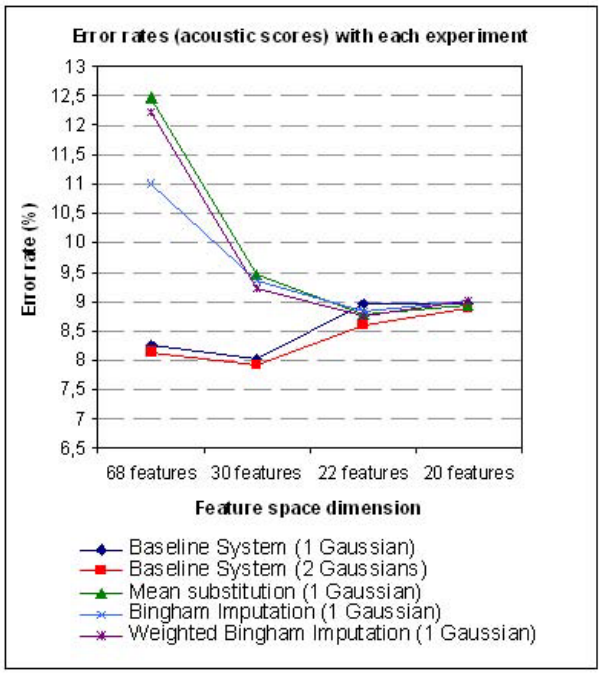


Figure 1. *Error rate comparison for acoustic scores.*

	Mean substitution		Basic Bingham imputation	
No of param	**Error rate (%)**	**Relative diff (%)**	**Error rate (%)**	**Relative diff (%)**
68	22.77	7.70	23.90	3.12
30	22.79	7.62	24.31	1.46

Table 4. *Error rates for LDA applied to phoneme duration.*

use both score and duration features we can improve the overall performance (8.6% error rate with 22 features and mean substitution).

6. CONCLUSIONS

In this work, we present a feature selection approach that makes use of several missing data imputation techniques in order to complete the input vectors with a low distortion. The increase in error rate due to the dimensionality reduction for the acoustic scores is relatively small, and the identification task becomes easier and faster for a multiple-language task.

The different imputation approaches allows us to accurately predict the values of the most representative features, so the results are very similar to those obtained with the original feature space, but using 1 dimension instead of 22. The best of the applied techniques has been variance-weighted Bingham imputation with 22 original features, with a slight improvement regarding the other techniques. Performance is even similar to the baseline system using 22 features.

7. ACKNOWLEDGEMENTS

This work has been partially funded by the Spanish Ministry of Education & Science under contracts DPI2007-66846-c02-02 (ROBONAUTA) and TIN2005-08660C04-04 (EDECAN-UPM) and by UPM-DGUI-CAM under CCG07-UPM/TIC-1823 (ANETO).

8. REFERENCES

[1] R. Córdoba et al., "Integration of acoustic information and PPRLM scores in a multiple-Gaussian classifier for language identification," *IEEE Odyssey*, 2006.

[2] K.C. Sim and H. Li, "Fusion of contrastive acoustic models for parallel phonotactic spoken language identification," *Interspeech*, pp. 170–173, 2007.

[3] Q. Dan, W. Bingxi, and Z. Qiang, "Two discriminative training schemes of GMM for language identification," *IEEE International Conference on Signal Processing (ICSP)*, pp. 630–633, 2004.

[4] B. Yin, E. Ambikairajah, and F. Chen, "Hierarchical language identification based on automatic language clustering," *Interspeech*, pp. 178–181, 2007.

[5] J. Braun and H. Levkowitz, "Automatic language identification with recurrent neural networks," *IEEE World Congress on Computational Intelligence and Neural Networks*, vol. 3, pp. 2184–2189, 1998.

[6] L. Wang, E. Ambikairajah, and E.H.C. Choi, "Multi-layer Kohonen self-organizing feature map for language identification," *Interspeech*, pp. 174–177, 2007.

[7] R. Córdoba et al., "A multiple-Gaussian classifier for language identification using acoustic information and PPRLM scores," *IV Jornadas en Tecnología del Habla*, pp. 45–48, 2006.

[8] R. Córdoba, L.F. D'Haro, F. Fernández-Martínez, J.M. Montero, and R. Barra, "Language identification using several sources of information with a multiple-Gaussian classifier," *Interspeech*, pp. 2137–2140, 2007.

[9] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, Wiley Interscience, second edition, 2001.

[10] E. Acuña and C. Rodríguez, "The treatment of missing values and its effect in the classifier accuracy," *Classification, Clustering and Data Mining Applications*, pp. 639–648, 2004.

[11] C.R. Bingham, M. Stemmler, A.C. Petersen, and J.A. Graber, "Imputing missing data values in repeated measurement within-subject designs," *Methods of Psychological Research*, vol. 3, no. 2, pp. 131–155, 1998.

BIO-INSPIRED DYNAMIC FORMANT TRACKING FOR PHONETIC LABELLING

P. Gómez, J. M. Ferrández, V. Rodellar, R. Martínez, C. Muñoz, A. Álvarez, L. M. Mazaira

Grupo de Informática Aplicada al Tratamiento de Señal e Imagen, Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo, s/n, 28660 Madrid, Spain
e-mail: pedro@pino.datsi.fi.upm.es

ABSTRACT

It is a known fact that phonetic labeling may be relevant in helping current Automatic Speech Recognition (ASR) when combined with classical parsing systems as HMM's by reducing the search space. Through the present paper a method for Phonetic Broad-Class Labeling (PCL) based on speech perception in the high auditory centers is described. The methodology is based in the operation of CF (Characteristic Frequency) and FM (Frequency Modulation) neurons in the cochlear nucleus and cortical complex of the human auditory apparatus in the automatic detection of formants and formant dynamics on speech. Results obtained in formant detection and dynamic formant tracking are given and the applicability of the method to Speech Processing is discussed.

1. INTRODUCTION

Bio-inspired Speech Processing is the treatment of speech following paradigms used by the human sound perception system, which has developed specific structures for this purpose. The purpose of the present paper is to provide a hierarchical description of speech processing by bio-inspired methods discussing the fundamentals of speech understanding, helping to devise a general bio-inspired architecture for Cognitive Audio in the long range [6]. For such, the dynamic tracking of formants has been selected as an objective in improving ASR. Speech may be divided in voiced and unvoiced segments, depending if vocal fold activity is present or not. Each one of them would imply a different representation under the spectral point of view, voiced sounds being dominated by the action of strong harmonic series filtered by the changing vocal tract transfer function modified constantly by the articulation organs. This stands also for the vocalic core of the syllables (except in the case of whispered speech). For unvoiced speech there is still a strong coloring of the sibilant sounds produced in plosives and fricatives resulting from the positions where air constrains leading to turbulence occur. Normal speech may be perceived as sequences of harmonic series filtered by the resonances of the vocal tract (formants) with characteristic onsets and trails, which may be preceded or followed by noisy bursts as shown in Figure 1 for four syllables of the sort V-C-V where C stands for a specific voiced approximant.

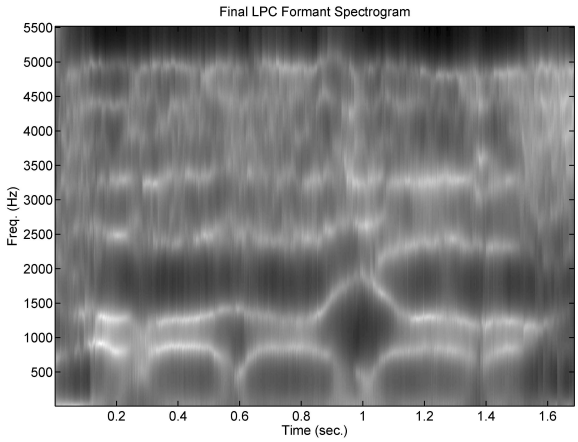


Figure 1. Adaptive Lineal Prediction (ALP) Spectrogram corresponding to the syllables /aβa-aða-aʒa-aɣa/ uttered by a Spanish male speaker. The IPA has been used for annotation [1].

This example has been selected for the fast dynamic movements of formants present in it, as dynamic formant tracking for phonetic class labelling is the aim of the work.

2. SPEECH PERCEPTION

Speech is perceived by the Auditory System described in Figure 2 as a chain of different sub-systems integrated by the Peripheral Auditory System (Outer, Middle and Inner Ear) and the Higher Auditory Centers. The most important organ of the Peripheral Auditory System is the Cochlea (Inner Ear), which carries out the separation in frequency and time of the different components of Speech and their transduction from mechanical to neural activity. The excitation of transducer cells (hair-cells) responsible for the mechanical to neural transduction process is tono-topic. Electrical impulses propagate to higher neural centers through auditory nerve fibers of different characteristic frequencies (CF) responding to the spectral components (F0, F1, F2...) of speech [10].

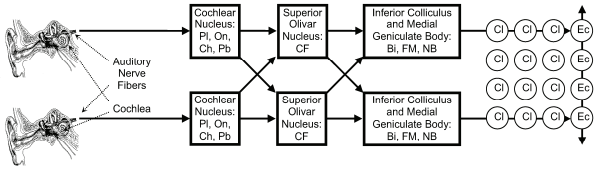


Figure 2. Speech Perception Model. The Cochlea produces time-frequency organized representations which are conveyed by the Auditory Nerve to the Cochlear Nucleus, where certain specialized neurons (Pl: Primary-like, On: Onset, Ch: Chopper, Pb: Pauser) are implied in temporal processing. Binaural information is treated in the Superior Olivar Nucleus, where tono-topic units (CF) have been identified. Other units specialized in detecting tonal movements (FM), broadband spectral densities (NB) and binaural processing (Bi) are found in the Inferior Colliculus and the Medial Geniculate Body. The Auditory Cortex shows columnar layered units (Cl) as well as massively extensive connection units (Ec).

Within the cochlear nucleus (CN) different types of neurons are specialized in segmenting the signals (Ch: chopper units), detecting stimuli onsets (On: onset cells), delaying the information (Pb: pauser units), or acting as relay stations (Pl: primary-like units). The Cochlear Nucleus feeds information to the Olivar Complex, where sound localization is derived from inter-aural differences, and to the Inferior Colliculus (IC) organized in spherical layers with orthogonal isofrequency bands. Delay lines are found in this structure to detect temporal features in acoustic signals (CF and FM components). The thalamus (Medial Geniculate Body) acts as a last relay station, and as a tonotopic mapper of information arriving to cortex as ordered feature maps.

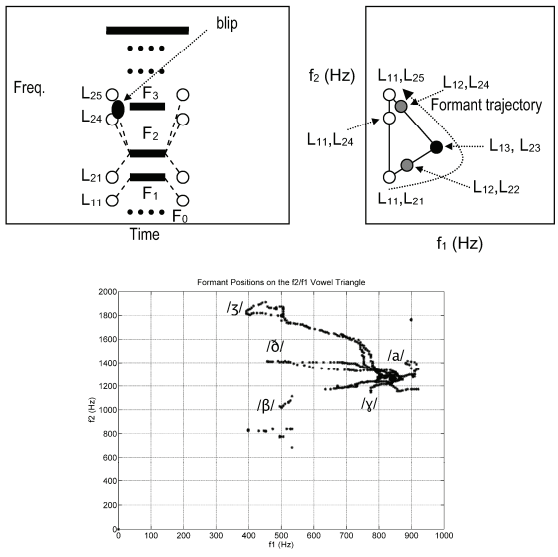


Figure 3. Generalized Phoneme Model. Top left: loci of the GPM on the vowel triangle. White circles indicate the positions of the loci. Top right: Dynamic trajectories on the vowel triangle. Bottom: Formant trajectories for the trace /aβaðaʒaɣa/ shown in Figure 1. The dark dot gives the position of the specific vowel modeled (/a/ in the present case).

Neurons have been found in the cortex that fire when FM-like frequency transitions are present (FM elements), while some others respond to specific noise bursts (NB components). Other neurons are specialized in detecting the combinations among these elements. In humans, evidence exists of a frequency representation map in the Heschl circumvolution and of a secondary map with word-addressing capabilities. A comprehensive review of the structures involved and their functionality is given in [4]. As a summary the specific processing of speech by the Auditory System is based on the hierarchical detection and association of stable frequencies, onset times, dynamic frequency changes, and tone bursts. At a higher hierarchy dynamic changes in formants (onset times and slopes) and specific broadband signals present before the onset time define specific clues to the perception of syllables, seen as structures of consonants and vowels as in C-V, C-V-C, V-C-V, etc. The perceptual interpretation of such structures is well known since the works of Delattre et al. [2]. From these studies a Generalized Phoneme Model may be issued as represented in Figure 3. The static version of the model is based on formant positions and loci (places marking the starting and ending points of formant trajectories. The dynamic version is based on a projection on the vowel triangle (f2 vs f1).

3. BIO-INSPIRED SPEECH PROCESSING

From the study of the Generalized Phoneme Model and the Auditory Speech Processing fundamentals, a Basic Neuron Set could be defined as an algorithmic structure operating both in the time and frequency domain modeling speech features, among these: Lateral Inhibition Units (LI) finite difference algorithms in the frequency domain profiling formants; Positive Frequency Modulation Units (PfM), detectors of up-hill formant displacements; Negative Frequency Modulation Units (NfM), detectors of down-hill formant displacements; Characteristic Frequency Units (CF), detectors of stable frequency positions; Vowel-Spotting Units (VS), detectors of stable or parallel-moving pairs of frequencies and Noise-Burst Units (NB), detectors of wide-band noise-like signals. These elementary processing units could be implemented by the general structure shown in Figure 4.

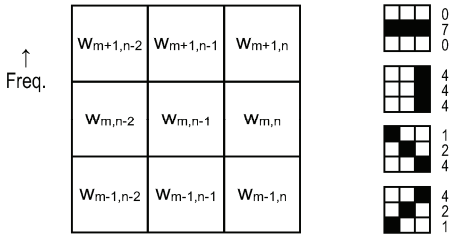


Figure 4. Basic Neuron Set for elementary operations on time-frequency representations of speech. Left: 3x3 weight mask. Right: Masks for feature detection on the formant spectrogram. Each mask is labelled with the corresponding octal code (most significant bits: bottom-right). Labels 070, 444, 124 and 421 correspond respectively with CfI, NB, NfM, PfM units.

In this way the problem of feature detection in formant spectrograms is related to a well known one in Digital Image Processing [8]. A classical method is based on the use of reticule masks on the spectrogram *X(m,n)*:

$$\tilde{X}(m,n) = \sum_{i=-1}^{I} \sum_{j=0}^{J} w_{i,j} X(m-i, n-j) \quad (1)$$

where $\{w_{ij}\}$ is a *IxJ* mask with a specific pattern and a set of weights, which may be adjusted adaptively. The spectrogram is built using ALP algorithms producing all-pole spectral positions which keep track of the vocal tract resonances [3]. Precise formant positions may be obtained from these rough representations applying lateral inhibition between neighbor CF units using specific weight configurations of the mask in Figure 4 as shown in Figure 5.

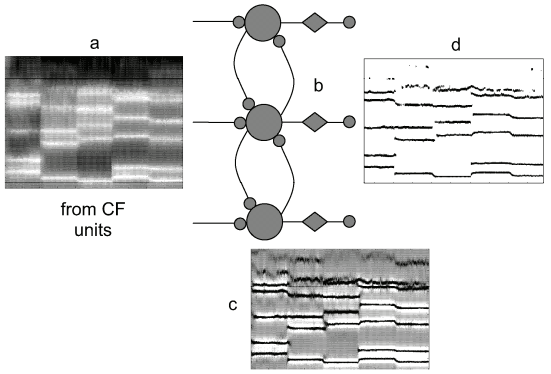


Figure 5. Formant Trajectory Profiling for the sequence /aeiou/ (Spanish, male speaker): The speech spectral density (a) as detected by CF units (see next section) is processed by columns of neurons implementing lateral inhibition (b), producing differentially expressed formant lines (c), which are transformed into narrow formant trajectories (d) after non-linear saturation.

The lateral inhibition filter produces sharp estimations of the spectral peaks (see Figure 5.b). The final formant distribution is given in Figure 5.d after adaptive saturation. Other personalized neurons can be used for the detection of time-frequency features, as CF or FM patterns as shown in the systemic framework given in Figure 6. The first operation on the LPC spectrogram will be to profile formant trajectories using lateral inhibition as described. The rest of the structure works as follows: PfM and NfM are neurons specialized in detecting positive and negative movements formants, firing in response to different slopes ($+fM_{1\text{-}k}$, $-fM_{1\text{-}k}$); CF are neurons detecting the stable positions of formants firing when a given channel is active during a specific interval ($f1_{1\text{-}k}$ and $f2_{1\text{-}k}$ being the bands associated to the first two formants); $NB_{1\text{-}k}$ are neurons which fire when broad band activity is detected; Σ units (middle left) are specialized in adding formant dynamics, integrating channel activity ($\int$) and thresholding (f). Lateral inhibition is again used in eliminating possible ambiguities in the final detected activity in the first two formants (Dynamic Tracking Units +fM1, -fM1, +fM2, -fM2); finally Vowel Spotting Units and Voiceless Activity may be derived from f1, f2 and NB channels.

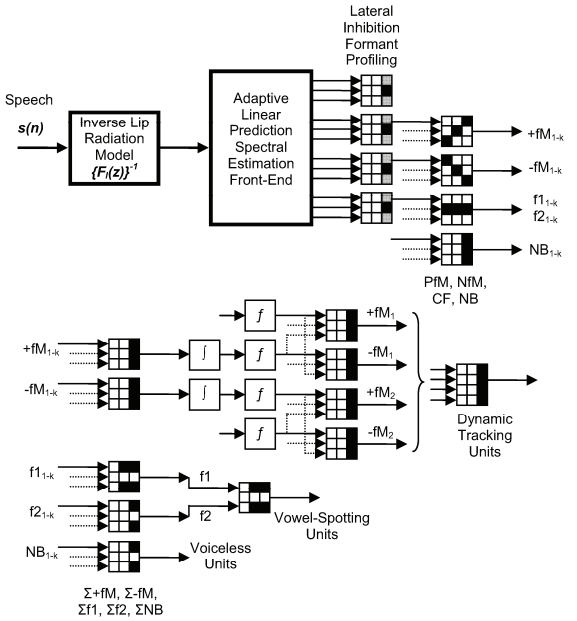


Figure 6. Bio-inspired Speech Processing Framework used in the study for a mono-aural channel.

4. RESULTS AND DISCUSSION

As an example Figure 7 illustrates the activity of Dynamic Tracking Units in processing a specific sentence as {es hábil un solo día}.

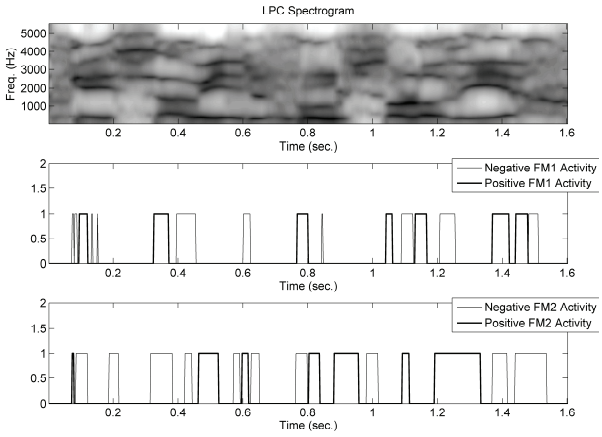


Figure 7. Detection of formant dynamics from the ALP spectrogram (top) using lateral inhibition and nonlinear saturation. The positive and negative slopes for f1 (middle) and f2 (bottom) have been detected from the sentence {es hábil un solo día} uttered by a male speaker (046).

The long and intense climbing up and sliding down of f2 for /δía/ can be appreciated in the lowest template of the between 1.2-1.35 and 1.37-1.53 (separated in two different intervals in this last case). The combinations of these four signals (+fM1, -fM1, +fM2, -fM2) are a first broad labeling of the series of approximants studied. The estimates of the values of the two slopes of /δía/ for 8 male and 8 female speakers are given in Table 1.

Table 1. Second formant positive and negative slopes for /δía/

Male speakers (Hz/sec)			Female speakers (Hz/sec)		
Speaker	+fM2	-fM2	Speaker	+fM2	-fM2
046	6899	-8191	A23	7867	-6921
081	6603	-5955	A66	769	-2216
115	8450	-3445	B31	5696	-18079
126	5662	-4915	B97	6591	-2791
160	6065	-4688	C76	3166	-5191
208	7660	-5666	D45	19539	-14086
231	5779	-3095	D77	6497	-5088
304	6302	-18542	D87	6544	-2512

It may be seen that with certain exceptions the values of the slopes range from 5000 to 8000 Hz/sec. More research is to be conducted to determine the robustness of the estimates. Dynamic formant trajectory detection and characterization is important for forensic applications. The methodology presented may be used also for the detection of the statistical vertices of vowel triangles for different speakers, as given in Figure 8, derived from the formant trajectories of the same sentence {es hábil un solo día} produced by the same set of 8 male and 8 female speakers.

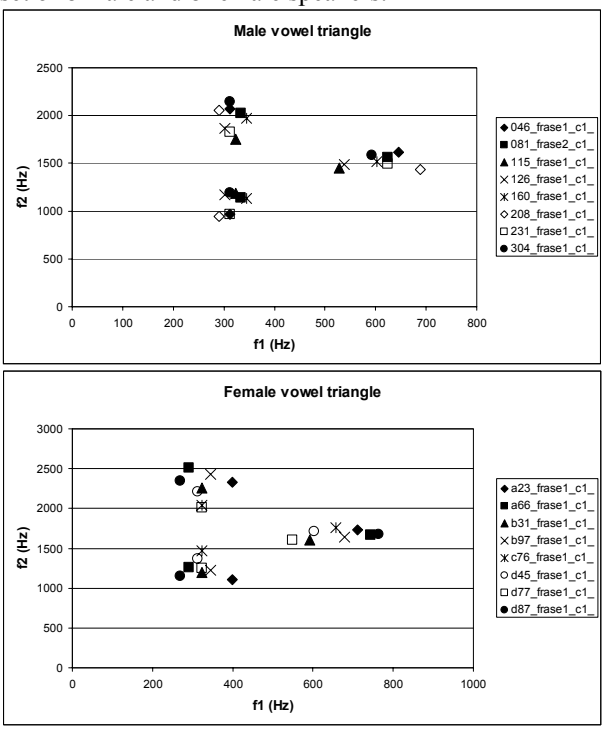


Figure 8. Detection of the vowel triangle centroids for eight male and eight female speakers using the bio-inspired methodology proposed. The positions of the vertices have been estimated using the lowest and highest quantiles of f1 and f2 statistical distributions for each speaker.

In general it may be observed that statistical spread is larger in female than in male, and that the upper left vertex is the one showing larger inter-gender differences.

5. CONCLUSIONS

Through the present work a hierarchical architecture to detect and label broad class phonetic features has been presented using replications of a Basic Neuron Set. The results show the viability of bio-inspired phonetic feature detection using combinations of these computationally inexpensive structures. The structures proposed are able of signaling stable, ascending and descending formants, and noise bursts. This may be of great help in improving recognition rates in ASR as much as 26% (see [7]) by simplifying State-Transition Graph Search in HMM parsing. More work is to be done to establish normalized thresholds and configuration parameters to improve robustness. Preliminary studies show that the statistical performance of the methodology show improvements in labeling of around 6-10% against blind supervised labeling, although this study is not complete yet. These questions remain the object of future study, as well as the role of the columnar organization of the Auditory Cortex [9] to include short-time memory and retrieval by Generalized Autoregressive Units.

ACKNOWLEDGMENTS

This work is being funded by grants TEC2006-12887-C02-01/02 from Plan Nacional de I+D+i, Ministry of Education and Science, by grant CCG06-UPM/TIC-0028 from CAM/UPM, and by project HESPERIA (http.//www.proyecto-hesperia.org) from the Programme CENIT, Centro para el Desarrollo Tecnológico Industrial, Ministry of Industry, Spain.

REFERENCES

[1] Available at http://www.arts.gla.ac.uk/IPA/ipachart.html

[2] Delattre, P., Liberman, A., Cooper, F.: Acoustic loci and transitional cues for consonants. *J. Acoust. Soc. Am*., Vol. 27, pp. 769-773, 1955.

[3] Deller, J. R., Proakis, J. G., and Hansen, J. H. L.: *Discrete-Time Processing of Speech Signals,* Macmillan, NY, 1993.

[4] Ferrández, J. M.: Study and Realization of a Bio-inspired Hierarchical Architecture for Speech Recognition. Ph.D. Thesis (in Spanish), Universidad Politécnica de Madrid, 1998.

[5] Goldstein, E. B., *Sensation and Perception*, Wadsworth, Belmont, CA., 2006.

[6] Gómez, P., Ferrández, J. M., Rodellar, V., Álvarez, A., Mazaira, L. M., "A Bio-inspired Architecture for Cognitive Audio", *Lecture Notes on Computer Science*, Vol. 4527, pp. 132-142, 2007.

[7] Gravier, G., Yvon, Y., Jacob B. and Bimbot, F., "Introducing contextual transcription rules in large vocabulary speech recognition", in *The integration of phonetic knowledge in speech technology*, William J. Barry and Win A. Van Domelen Eds, Springer series on Text, Speech and Language Technology, vol. 25, chapter 8, pp. 87-106, 2005.

[8] Jähne, B., *Digital Image Processing*, Springer, Berlin, 2005.

[9] Mountcastle, V. B., "The columnar organization of the neocortex", *Brain*, Vol. 120, pp. 701-722, 1997.

[10] Shamma, S., "On the role of space and time auditory processing", Trends in Cognitive Sciences, Vol., No. 8, pp. 340-348, 2001.

COMUNICA - PLATAFORMA PARA EL DESARROLLO, DISTRIBUCIÓN Y EVALUACIÓN DE HERRAMIENTAS LOGOPÉDICAS ASISTIDAS POR ORDENADOR

Oscar Saz, W.-Ricardo Rodríguez, Eduardo Lleida, Carlos Vaquero, Antonio Escartín

Grupo de Tecnologías de las Comunicaciones (GTC)
Instituto de Investigación en Ingeniería de Aragón (I3A)
Universidad de Zaragoza, España
{oskarsaz,wricardo,lleida,cvaquero}@unizar.es, aescartinv@gmail.com

RESUMEN

Este trabajo presenta *Comunica*, la plataforma desarrollada por el Instituto de Investigación en Ingeniería de Aragón (I3A) en colaboración con diversas entidades educativas para el desarrollo y distribución de diferentes herramientas de logopedia y enseñanza lingüística basadas en Tecnologías del Habla. El conjunto de aplicaciones desarrolladas proporciona entrenamiento desde los niveles prelingüísticos a los niveles pragmáticos del lenguaje utilizando diferentes Tecnologías del Habla para proveer el aprendizaje de una forma rápida, efectiva y no supervisada. *Comunica* propone también un cauce de comunicación directo con la comunidad educativa a través de Internet que permite conocer su valoración de las herramientas, evaluarlas y evolucionarlas para cubrir las necesidades existentes. La viabilidad del proyecto se estudia mediante la valoración de las herramientas por los usuarios y mediante el estudio y desarrollo de nuevos algoritmos para el trabajo con pacientes con alteraciones en el habla y el lenguaje.

1. INTRODUCCIÓN

Existen cada vez un mayor número de aplicaciones orientadas a la enseñanza que buscan aprovechar la potencialidad de interacción que presenta hoy en día la expansión y crecimiento de los ordenadores personales e Internet. En estos casos, se busca aprovechar la facilidad que las nuevas generaciones de alumnos presentan con estos elementos y utilizar su multimodalidad en la interacción para llevar a cabo un aprendizaje de un modo semisupervisado que apoye la labor del educador. Esta interactividad de la enseñanza ha encontrado un especial desarrollo en las aplicaciones de apoyo a la enseñanza del lenguaje, tanto en cuanto se permite el uso de la multimodalidad con imagen y sonido para la interacción y las Tecnologías del Habla que son la base de estas aplicaciones permiten una enseñanza robusta y no supervisada. Los proyectos realizados en este sentido han sido variados y se han orientado a mejorar la capacidad articulatoria de los alumnos [1], su capacidad de lectura oral y comprensión [2] o el aprendizaje de idiomas extranjeros [3], entre otros.

Este tipo de aplicaciones requieren de avances en las Tecnologías del Habla en que se basan, Reconocimiento Automático del Habla [4] y verificación de pronunciaciones principalmente, para proveer de una realimentación efectiva y robusta al alumno sobre sus capacidades y necesidades en el proceso de aprendizaje. Actualmente ya se han desarrollado avances en este sentido que ponen estas tecnologías al alcance de la comunidad logopédica.

Es por eso, que el Instituto de Investigación en Ingeniería de Aragón (I3A) ha puesto en marcha *Comunica* para utilizar el conocimiento adquirido en otras tareas de Tecnologías del Habla y aplicarlo al campo de la discapacidad, pudiendo llegar a cubrir el mayor número de áreas posibles. El presente trabajo presenta las circunstancias que rodean la implantación de *Comunica* como plataforma para la distribución y el desarrollo de las aplicaciones logopédicas desarrolladas. En trabajos previos se pueden encontrar resultados científico-técnicos sobre las aplicaciones y algoritmos desarrollados [5, 6, 7, 8].

La organización del artículo es la siguiente: En la Sección 2 se presentan las motivaciones y objetivos principales en *Comunica*. La Sección 3 proporciona una revisión de las herramientas desarrolladas en *Comunica*. Los medios de distribución y evaluación de las herramientas son presentados en las Secciones 4 y 5 respectivamente. Finalmente, las conclusiones de este trabajo se extraen en la Sección 6.

2. OBJETIVOS Y REQUERIMIENTOS

Los objetivos dentro de *Comunica* son el proveer de un marco de desarrollo, evaluación y distribución estable de herramientas informáticas de ayuda a la logopedia. Los requerimientos que se propusieron son los siguientes:

Desde el punto científico-técnico, se busca que las herramientas provean una mejora logopédica a través de su uso de forma fiable. Los sistemas de Tecnologías del Habla utilizados en ellas deben ser robustos y capaces de enfrentarse a cualquier característica personal en el trastorno del usuario. La evaluación de los sistemas debe ser llevada a cabo para garantizar esta situación.

Desde el punto de vista de aplicación, interfaz e interacción, se debe buscar facilitar en todo momento a los

Este trabajo ha sido subvencionado por el proyecto TIN-2005-08660-C04-01 del Ministerio de Educación y Ciencia del Gobierno de España.

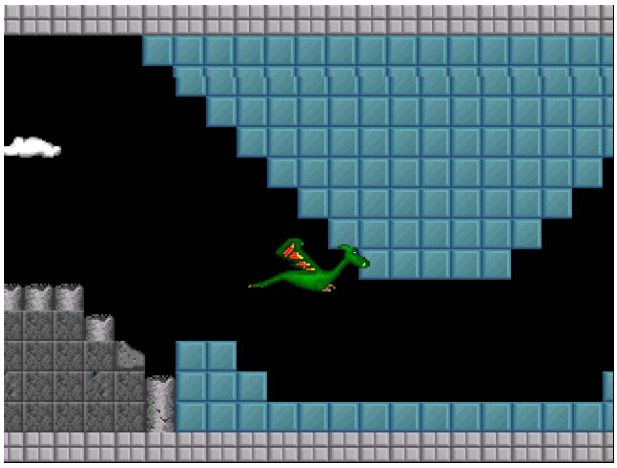


Figura 1. *Realimentación visual en* PreLingua.

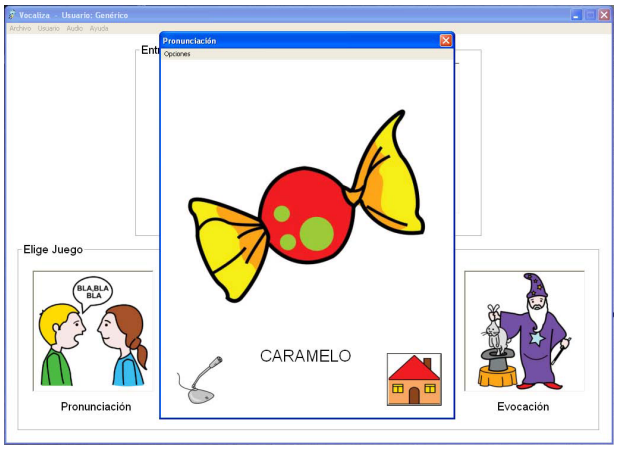


Figura 2. *Interfaz visual en* Vocaliza *y* VocalizaL2.

usuarios un manejo sencillo de las herramientas. La complejidad de las Tecnologías del Habla implementadas no debe verse en la superficie y tanto alumnos como educadores deben poder configurar y trabajar fácilmente con las herramientas. El trabajo con la herramienta debe poderse llevar a cabo de forma no supervisada para facilitar las sesiones de logopedia y alcanzar al mayor número de alumnos posible. Por otro lado, se ahonda en la necesidad de trabajar con sistemas de comunicación aumentada y alternativa que permitan la total accesibilidad por parte de los alumnos independientemente de sus capacidades físicas o psíquicas.

3. HERRAMIENTAS DESARROLLADAS

En esta Sección se presentan las cuatro herramientas desarrolladas bajo *Comunica* orientadas a apoyar el aprendizaje lingüístico para cuatro grupos de usuarios distintos.

3.1. PreLingua

PreLingua es una herramienta orientada al trabajo de las características prelingüísticas por parte de alumnos en fase pre-oral [6]. Consiste en un conjunto de aplicaciones que mediante realimentación visual como la presentada en la Figura 1 buscan hacer consciente al alumno de las características básicas de la producción oral. Los elementos trabajados actualmente por *PreLingua* son la distinción entre voz y no voz (distinguiendo a su vez la producción de sonidos sordos de los sonidos sonoros), el control de la intensidad de la voz, de la frecuencia fundamental y la producción de las vocales del alumno.

La tecnología debajo de las aplicaciones se basa en procesado de señal de voz como detectores de voz-silencio, estimadores de pitch basados en error de predicción de los Coeficientes de Predicción Lineal (LPC) y detección y análisis de formantes. La Figura 1 presenta una actividad consistente en salir de un laberinto en la que el movimiento en el eje horizontal se produce según la presencia o no de voz y el movimiento en el eje vertical se puede controlar por la intensidad de la emisión o por el valor del pitch.

3.2. Vocaliza

Vocaliza se dirige al trabajo con alumnos con problemas en el nivel articulatorio del lenguaje [5] mediante el trabajo con la producción de palabras aisladas, frases simples e introducción de la semántica mediante adivinanzas con respuesta en forma también de palabras aisladas. Las actividades con las que se trabaja esta capacidad articulatoria del alumno presentan una interacción basada en texto, imagen y audio como se puede ver en la Figura 2 para facilitar el trabajo del alumno. Cada actividad provee de una forma audio-visual realimentación al mismo sobre su capacidad oral para llevarla a cabo.

La aplicación utiliza Reconocimiento Automático del Habla en cada una de las actividades para decidir cuál ha sido la realización oral del usuario y dar o no la actividad por superada. Para evaluar la capacidad articulatoria del usuario se utiliza un algoritmo de evaluación de pronunciaciones a nivel de palabra [5] que le indica la calidad de su pronunciación una vez que la actividad ha sido superada.

3.3. Cuéntame

Cuéntame apunta al trabajo en niveles superiores del lenguaje que en *PreLingua* y *Vocaliza*. En este caso, se trabaja con alumnos con una articulación buena que, debido a trastornos del desarrollo presentan problemas en el uso del lenguaje como herramienta de comunicación. Este tipo de trastornos pueden suponer que el alumno no sea capaz de crear frases completas o sufra un bloqueo lingüístico a la hora de llevar una conversación o responder a preguntas concretas de sus padres o educadores.

La aplicación trabaja todos estos elementos con un interfaz audio-visual similar a *Vocaliza* planteando tres actividades diferentes: Respuesta a preguntas, donde el alumno debe responder con frases sintácticamente correctas a las diferentes preguntas planteadas por la aplicación; descripción de objetos, donde el alumno da la descripción de los objetos presentados de acuerdo a unas cualidades predefinidas que la aplicación le propone; y, por último, navegación por un escenario e interacción virtual con objetos como se presenta en la Figura 3, donde

Figura 3. *Interfaz visual en* Cuéntame *en la actividad de escenario.*

el alumno lleva a cabo un diálogo con la aplicación para conseguir un objetivo planteado por la misma. Esta herramienta prioriza la evaluación de la capacidad de generación del lenguaje del alumno sobre la articulatoria, evaluando la complejidad de la frase utilizada por el alumno y su aproximación al modelo de lenguaje estimado por la propia aplicación.

3.4. VocalizaL2

VocalizaL2 es una evolución de *Vocaliza* que proporciona una realimentación más precisa al usuario en su pronunciación. *Vocaliza* fue diseñado para proporcionar una evaluación de la pronunciación a nivel de palabra que fuese más fácil para los alumnos de educación especial a los que estaba inicialmente dirigido. *VocalizaL2* busca dar de una evaluación a nivel fonético que permita al usuario conocer con precisión qué partes del conjunto de fonemas del Español le presentan más dificultad.

De esta forma, *VocalizaL2* se dirige también especialmente a usuarios interesados en el aprendizaje del Español como segundo idioma. Desde el punto de vista tecnológico, la aplicación da la evaluación a nivel de fonema a través de una medida de confianza calculada como el resultado de obtener la red de fonemas que mejor se aproxima en términos de verosimilitud mediante el algoritmo de Viterbi a la señal producida por el usuario [8] y compararla a la cadena de fonemas teóricamente producida por el usuario según se le ha requerido por la aplicación como se muestra en la Figura 2.

4. DISTRIBUCIÓN

La distribución es uno de los motores que se le ha querido dar a *Comunica*. Ninguna de las aplicaciones desarrolladas tendría sentido si no se buscara hacerlas llegar al mayor número posible de usuarios. En este caso, se utiliza la potencialidad que Internet ofrece para llegar al mayor número de personas. La distribución a través de la página web de *Comunica* [9] permite una distribución sin trabas y en contacto con todos los usuarios, educadores y logopedas de España y Latino América que han mostrado su interés en las aplicaciones desarrolladas.

Entre los recursos que proporciona la página web está la posibilidad de realizar consultas sobre las aplicaciones, recibir boletines con las novedades que se van relizando, descargar de forma gratuita todas las aplicaciones y sus recursos y acceder a las publicaciones científico-técnicas generadas en *Comunica* entre otras. En la web, también se puede obtener una selección de pictogramas desarrollados por el Centro Aragonés de Tecnologías de la Educación (CATEDU) para su integración en *Vocaliza* y que también se encuentran disponibles en el portal Aragonés de la Comunicación Aumentativa y Alternativa (ARASAAC).

5. EVALUACIÓN

La evaluación de los sistemas en *Comunica* se lleva a cabo trabajando en dos direcciones: Por un lado, evaluar la capacidad pedagógica de las herramientas y por otro lado, evaluar los algoritmos de Tecnologías del Habla utilizados en las herramientas y evolucionarlos.

5.1. Evaluación Pedagógica

Para la evaluación de las herramientas, se ha trabajado con diversas instituciones educativas. En el desarrollo tanto de *PreLingua* como de *Vocaliza* y *Cuéntame* se ha trabajado conjuntamente con el Centro de Educación Especial (CPEE) Alborada [10]. Sus educadores han valorado positivamente el uso de la multimodalidad y la interacción oral como forma de motivar al alumnado de la escuela para el trabajo diario en logopedia. También valoran la facilidad que tienen las Tecnologías del Habla para proporcionar una realimentación fiable a los alumnos. El trabajo con estos profesionales ha permitido también valorar el uso de las herramientas desarrolladas para otras tareas, como es el caso de *PreLingua* para tareas de estimulación temprana.

VocalizaL2 ha contado con la evaluación realizada por personal y alumnado de las clases de Español del Vienna International School, donde jóvenes de diferentes nacionalidades llevan a cabo sus estudios de Español. Una evaluación más completa de todas las aplicaciones se está llevando a cabo actualmente recogiendo la opinión de los usuarios de la herramientas a través de la web; donde se les inquiere sobre la interacción y multimodalidad que ofrece la aplicación, así como sobre la utilidad que proporcionan las Tecnologías del Habla implementadas.

5.2. Evaluación Científica

Para la evaluación de todos los algoritmos requeridos en las aplicaciones se ha adquirido un corpus que contiene 3.192 palabras aisladas de un vocabulario de 57 palabras provenientes de 14 jóvenes locutores con trastornos en el habla y el lenguaje [7]. Para la validación de los resultados obtenidos se ha obtenido también un corpus con 9.576 palabras aisladas de locutores sin discapacidad en el mismo rango de edad como referencia del habla infantil y juvenil; y se ha realizado una anotación manual de los errores en la pronunciación por parte de los locutores con alteraciones lingüísticas con el objetivo de conocer

la afección exacta de cada locutor en su nivel articulatorio. Esta anotación ha demostrado que un 17,61 % de los fonemas en el corpus han sido eliminados o sustituidos, afectando a un 47.71 % de las palabras, dando cuenta de la gravedad del trastorno lingüístico de alguno de los locutores.

Con estos corpora se ha trabajado en varias líneas de trabajo: En términos de Reconocimiento Automático del Habla se ha estudiado el tema de la adaptación al locutor y modelado acústico probando cómo un sistema de Reconocimiento bien adaptado puede descartar con una exactitud del 88 % las palabras con varios fonemas incorrectamente pronunciados [5]. La adaptación al locutor obtiene una reducción en tasa de error del 61 % en la tarea de Reconocimiento, en un trabajo que no sólo aporta conocimiento sobre el funcionamiento de *Vocaliza*, *VocalizaL2* y *Cuéntame* sino que va en la dirección del desarrollo de sistemas de interacción oral con el entorno para discapacitados. La adaptación y modelado léxico también se ha trabajado para intentar introducir la información de cómo desvirtúan los locutores la pronunciación canónica de las palabras en el sistema de Reconocimiento. Por último, se ha trabajado en la evaluación de medidas de confianza para verificación de pronunciaciones, obteniéndose resultados prometedores utilizando algoritmos basados en determinar cuál es la secuencia de fonemas pronunciados más probable y compararla con la pronunciación canónica de la palabra [8] obteniéndose un 18.6 % en el Equal Error Rate.

6. CONCLUSIONES Y LÍNEAS FUTURAS

En este trabajo se ha presentado el marco de trabajo de *Comunica*, y los objetivos planteados en su creación. Se ha visto como el conjunto de herramientas desarrolladas abarcan todos los niveles en la adquisición del lenguaje (prelenguaje, articulación y pragmática) mediante el uso de Tecnologías del Habla. Por otro lado, se ha visto como utiliando canales directos de comunicación a través de Internet con los usuarios finales se puede realizar una evaluación y evolución de los sistemas más rápida y directa.

Como líneas de investigación futuras dentro de *Comunica* queda el desarrollo de nuevas herramientas como tutores de lectura, así como el constante estudio para la mejora de las ya existentes. La evaluación final de las aplicaciones mediante estudios y encuestas, así como el desarrollo de nuevos algoritmos para implementar dentro de las mismas es la otra gran línea de trabajo a llevar para conseguir los objetivos finales planteados en *Comunica*.

7. AGRADECIMIENTOS

Los autores quieren agradecer la colaboración para este trabajo de José Manuel Marcos y César Canalís del C.P.E.E. Alborada y Victoria Rodríguez del Vienna International School.

8. BIBLIOGRAFÍA

[1] A. Hatzis, P. Green, J. Carmichael, S. Cunningham, R. Palmer, M. Parker, y P. O'Neill, "An integrated toolkit deploying speech technology for computer based speech training with application to dysarthric speakers" in *Proceedings of the 8th Eurospeech-Interspeech*, Geneva, Switzerland, September 2003.

[2] J. Duchateau, L. Cleuren, H. Van Hamme, y P. Ghesquiere, "Automatic assessment of children's reading level" in *Proceedings of the 10th Eurospeech-Interspeech*, Antwerp, Belgium, September 2007.

[3] F.-C. Chou, "Ya-Ya language box - A portable device for English pronunciation training with speech recognition technologies" in *Proceedings of the 9th Eurospeech-Interspeech*, Lisbon, Portugal, September 2005, pp. 169–172.

[4] M.-S. Hawley, P. Green, P. Enderby, S. Cunningham, y R.-K. Moore, "Speech technology for e-inclusion of people with physical disabilities and disordered speech" in *Proceedings of the 9th Eurospeech-Interspeech*, Lisbon, Portugal, September 2005, pp. 445–448.

[5] C. Vaquero, O. Saz, E. Lleida, y W.-R. Rodríguez, "E-inclusion technologies for the speech handicapped" in *Proceedings of the 2008 ICASSP*, Las Vegas (NV), USA, April 2008, pp. 4509–4512.

[6] W.-R. Rodríguez, C. Vaquero, O. Saz, y E. Lleida, "Speech technology applied to children with speech disorders" in *Proceedings of the 4th International Conference on Biomedical Engineering*, Kuala Lumpur, Malaysia, June 2008, pp. 247–250.

[7] O. Saz, W. Rodríguez, E. Lleida, y C. Vaquero, "A novel corpus of children's impaired speech" in *Proceedings of the Workshop on Children, Computer and Interaction*, Chania, Greece, October 2008.

[8] S.-C. Yin, R. Rose, O. Saz, y E. Lleida, "Verifying pronunciation accuracy from dysarthric speech" in *Proceedings of the 10th ICSLP-Interspeech*, Brisbane, Australia, September 2008.

[9] A. Escartín, O. Saz, C. Vaquero, W.-R. Rodríguez, y E. Lleida, "*Comunica* framework Web site: http://www.vocaliza.es" 2008.

[10] B. Martínez, P. Peguero, J. Ezpeleta, J. Falcó, E. Lleida, J. Mínguez, y O. Saz, "Universidad y educación especial: Desarrollo y resultados de la colaboración entre el CPS y el CEE "Alborada" in *Proceedings of the III Congreso sobre Universidad y Discapacidad*, Zaragoza, Spain, November 2007.

DETECCIÓN DE CAMBIOS DE TOMA CON INFORMACIÓN DE CONTENIDO VISUAL Y AUDITIVO

Alejandro Abejon e Ismael Mateos[1]

ATVS (Biometric Recognition Group), C/ Francisco Tomas y Valiente 11,
Universidad Autonoma de Madrid, E28049 Madrid, Spain
{alejandro.abejon, ismael.mateos}@uam.es

RESUMEN

En este artículo se aborda la detección de cambios de toma en contenidos audiovisuales desde dos perspectivas complementarias, la información existente en la parte visual y la información de audio. Adicionalmente se proponen varios métodos de combinación de estas informaciones para obtener un sistema final robusto. La extracción de características de la parte visual se realiza a través de los descriptores *GoF/GoP* y *Scalable Color* de MPEG-7. Para la extracción de la información de la parte de audio se emplea BIC *(Bayesian Information Criterion)*. En la sección experimental se demuestra como la técnica de audio presenta una alta precisión en la detección, mientras que la técnica visual muestra un nivel de *recall* elevado. La combinación de ambas técnicas mejora sensiblemente el comportamiento final de la detección.

Palabras claves: detección de cambios de toma, MPEG-7, *GoF/GoP*, *Scalable Color,* BIC.

1. INTRODUCCIÓN

Durante los últimos años se ha experimentado un gran aumento en la cantidad de contenidos audiovisuales. Este aumento ha venido propiciado principalmente por dos causas: en primer lugar la constante aparición de nuevos dispositivos cada vez más baratos y con mejores prestaciones, en segundo lugar el desarrollo de estándares (compresión, codificación, etc.) que hacen cada vez más sencilla y eficiente la distribución de contenidos.

La anotación automática es un proceso importante a la hora de tratar con grandes bases de datos de contenidos audiovisuales (clasificación, almacenamiento, distribución, etc.). Este conjunto de técnicas, también conocido como servicios de valor añadido [1, 2], enriquecen la información multimedia de cara a procesos de búsqueda, indexación y auto-resúmenes.

Todos estos sistemas tiene en común la necesidad de aplicar una segmentación temporal de los contenidos para poder realizar una anotación automática de los mismos. Existen dos conceptos clave en este proceso: por un lado la segmentación debe ser lo mejor posible, por otro lado la velocidad debe ser elevada, ya que es necesario que la segmentación sea más rápida que tiempo real.

Una de las formas más comunes de segmentar contenido audiovisual consiste en la detección de cambios de toma. Esta tarea ha sido enfocada tradicionalmente desde el punto de vista de la señal de video [3]. Aunque hay algunos autores como Shu-Ching Chen en 2002 y Yingying Zhu y Dongru Zhou en el año 2003 [4, 5], que emplearon una combinación de la información obtenida del audio y la imagen de forma que el sistema contara con una mayor robustez. El problema principal de estas técnicas es la combinación de la información de audio y video [6]. Se han propuesto varias soluciones para dicha combinación de información [7, 8] sin llegar a obtener un resultado plenamente satisfactorio.

La estructura que se va a seguir en este artículo es la siguiente. En primer lugar, en la sección 2 se realiza un recorrido por el estado del arte en técnicas de detección de cambios de toma con información visual, información de audio y posibles combinaciones de ambas. Posteriormente, en la sección 3 se detalla el sistema implementado, la técnica empleada a nivel visual, a nivel de audio y las combinaciones propuestas. Los resultados obtenidos se muestran en la sección 4. Por último, la sección 5 contiene las conclusiones extraídas de la realización de este trabajo.

2. ESTADO DEL ARTE

2.1. Detección a nivel visual

La detección de cambios de toma basada en información visual es la aproximación más utilizada. Podemos distinguir dos tipos de cambios de toma: los cambios abruptos en los que el cambio se da de un frame al siguiente; por otro lado están los cambios graduales cuya duración es de varios frames. Los cambios abruptos son los más fáciles de detectar, en los últimos años se han propuesto varios algoritmos que gozan de una gran precisión [9, 10]. Por el contrario, la complejidad en la detección de cambios graduales es mayor. En la literatura podemos distinguir dos vertientes: la que afronta cierto tipo de cambios específicos (ej. disoluciones) con resultados aceptables [11], por otro lado, podemos encontrar técnicas que tratan de construir modelos generales que engloben cualquier tipo de cambio gradual posible [12, 13]. Esta última aproximación presenta una fiabilidad menor que la anterior.

El histograma de color es una de las variables más comúnmente empleadas en sistemas visuales de detección de cambios de toma. Este hecho es debido a que un cambio de toma suele llevar consigo un cambio en las distribuciones de color de la imagen [14]. Otro factor que se observa cuando hay un cambio de toma es el incremento del número y módulo de los vectores de movimiento [15]. Codificaciones como MPEG-2 o H.264 se basan en la similitud de imágenes sucesivas para codificar bloques en función de bloques de imágenes adyacentes. Un cambio de toma llevará consigo un

[1] Este trabajo ha sido apoyado por el Ministerio de Ciencia y Educación, TEC2006-13170-C02-01. Los autores desean mostrar su agradecimiento al grupo GTI de la Escuela Politécnica Superior (UAM) por su apoyo.

incremento en el número de vectores de movimiento, debido al menor parecido entre imágenes.

En el proceso de la detección del cambio de toma podemos distinguir dos fases: medida de disparidad entre parejas de frames, que pueden estar consecutivas o distanciadas y toma de decisión de si esa disparidad entre frames es o no un cambio de toma.

2.2. Detección a nivel de audio

La tarea de detección de cambios en la señal de voz está englobada dentro de lo que se conoce como *anotación de un fichero de audio*. El objetivo de estas técnicas es enriquecer cualquier contenido de audio, dotándole de una cierta información acerca de locutores, música, silencios, ruido, etc. que en el fichero aparezcan.

Existen distintos niveles a la hora de anotar un fichero de audio, desde el más sencillo que consistiría simplemente en indicar los tramos del fichero de voz y no voz (dentro de la categoría de no voz se incluye música, silencio, ruido, etc.), hasta los más elaborados en los que se indican los tramos donde está presente el mismo locutor, tramos con música, silencio, etc.

Una hipótesis de partida podría ser que cambios de toma en la señal visual llevan asociados cambios en la señal de audio. Por tanto el objetivo será detectar variaciones en la señal sonora: como por ejemplo el paso de voz a música. Otra alternativa sería tener en cuenta los cambios de audio como puntos clave a la hora de segmentar, sin necesidad de que lleven asociados cambios de toma.

Para llevar a cabo la detección de cambios de toma existen principalmente dos fases: la primera de ellas consistente en la detección de voz y silencio, la segunda está relacionada con la detección de tramos con música, ruido, diferentes locutores, etc. dentro de los segmentos de voz. Para la primera de las etapas se suelen emplear los cruces por cero de la señal aunque la aproximación más habitual está basada en el cálculo de la energía. Para llevar a cabo la segunda etapa existen dos técnicas principalmente, ambas hacen uso de ventanas que se van desplazando a lo largo del fichero de audio.

El criterio de información bayesiano *(BIC)* [16] es una de las dos aproximaciones para la detección de distintos tramos dentro de segmentos de audio. Esta técnica busca puntos de cambio dentro de una ventana, para ello se comprueba si los datos de la ventana se modelan mejor con una única distribución (no hay punto de cambio) o con dos distribuciones (punto de cambio). Si se encuentra un punto de cambio se resetea la ventana y se reinicia la búsqueda a partir de ese punto. Si por el contrario no se encuentra punto de cambio se incrementa la ventana y se vuelve a realizar la búsqueda. La búsqueda completa a lo largo del fichero es muy costosa computacinalmente, del orden de N^2 siendo N el número de muestras del fichero, por tanto la mayor parte de los sistemas emplean versiones simplificadas de este algoritmo.

La segunda técnica empleada se propuso en 1997 [17], consiste en usar ventanas de una longitud fija (normalmente entre 2 y 5 segundos) que serán representadas por una gausiana. Posteriormente se calculan las distancias entre ventanas. Viendo si estas distancias superan o no un umbral seremos capaces de encontrar los puntos de cambio. La longitud de las ventanas limita la detección de cambios de corta duración.

En los últimos tiempos han aparecido otras técnicas de mayor complejidad aunque su comportamiento es comparable al de las técnicas anteriores. Dichas técnicas utilizan modelos de mezclas de gausianas *(GMM o Gaussian Mixture Models)* o modelos ocultos de Markov *(HMM o Hidden Markov Models)* para modelar música, locutores y ruido ambiental. El gran inconveniente de estas técnicas es que son supervisadas y para su entrenamiento se necesita una gran cantidad de datos etiquetados que modelen la generalidad de aquello que se quiere detectar.

2.3. Combinación de información de audio y video

La combinación de la información procedente del audio y del vídeo es una de las labores más complejas en un sistema de detección de cambios de toma. Aunque la información de audio va ligada a la información visual no tienen porque coincidir con los cambios de toma reales. Puede darse el caso de que se produzca un cambio en el audio, como por ejemplo la intervención de un nuevo locutor, que no lleve consigo un cambio de toma y viceversa.

Son pocos los trabajos encontrados en la literatura que traten este problema. En [4] se hace una primera división basada en la señal de audio que divide la señal en silencio, voz y música; dentro de la parte de voz se hace distinción entre locutores. Por otro lado se realiza la segmentación basada en el contenido visual, para posteriormente combinar ambas informaciones dando un mayor peso a la señal de audio.

Existen otras posibilidades a la hora de realizar la fusión entre las informaciones de audio y vídeo. Desde los trabajos en los que no se realiza ninguna fusión [8], pasando por trabajos en los que se realiza una segmentación a nivel de audio y después se usa la información visual para comprobar si es correcto [18], hasta variantes de la aproximación anterior donde se confirman los cambios extraídos del vídeo con el audio [5].

3. SISTEMA IMPLEMENTADO

3.1. Segmentación a nivel visual

La técnica implementada realiza una detección de tomas mediante un modelo general, es decir, lo que buscamos es que dicho modelo detecte tanto los cambios de toma abruptos como los graduales. Para ello nos basaremos en dos de los descriptores definidos por MPEG-7: *GoF/GoP* y *Scalable color* (más información en: ISO/IEC 15938-3 6.5, ISO/IEC 15938-3 6.8, ISO/IEC TR 15938-8). El objetivo original de estos descriptores es la búsqueda de similitudes entre vídeos y/o imágenes.

Este sistema es equivalente a realizar una detección de cambio de toma por histograma de color, ya que los coeficientes sobre los que se aplican las medidas de dispersión miden la distribución de color en cada frame analizado.

Las medidas de dispersión evaluarán el grado de similitud entre frames, dichas medidas se aplican sobre los coeficientes resultantes del descriptor. En lugar de calcular los coeficientes para todos las frames se realiza para una de cada 25 frames, es decir un frame por segundo. Con este procedimiento conseguimos dos objetivos, por una lado reducir la carga computacional y por otro la detección de cambios graduales. Como medidas de dispersión se emplearon 3 de las que mejor comportamiento presentaban en [14].

$$\delta_1 = \sum\sum \left| c_a(x,y) - c_b(x,y) \right|^2 \quad (1)$$

$$\delta_2 = 1 - \min\left(LR, \frac{1}{LR} \right) \quad (2)$$

$$LR = \frac{\left[\frac{S_a^2 + S_b^2}{2} + \left(\frac{c_a - c_b}{2} \right)^2 \right]^2}{S_a^2 + S_b^2} \quad (3)$$

donde *s* es la dispersión y *c* es la media de cada trama.

La tercera de las medidas de dispersión es una combinación de las dos anteriores, propuesta también en [14]. Consiste en aplicar un filtro de mediana de longitud tres, a las dos medidas de dispersión anteriores y restarlas a las originales. Esta técnica proporciona una mayor robustez frente a flashes y cambios de iluminación. Una vez tenemos calculados el vector diferencia realizamos el calculo del modulo de dicho vector.

$$\lambda = \left\| (\delta_1, \delta_2) - mediana(\delta_1, \delta_2) \right\| \quad (4)$$

Nuestra propuesta consiste en establecer el umbral de forma dinámica calculando el 20% del margen dinámico de cada una de las tres medidas indicadas. Siempre que las tres componentes superen este umbral se decidirá que existe un cambio de toma. Esta forma de establecer el umbral ofrece una mayor robustez, pero tiene el inconveniente de tener que recorrer el fichero antes de la fase de toma de decisión.

3.2. Segmentación a nivel de audio

El proceso seguido para obtener la información de audio consta de dos pasos básicamente: *i)* una vez separado el contenido de audio del vídeo se realiza una parametrización MFCC *(Mel Frequency Cepstral Coefficients)* de 13 coeficientes; *ii)* a continuación se aplica el algoritmo de detección de cambios, BIC (véase sección 2.2).

3.3. Combinación de la información visual y de audio

Antes de llevar a cabo la combinación de la información obtenida del audio y del vídeo se realiza un filtrado de los resultados para subsanar en lo posible errores en la detección. Este filtrado consiste en eliminar cambios de toma sucesivos que tengan entre sí menos de un segundo de duración. Se presupone que un cambio de toma por muy rápido que sea debe durar más de un segundo, por lo que los cambios de este tipo que resulten de nuestros algoritmos serán eliminados.

Una vez realizado este filtrado estamos en disposición de combinar la información de los dos métodos para obtener un resultado global más robusto. Como se observa en el estado del arte, esta combinación no resulta sencilla. En nuestro caso se propone dos tipos de combinaciones distintas.

En primer lugar, la más restrictiva de ellas (combinación &), consiste en indicar sólo aquellos cambios de toma en los que la información obtenida del audio y del vídeo coincida. Con el fin de dotar de mayor flexibilidad al sistema introduciremos un cierto margen de error programable que para las pruebas que mostraremos en la sección 4 será de 3 segundos. El sistema dictaminará que hay un cambio de toma siempre que vídeo y audio coincidan con un margen de error de más menos 3 segundos.

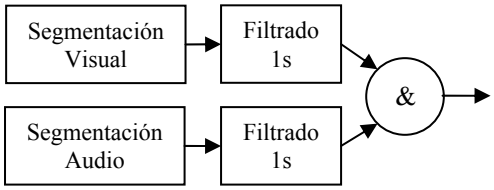


Figura 1. *Esquema combinación restrictiva segmentaciones, comb. &*

En segundo lugar emplearemos como método de combinación algo bastante sencillo (combinación +), la agrupación de los cambios indicados por cada uno de los métodos por separado. Una vez hecho esto se realizará el proceso de filtrado mencionado anteriormente, pero con menos de tres segundos de duración.

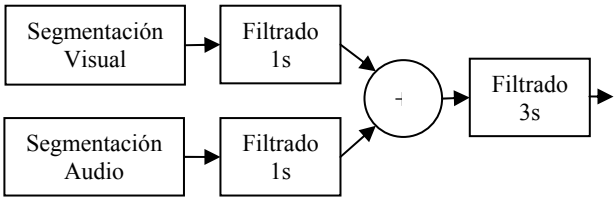


Figura 2. *Esquema combinación concatenada segmentaciones, comb. +*

4. RESULTADOS

El vídeo de ejemplo seleccionado cosiste en 5 minutos (7500 frames), extraídos del programa *Informe Semanal*. Este tipo de vídeo se puede considerar dentro de la categoría de informativos, una de las más estudiadas a lo largo de los últimos años. El video a nivel visual contiene 7 cambios graduales y 22 cambios abruptos. La señal de audio consta de 17 cambios, no todos ellos coincidentes con el vídeo y por tanto no detectables por técnicas basadas en audio.

La Tabla 1 muestra los resultados obtenidos con la detección visual por un lado y la detección a través de audio por otro. Además se muestran estos resultados tras el filtrado y las combinaciones de ambas técnicas propuestas.

	OK	NO	FP	R (%)	P (%)
Visual	10	19	19	34	34
Visual filtrado	9	20	7	31	56
Audio	7	22	1	24	88
Audio filtrado	7	22	1	24	88
Comb. &	3	27	1	11	75
Comb. +	11	18	10	38	52

Tabla 1. *Resultados de detección de cambios de tomas; OK cambios correctos, NO cambios no detectados, FP falsos positivos, R recall, P precesión*

A la vista de los resultados vemos como el filtrado aumenta la precisión del sistema, sobre todo en la parte visual. En la parte de audio no se observa cambio alguno ya que no se detectaban cambios tan seguidos. Vale la pena destacar que cuando la detección es solo con audio, la precisión que se obtiene es de las más altas, lo que nos indica que los cambios detectados con el audio tienen una alta probabilidad de ser correctos. Además todas medidas están calculadas sobre el número total de cambios a nivel visual, por tanto la técnica a nivel de audio está en clara desventaja.

También se observa que la primera combinación adoptada (comb. &), aquella que implica que el cambio se dé tanto en la parte visual como en la de audio es demasiado restrictiva, lo cual provoca que a penas se detecten cambios de toma. Por el contrario cuando simplemente unimos los cambios detectados en el audio y en la imagen (comb, +) se observa que la cantidad de cambios de toma correctos es elevada. Las Figuras 3 y 4 muestran los cambios de toma detectados con las dos combinaciones propuestas.

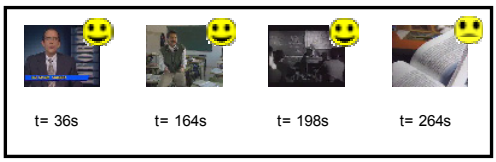


Figura 3. *Resultados combinación de segmentaciones &*

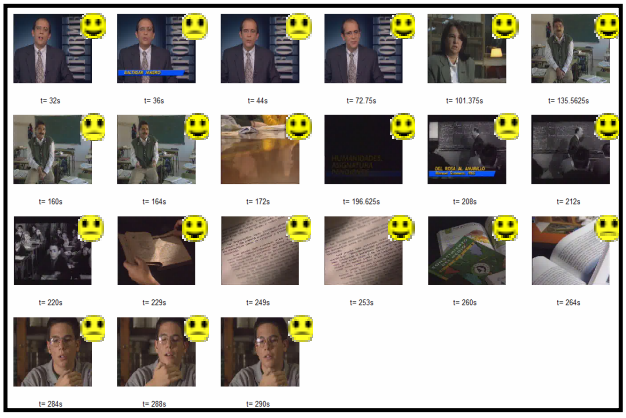


Figura 4. *Resultados combinación de segmentaciones +*

5. CONCLUSIONES

En este artículo se ha abordado uno de los problemas más habituales en la anotación de contenidos audiovisuales: la detección de cambios de toma. Para llevar a cabo esta labor se emplean dos tipos de informaciones: *i)* la información de audio y *ii)* la información de vídeo. Ambas informaciones son combinadas con el objetivo de obtener un sistema global más robusto. La información de cambios de audio se extrae mediante BIC. La información del contenido visual se obtiene siguiendo las indicaciones propuestas por MPEG-7 en dos de sus descriptores: *GoF/GoP* y *Scalable Color*.

La sección de resultados muestra como la información de audio es importante y ayuda a la mejora del rendimiento del sistema global. La precisión obtenida a través del contenido de audio es de las más elevadas. Este hecho arroja resultados esperanzadores y anima a la búsqueda de métodos de combinación más eficientes. Por otro lado, una alternativa a la detección de cambios de toma de cara a la realización de auto-resúmenes sería la segmentación del contenido basada únicamente en la información del audio. La señal de audio contiene información suficiente como para poder llevar a cabo esta tarea sin apoyarse en la información visual.

Dos aspectos importantes de la implementación son en primer lugar la velocidad del análisis, siendo para el audio muy eficiente. En segundo lugar, la parte visual está basada en descriptores escalables, este hecho hace que tanto la velocidad en la extracción de coeficientes como la precisión sean configurables y adaptables a las necesidades de cada sistema.

6. BIBLIOGRAFÍA

[1] MERL Video Summarization for PVRs (2008) http://www.merl.com/projects/VideoSummarization [Online].
[2] Lire (Lucene Image REtrieval) (Junio 2008) http://www.semanticmetadata.net [Online].
[3] Koprinska and S. Carrato, "Temporal video segmentation, a survey," *Signal Processing. Image Commun*., vol. 16, no. 5, pp. 450-477, Jan. 2001.
[4] S. Chen, et al., "Scene Change Detection by Audio and Viedo Clues", *IEEE Transactions on Speech and Audio Processing*, Vol. II, pp. 365- 368, 2002.
[5] Y. Zhu and D. Zhou, "Scene Change Detection Based on Audio and Video Content Analysis", *ICCIMA 2003.*
[6] A. Yoshitaka, and M. Miyake, "Scene Detection by Audio-Visual Features," *IEEE International Conference on Mulatimedia and Expo (ICME01)*, pp. 49-52, 2001.
[7] H. Sundaram and S.-F. Chang, "Video Scene Segmentation Using Video and Audio Features," *IEEE International Conference on Multimedia and Expo (ICME00)*, pp. 1145-1148, 2000.
[8] T. Muramoto and M. Sugiyama, "Visual and Audio Segmentation for video streams," *IEEE International Conference on Mulatimedia and Expo (ICME00),* pp 1547-1550, 2000.
[9] L. Qianlei, et al., "Twi-difference Algorithm and Pixel-matching Twi-difference Algorithm for Video Abrupt Shot Change Detection",*Journal of Image and Graphics* A, Vol. 2, No. 2, pp. 161-168, 2003.
[10] S. Gong and Y. Fan, "Video abrupt shot change detection based on relation of the partial interframe differences," *Machine Learning and Cybernetics*,. Volume 9, Page(s):5255-5260 Vol. 9, Aug. 2005.
[11] C. Su et al., "A motion-tolerant dissolve detection algorithm", *IEEE Transactions Multimedia,* Volume 7, Issue 6, Page(s):1106 – 1113 Dec. 2005.
[12] W. Xiong and J. C. M. Lee, "Efficient scene change detection and camera motion annotation for video classification," *Comput. Vis. Image Understanding*, vol. 71, no. 2, pp. 166–181, Aug. 1998.
[13] P. Bouthemy, et al., "A unified approach to shot change detection and camera motion characterization," *IEEE Trans. Circuits Syst. Video Technol*., vol. 9, pp. 1030–1044, 1999.
[14] J. Bescós, "Real-Time Shot Change Detection Over Online MPEG-2 Vedeo", *IEEE Trans. Circuits Syst. Video Technol* Vol 14 No 4 pp 475-484 April 2004.
[15] M. Zhi and A. Cai, "Shot change detection with adaptive thresholds",*VLSI Design and Video Technology*, Page(s):147 – 149 May 2005.
[16] S. S. Chen and P. S. Gopalakrishnam, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA, 1998, pp. 127–132.
[17] M. A. Siegler, et al.,"Automatic segmentation, classification and clustering of broadcast news," in *Proc. DARPA Speech Recognition Workshop*, Chantilly, VA, Feb. 1997, pp. 97–99.
[18] H. Jiang, et. al., "Video Segmentation with the assistance of audio content analysis," *IEEE International Conference on Mulatimedia and Expo*, pp. 1507-1510, 2000.

FEATURE SELECTION VS. FEATURE TRANSFORMATION IN REDUCING DIMENSIONALITY FOR SPEAKER RECOGNITION

Maider Zamalloa[1,2]*, L. J. Rodríguez-Fuentes*[1]*, Mikel Peñagarikano*[1]*, Germán Bordel*[1]*, Juan P. Uribe*[2]

(1) Grupo de Trabajo en Tecnologías del Software, DEE, ZTF/FCT
Universidad del País Vasco / Euskal Herriko Unibertsitatea
Barrio Sarriena s/n, 48940 Leioa, SPAIN

(2) Ikerlan – Technological Research Centre
Paseo J.M. Arizmendiarrieta 2, 20500 Arrasate-Mondragón, SPAIN

e-mail: maider.zamalloa@ehu.es

ABSTRACT

Mel-Frequency Cepstral Coefficients and their derivatives are commonly used as acoustic features for speaker recognition. Reducing the dimensionality of the feature set leads to more robust estimates of the model parameters, and speeds up the classification task, which is crucial for real-time speaker recognition applications running on low-resource devices. In this paper, a feature selection procedure based on genetic algorithms (GA) is compared to two well-known dimensionality reduction techniques based on linear transforms, namely Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Evaluation is carried out for two speech databases, containing laboratory read speech and telephone spontaneous speech, and applying a state-of-the-art speaker recognition system. Results with GA-based feature selection suggest that dynamic features are less discriminant than static ones, since the low-size optimal subsets found by the GA did not include dynamic features. GA-based feature selection outperformed PCA and LDA when dealing with clean speech, but not for telephone speech, probably due to some noise compensation implicit in linear transforms, which cannot be accomplished just by selecting a subset of features.

1. INTRODUCTION

Mel-Frequency Cepstral Coefficients (MFCC) are commonly used as acoustic features for speaker recognition, since they convey not only the frequency distribution identifying sounds, but also information related to the glottal source and the vocal tract shape and length, which are speaker specific features. Additionally, it has been shown that dynamic information improves the performance of recognizers, so first and second derivatives are appended to MFCC. The resulting feature vector ranges from 30 to 50 dimensions. However, for applications requiring real-time operation on low-resource devices, high dimensional feature vectors do not seem suitable and some kind of dimensionality reduction must be applied, maybe at the cost of performance degradation.

A simple approach to dimensionality reduction is feature selection, which consists of determining an optimal subset of K features by exhaustively exploring all the possible combinations of D features. Most feature selection procedures use the classification error as the evaluation function. This makes exhaustive search computationally infeasible in practice, even for moderate values of D. The simplest method consists of evaluating the D features individually and selecting the K most discriminant ones, but it does not take into account dependencies among features. So a number of suboptimal heuristic search techniques have been proposed in the literature, which essentially trade-off the optimality of the selected subset for computational efficiency [1].

Genetic Algorithms (GA) suitably fit this kind of complex optimization problems. A major advantage of GA over other heuristic search techniques is that they do not rely on any assumption about the properties of the evaluation function. Multiobjective evaluation functions (e.g. combining the accuracy and the cost of classification) can be defined and used in a natural way [2]. GA can easily encode decisions (about selecting or not selecting features) as sequences of boolean values, allow to smartly explore the feature space by retaining those decisions that benefit the classification task, and simultaneously avoid local optima due to their intrinsic randomness. GA have been recently applied to feature extraction [3], feature selection [4] and feature weighting [5] in speaker recognition.

Alternatively, the problem of dimensionality reduction can be formulated as a linear transform which projects feature vectors on a transformed subspace defined by relevant directions. Among others, two well-known dimensionality reduction techniques, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), fall into this category.

This work has been jointly funded by the Government of the Basque Country, under projects S-PE06UN48, S-PE07UN43, S-PE06IK01 and S-PE07IK03, and the University of the Basque Country, under project EHU06/96.

In this paper, a feature selection procedure based on a GA-driven search is compared to PCA and LDA in a speaker recognition task. GA-based feature selection projects the original D-dimensional feature space into a reduced K-dimensional subspace by just selecting K features. PCA and LDA not only reduce but also scale and rotate the original feature space, through a transformation matrix A which optimizes a given criterion on the training data. From this point of view, PCA and LDA generalize feature selection, but the criteria applied to compute A (the highest variance in PCA, and the highest ratio of between to within class variances in LDA) do not match the criterion applied in evaluation (the speaker recognition rate). This is the strong point of GA, since feature selection is performed in order to maximize the speaker recognition rate on an independent development corpus.

2. FEATURE SELECTION USING GENETIC ALGORITHMS

The GA-driven selection process begins by fixing the target size K of the reduced feature subspace. Then, an initial population of candidate solutions (K-feature subsets) is randomly generated. In this work, each candidate is represented by a D-dimensional vector of positive integers $R = \{r_1, r_2, \ldots, r_D\}$, ranging from 0 to 255 (8 bits), the K highest values determining what features are selected. To evaluate the K-feature subset $= \{f_1, f_2, \ldots, f_K\}$, the following steps are carried out (1) the acoustic vectors of the whole speech database are reduced to the components enumerated in ; (2) speaker models are estimated using the training corpus; (3) utterances in the development corpus are classified by applying the speaker models; and (4) the speaker recognition accuracy obtained for the development corpus is used to evaluate .

At the end of each iteration/generation, after all the K-feature subsets in the population are evaluated, some of them (usually the fittest ones), are selected, mixed and mutated in order to get the population for the next generation. Mutation is used to introduce small variations that help decrease the chances of getting local optima. On the other hand, *elitism* (copying some of the fittest individuals to the next generation) is applied to guarantee that the fitness function increases monotonically with successive generations. If that increase is smaller than a given threshold, or a maximum number of generations is reached, the algorithm stops and the optimal K-feature subset $\hat{} = \{\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_K\}$ is returned.

3. EXPERIMENTAL SETUP

3.1. Acoustic features

In this work, MFCC, energy and their first and second derivatives were taken as acoustic features. Speech was analysed in 25-millisecond frames, at intervals of 10 milliseconds. A Hamming window was applied and an FFT computed, whose length depended on the sampling frequency: 256 points for signals sampled at 8 khz and 512 points for signals sampled at 16 kHz. FFT amplitudes were then averaged in 20 (8 kHz) or 24 (16 kHz) overlapped triangular filters, with central frequencies and bandwidths defined according to the Mel scale. A Discrete Cosine Transform was finally applied to the logarithm of the filter amplitudes, obtaining 10 (8 kHz) or 12 (16 kHz) Mel-Frequency Cepstral Coefficients (MFCC). To increase robustness against channel distortion, Cepstral Mean Normalization was applied on an utterance-by-utterance basis. The first and second derivatives of the MFCC, the frame energy (E) and its first and second derivatives were also computed, thus yielding a 33-dimensional (8 khz) or a 39-dimensional (16 kHz) feature vector.

3.2. Speaker models

Most speaker recognition systems represent the distribution of feature vectors extracted from a speaker's speech by a linear combination of M multivariate Gaussian densities, known as *Gaussian Mixture Model* (GMM) [6], whose parameters are estimated from speaker samples by applying the *Maximum Likelihood* (ML) criterion. In this work, speaker recognition was performed using 32-mixture diagonal covariance GMMs as speaker models.

3.3. Speech databases

Two speech databases were used in this work: *Albayzín* (a phonetically balanced read speech database in Spanish, recorded at 16 KHz in laboratory conditions, containing 204 speakers) and *Dihana* (a spontaneous task-specific speech corpus in Spanish, recorded at 8 kHz through telephone lines, containing 225 speakers), each partitioned in three disjoint datasets: (1) the training set, used to estimate the speaker models and the PCA and LDA transforms; (2) the development set, used by the GA to compute the fitness function; and (3) the test set, used to evaluate the performance of the optimal K-feature subsets provided by GA, PCA and LDA.

3.4. GA, PCA and LDA Implementations

The well-known *Simple Genetic Algorithm* (SGA) [7], implemented by means of ECJ [8], was applied to search for the optimal feature set. Offspring was bred by first selecting and then mixing two parents in the current population. The first parent was selected according to the fitness-proportional criterion, by picking the fittest from seven randomly chosen individuals. The second parent was chosen the same way, but only from two randomly chosen individuals, to allow diversity and avoid local optima. One-point crossover was applied and the mutation probability was set to 0.01. Finally, the simplest case of elitism was applied by keeping the fittest individual for the next generation. The maximum number of generations was fixed to 40. A public domain software developed at the MIT Lincoln Laboratory, *LNKnet* [9], was used to perform PCA. Regarding LDA, a custom implementation was developed in Java.

Table 1. Optimal feature sets found by the GA in speaker recognition experiments for Albayzín and Dihana, for K = 30, 20, 13, 12, 11, 10, 8 and 6. Selected features are marked with a star ($\star$). Cells containing a dash (–) correspond to features not computed for Dihana.

		E	c01	c02	c03	c04	c05	c06	c07	c08	c09	c10	c11	c12	dE	d01	d02	d03	d04	d05	d06	d07	d08	d09	d10	d11	d12	ddE	dd01	dd02	dd03	dd04	dd05	dd06	dd07	dd08	dd09	dd10	dd11	dd12
Albayzín	30	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆				⋆	⋆		⋆			⋆	⋆	⋆	⋆	⋆			⋆		⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆
	20	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆						⋆				⋆	⋆							⋆		⋆	⋆				⋆	
	13	⋆	⋆	⋆	⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆																			⋆							
	12	⋆	⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆																										
	11	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆																										
	10	⋆	⋆	⋆	⋆	⋆		⋆			⋆	⋆	⋆	⋆																										
	8	⋆		⋆		⋆		⋆		⋆	⋆		⋆	⋆																										
	6	⋆				⋆	⋆			⋆		⋆	⋆																										
Dihana	30	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–	⋆			⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–
	20	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–			⋆	⋆	⋆	⋆	⋆	⋆		⋆	⋆	–	–										⋆		–	–
	13		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–						⋆		⋆	⋆			–	–												–	–
	12		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–				⋆				⋆				–	–												–	–
	11		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–								⋆				–	–												–	–
	10		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–												–	–												–	–
	8		⋆	⋆	⋆		⋆	⋆	⋆	⋆		⋆	–	–												–	–												–	–
	6		⋆				⋆	⋆	⋆	⋆		⋆	–	–												–	–												–	–

4. RESULTS

4.1. Feature selection using GA

The optimal K-dimensional feature sets provided by the GA for Albayzín and Dihana in speaker recognition experiments are shown in Table 1. The terms *cXX* and *E* correspond to the MFCC and the frame energy, and *dXX*/*dE* and *ddXX*/*ddE* to their first and second derivatives, respectively. As noted above, the computation of MFCC depends on the sampling frequency, so the dimension (D) of the full feature vectors is 39 (12 MFCC + energy + first and second derivatives) for Albayzín, and 33 (10 MFCC + energy + first and second derivatives) for Dihana.

Focusing on the results for Albayzín, note that the presence of a feature in the optimal subset of size K does not imply that the same feature will be present in the optimal subsets for larger values of K. For instance, *c05* appears in the optimal subset for $K = 6$, but not for $K = 8$ and $K = 10$. This suggests that optimal subspaces cannot be determined in an incremental way, by sequentially reducing its size. In other words, it seems that an exhaustive search is needed which explores all the feature combinations. Note also that the GA-optimal subsets for $K \leq 12$ consist of a number of MFCC plus the frame energy. Three of them, *E*, *c04* and *c11* are always selected by the GA; three other, *c02*, *c06* and *c09*, are selected always but for the smallest subset ($K = 6$). This suggests that static features are more relevant for speaker recognition than dynamic features.

The optimal subsets found by the GA for Dihana show an almost perfect sequential behaviour, contrasting with that obtained for Albayzín. Only two cases of non-sequential behaviour are found: *d08*, from $K = 20$ to $K = 13$; and *d03*, from $K = 13$ to $K = 12$. It would be worth to investigate this issue more deeply, since sequential optimization is much faster that an exhaustive search. On the other hand, the optimal sets for low values of K ($K \leq 10$) are composed exclusively of MFCC (the frame energy and dynamic features do not appear). Again, it seems that MFCC convey more relevant information about speaker characteristics than their derivatives. Interestingly, the frame energy does not appear in any of the optimal sets for $K \leq 13$, suggesting that this feature is not as robust for telephone spontaneous speech as for laboratory read speech.

4.2. Comparing GA to PCA and LDA

GA-based feature selection, PCA and LDA were tested in speaker recognition experiments over Albayzín and Dihana. First, D-dimensional feature vectors were transformed into reduced K-dimensional feature vectors, according to the optimal subset/transformation given by GA, PCA or LDA, then speaker models were estimated on the training corpus and finally speaker recognition experiments were carried out on the test corpus. Results are shown in Table 2.

Confidence intervals are shown to allow significant performance comparisons among different feature sets. This deserves a brief explanation. Model estimations start from random initializations. Preliminary experimentation showed that, fixed the set of features and the training database, random initializations led to slightly different model parameters after convergence, and therefore slight differences in speaker recognition performance were observed. This intrinsic uncertainty can be taken into account in performance comparisons by computing the confidence interval of an average error rate. It is assumed that the underlying distribution of error rates is Gaussian. So,

Table 2. Average error rates and 95% confidence intervals in speaker recognition experiments on test data for Albayzín and Dihana, using the optimal K-dimensional feature sets provided by GA, PCA nd LDA, for K = 6, 8, 10, 11, 12, 13, 20 and 30.

K	Albayzín			Dihana		
	GA	PCA	LDA	GA	PCA	LDA
6	**5.71±0.09**	14.37±0.15	8.11±0.14	34.23±0.16	**33.23±0.12**	35.52±0.14
8	**1.81±0.09**	5.86±0.12	2.64±0.09	**23.90±0.14**	24.19±0.13	25.06±0.13
10	**0.94±0.04**	2.73±0.12	1.21±0.06	19.70±0.12	20.67±0.12	**19.43±0.12**
11	**0.35±0.04**	1.61±0.07	1.12±0.06	19.32±0.14	20.27±0.13	**18.10±0.13**
12	**0.30±0.04**	0.94±0.06	0.79±0.06	19.27±0.14	19.75±0.16	**18.18±0.12**
13	**0.33±0.05**	0.56±0.05	0.88±0.04	19.12±0.11	19.63±0.10	**17.66±0.10**
20	**0.16±0.02**	0.19±0.02	0.39±0.04	19.99±0.11	17.61±0.13	**17.24±0.11**
30	**0.13±0.02**	0.15±0.03	0.33±0.04	19.10±0.14	**15.97±0.15**	18.17±0.12

in order to compute the average error rate and the 95% confidence interval, the whole process of training speaker models and carrying out speaker recognition experiments was repeated 20 times for each feature set.

In the case of Albayzín, neither PCA nor LDA outperformed GA. PCA yielded lower error rates than LDA for $K > 12$. For $K \leq 12$, LDA outperformed PCA. However, the error rates are too low and the differences in performance too small for these conclusions to be statistically significant.

Error rates for Dihana were much higher, because it was recorded through telephone channels in an office environment and a large part of it consists of spontaneous speech. The presence of channel and environment noise in Dihana makes PCA and LDA more suitable than GA, because feature selection cannot compensate for noise, whereas linear transforms can do it to a certain extent. This may explain why either PCA or LDA outperformed GA in all cases but for $K = 8$. LDA was the best approach in most cases (for K = 10, 11, 12, 13 and 20). GA was the second best approach for K = 6, 10, 11, 12 and 13. Finally, the lowest error rate (15.97%) was obtained for $K = 30$ using PCA.

In summary, GA-based feature selection seems to be competitive only when dealing with clean speech, though it performs quite well even for noisy speech when the target K is small. Authors that argue against GA optimization say that it is too costly, since it requires iteratively evaluating candidate solutions in classification experiments over a development dataset. It must be noted, however, that GA optimization is done off-line, so the computational cost is not an issue in practice. Moreover, during recognition, feature selection is less costly than feature transformation.

5. CONCLUSIONS

Feature selection based on GA suggests that static features are more discriminant than dynamic features for speaker recognition applications. In the case of telephone speech, the smallest feature subsets ($K \leq 13$) did not include the frame energy, which reveals that channel and/or environment noise is distorting the information it conveys. Summarizing, if a reduced set of features had to be selected (due to storage or computational restrictions), MFCC would be the best choice, augmented with the frame energy when dealing with clean-laboratory speech.

GA outperformed PCA and LDA only when dealing with clean speech, whereas PCA and LDA outperformed GA in most cases when dealing with telephone speech, probably due to some noise compensation implicit in linear transforms, which cannot be accomplished just by selecting a subset of features. In any case, since applying a linear transform is more costly than selecting a subset of features, depending on the target K, the gain in performance might not be worth the additional effort.

6. REFERENCES

[1] A. K. Jain, R. P. W. Duin, y J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, January 2000.

[2] L. S. Oliveira, R. Sabourin, F. Bortolozzi, y C. Y. Suen, "A Methodology for Feature Selection Using Multiobjective Genetic Algorithms for Handwritten Digit String Recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, no. 6, pp. 903–929, 2003.

[3] C. Charbuillet, B. Gas, M. Chetouani, y J. L. Zarader, "Filter Bank Design for Speaker Diarization Based on Genetic Algorithms," in *Proceedings of the IEEE ICASSP'06*, Toulouse, France, 2006.

[4] M. Zamalloa, G. Bordel, L. J. Rodriguez, y M. Peñagarikano, "Feature Selection Based on Genetic Algorithms for Speaker Recognition," in *IEEE Speaker Odyssey: The Speaker and Language Recognition Workshop*, Puerto Rico, June 2006, pp. 1–8.

[5] M. Zamalloa, G. Bordel, L. J. Rodríguez, M. Peñagarikano, y J. P. Uribe, "Using Genetic Algorithms to Weight Acoustic Features for Speaker Recognition," in *Proceedings of the ICSLP'06*, Pittsburgh (USA), September 2006.

[6] D. A. Reynolds y R. C. Rose, "Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, January 1995.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[8] ECJ 16, ," http://cs.gmu.edu/ eclab/projects/ecj/.

[9] R. P. Lippmann, L. Kukolich, y E. Singer, "LNKnet: Neural Network, Machine Learning and Statistical Software for Pattern Classification," *Lincoln laboratory Journal*, vol. 6, no. 2, pp. 249–268, 1993.

n-gram Frequency Ranking with additional sources of information in a multiple-Gaussian classifier for Language Identification

Ricardo Cordoba, Luis F. D'Haro, Juan M. Lucas, Javier Zugasti

Speech Technology Group. Dept. of Electronic Engineering. Universidad Politécnica de Madrid
E.T.S.I. Telecomunicación. Ciudad Universitaria s/n, 28040-Madrid, Spain
{cordoba,lfdharo,juanmak,jzugasti}@die.upm.es

Abstract

We present new results of our n-gram frequency ranking used for language identification. We use a Parallel phone recognizer (as in PPRLM), but instead of the language model, we create a ranking with the most frequent n-grams. Then we compute the distance between the input sentence ranking and each language ranking, based on the difference in relative positions for each n-gram. The objective of this ranking is to model reliably a longer span than PPRLM. This approach outperforms PPRLM (15% relative improvement) due to the inclusion of 4-gram and 5-gram in the classifier. We will also see that the combination of this technique with other sources of information (feature vectors in our classifier) is also advantageous over PPRLM, showing also a detailed analysis of the relevance of these sources and a simple feature selection technique to cope with long feature vectors. The test database has been significantly increased using cross-fold validation, so comparisons are now more reliable.

Index Terms: Language Identification, n-gram frequency ranking, score normalization, feature selection, PPRLM

1. Introduction

The most used technique in Language identification (LID) is the phone-based approach, like Parallel phone recognition followed by language modeling (PPRLM) [1]-[2], which classifies languages based on the statistical characteristics of the allophone sequences with a very good performance. An interesting variant of PPRLM is presented in [5] with several proposals: different ways to combine the allophone sequence information with the acoustic models, use of durations (prosodic information) and a tree-based language model. It is remarkable the integration of several sources of information. In [7] they compare the performance of a neural network with a Gaussian classifier as ours. Another recent line of research is the fusion of different sources of information, as in [8] or [9], which we also address.

PPRLM does not model long-span dependencies: with 4-gram language models results are slightly worse, probably due to unreliable estimation. To solve this, we decided to use a ranking of occurrences of each n-gram with higher n-grams [4], in a similar way to [6] where the ranking is applied to written text. Although the information source is very similar to PPRLM (frequency of occurrence of n-grams), results are clearly better.

This paper is a continuation of the work done in [3] with several information sources and [4]. Section 2 describes the system setup and basic techniques. In Sections 3 and 4 the n-gram ranking technique and new information sources are described. In Section 5, results are presented and discussed. Finally, conclusions are presented in Section 6.

2. System description

2.1. Database

We use a continuous speech database (Invoca), which consists of very spontaneous conversations between controllers and pilots. It is a difficult task, noisy and very spontaneous, with one big drawback: all speakers are native Spanish. So, many of them do not reflect all the phonetic variations in English, and they mix Spanish for greetings and goodbyes even when the rest of the sentence is in English.

In total, we had some 9 hours of speech for Spanish (4998 sentences) and 7 hours for English (3132 sentences). We have considered sentences with a minimum of 0.5 sec., and a maximum of 10 sec., with an average duration of just 4.5 sec., which is another important complication for the LID task. To increase the reliability of results we have performed a cross-fold validation, dividing all the material available in 9 subsets. In each pass we dedicated:

- 4 blocks for estimating the acoustic models & the Gaussian distribution for the LMs and the ranking
- 3 blocks for estimating the language models for PPRLM and the n-gram ranking & the Gaussian distribution for the acoustic scores and duration
- 1 block for the test-set and parameter fine-tuning
- 1 block for the validation set

So, results are more reliable because they use 7 times more material and are for a validation set with unseen data. We checked in [2] that to estimate the Gaussian distribution for the LMs we could use the acoustic models training list, as this data does not participate in the LM estimation. The same applies for the distribution estimation of acoustic scores with the LMs training list.

2.2. General conditions of the experiments

The system uses a front-end with PLP coefficients derived from a mel-scale filter bank (MF-PLP), with 13 coefficients including c0 and their first and second-order differentials, giving a total of 39 parameters per frame. For the phone recognizers, we have used context-independent continuous HMM models. For Spanish, we have considered 49 different allophones and, for English, 61 different allophones. All models use 10 Gaussians densities per state per stream.

2.3. Brief description of PPRLM

The main objective of PPRLM (Parallel Phone Recognition Language Modeling) is to model the frequency of occurrence of different allophone sequences in each language. This system has two stages. First, a phone recognizer takes the speech utterance and outputs the sequence of allophones

corresponding to it. Then, the sequence of allophones is used as input to a language model (LM) module. In recognition, the LM module scores the probability that the sequence of allophones corresponds to the language. It can use several phone recognizers modeled for different languages. Interpolated n-gram language models are used to approximate the n-gram distribution as the weighted sum of the probabilities of the n-grams considered (weights α_1, α_2, and α_3 for unigram, bigram and trigram, respectively). All systems using 4-gram LMs provided worse results [2].

2.4. Gaussian classifier for LID

The general PPRLM approach has a bias problem in the log-likelihood score for the languages considered, especially when the phone recognizers have a different number of units (we have 49 units for Spanish and 61 for English). The language with fewer units will have higher probabilities in the LM score, and so the classifier will tend to select that language. To tackle this issue, we proposed in [2] to use a Gaussian classifier instead of the usual decision formula applied in PPRLM. With all the scores provided by every LM in the PPRLM module we prepare a score vector. With all the sentences in the training database we estimate a Gaussian distribution each language. In recognition, the distance between the input vector of LM scores and the Gaussian distributions for every language is computed, using a diagonal covariance matrix, and the distribution which is closer to the input vector is the one selected as identified language.

One nice feature of a Gaussian classifier is that we can increase the number of Gaussians to better model the distribution that represents our classes and have a Multiple-Gaussian classifier. To increase the number of Gaussians we followed the classical HMM modeling approaches (Gaussian splitting and Lloyd reestimation after each splitting).

One important conclusion of that work is that, instead of absolute values, we need to use differential scores: the difference between the score obtained by the LM of the same language of the acoustic models considered (Spa-Spa or Eng-Eng) and the score obtained by the other 'competing' language(s): SC0 – SC1 and SC3 – SC2 in Figure 1. So, this score can be computed both in training and testing. We applied it to unigram, bigram and trigram separately, with 6 features in total that are listed in Table 1.

Figure 1. PPRLM Scores

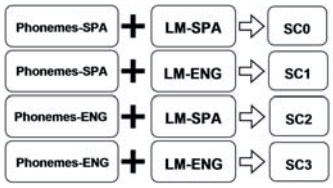


Table 1. Differential score vector

Phonemes-SPA	SCO-SC1 for unigram
	SCO-SC1 for bigram
	SCO-SC1 for trigram
Phonemes-ENG	SC3-SC2 for unigram
	SC3-SC2 for bigram
	SC3-SC2 for trigram

We observed that these differential scores are much more homogeneous, being the result that the estimated distributions exhibit a much smaller overlap with the competing language.

In a multiple language system the proposal for the differential score would be:

$SC_{\text{current language}} - \text{Average}\ (SC_{\text{other languages}})$

One problem that has to be solved is how the weights of the n-grams α_1, α_2, and α_3 from the basic PPRLM equation (1) can be integrated in this approach, as the scores for unigram, bigram, and trigram are independent in our vector.

$$S(w_t, w_{t-1}, w_{t-2}) = \alpha_3 \cdot P(w_t \mid w_{t-1}, w_{t-2}) + \alpha_2 \cdot P(w_{t-1} \mid w_{t-2}) + \alpha_1 \cdot P(w_{t-2}) + \alpha_0 \cdot P_0 \quad (1)$$

We introduce a new contribution: instead of multiplying each feature by its weight in the distance measure, it is much better to divide the variance of the distribution of each score by the corresponding α_i weight (equation (2)). For low α_i, variances increase and so distances are smoothed (which is good for less discriminative features). This smoothing weight is quickly adjusted with good results using the test set.

$$\sigma_i^{final} = \sigma_i^{original} / \alpha_i \quad (2)$$

3. n-gram Frequency Ranking

3.1. Description

We use the same input as PPRLM: the sequence of allophones generated by the phone recognizer. As proposed in [6], we use all training data to compute the number of occurrences of each n-gram (n=1 to 5). We sort those counts, and keep only the M most frequent n-grams, which will form the ranking for that input language. It is known ([6]) that the top n-grams are almost always highly correlated to the language. So, we will use this ranking instead of the LM module considered in PPRLM (see Figure 1).

In testing, for each input sentence a ranking is created using the same procedure. Then, the distance between the input sentence ranking and each ranking is computed. The distance measure is the following (we add the difference in the ranking position for all n-grams in the input sentence):

$$d = \frac{1}{L}\sum_{i=1}^{L} abs(pos\ input_i - pos\ global_i) \quad (3)$$

where L is the number of n-grams in the input sentence. If an n-gram does not appear in the ranking (meaning that it has not appeared in training or it is not in the top n-grams selected) it is assigned the worst distance: the ranking size. The language identified by the system will be the one with the lowest distance. For the Gaussian classifier we now have 10 features in our vector (unigram to 5-gram in both languages).

In [4] we obtained the following conclusions for this technique: optimum ranking sizes range in 3000; it is better to have n-gram specific rankings, instead of a global ranking for all n-grams which include too many unigrams and bigrams which are less discriminative; and rankings should be discriminative.

We wanted to give more relevance in the ranking (higher positions) to the items that are actually more specific to the identified language, i.e. n-grams that appear a lot for one language but appear very little, or never, in the competing languages. We propose a variation of tf-idf, which is used for topic classification. Given the following normalized values:

n_1' = occurrences of item *i* in the current language

n_2' = occurrences of item *i* in the competing language (the average to extend the metric to multiple languages)

The best formula with the same philosophy as tf-idf for the final number of occurrences considered for the ranking (which we will call n_1'') is (more details in [4]):

$$n_1'' = n_1' * (n_1' - n_2') / (n_1' + n_2')^2$$

which normalizes the values between 1 and -1: 1 meaning that the n-gram appears in the current language but not in the other competing ones (n_2'=0), so it is especially relevant for that language; -1 meaning just the opposite (n_1'=0), so the n-gram does not appear in the current language.

4. Inclusion of several information sources

We propose the inclusion of acoustic information in two complementary ways: the average acoustic score of the sentence and the average acoustic score for each phoneme. At the same time, phoneme duration generated by the phone recognizer can be very different depending on the input language, so we can take advantage of that too. For these three sources of information we will just add another feature vector in our classifier, as we will see in this section.

4.1. Inclusion of the sentence acoustic score

First, we will consider the global acoustic score of the sentence (phone recognizer score normalized by the number of frames). We have a vector with two features: the acoustic score obtained in the phone recognizers for each language. So, the approach can be easily extended to several languages.

The acoustic score values were not homogeneous at all, and so, the estimated distributions for competing languages had a big overlap. Then, we decided to use again the "differential scores" idea: we used the difference between the phone recognizer score for Spanish and English as feature value. To extend this approach to several languages:

$$\text{AcScore}_{\text{current language}} - \text{Average}(\text{AcScore}_{\text{other languages}})$$

4.2. Inclusion of the acoustic score for each phoneme

We now considered that the acoustic score for each individual phoneme could also have a strong variation depending on the language. Using our classifier, we modeled the Gaussian distribution for the acoustic score of each phoneme.

For each input sentence we have its corresponding sequence of phonemes using the Spanish and English phone recognizers. We compute the average score for each phoneme appearing in the sentence (averaging the score over all frames belonging to that phoneme) obtaining a feature vector with as many features as the number of phonemes in the system. Obviously, phonemes not appearing in the sentence do not contribute to the final score in the classifier.

Again, the "differential scores" approach is a must, because these scores have a strong variability. To normalize, for every frame: $SC = SC_{Spanish} - SC_{English}$, which is added for all phoneme frames. This approach is clearly better than normalizing using the sentence average score for the "competing" language.

To reduce the size of the feature vector, we grouped some allophonic variations and considered 34 different phonemes for each language. So, we have a vector of 68 features. This vector is obviously too large to have it reliably estimated. In this version of our system we decided to apply a feature selection algorithm to reduce the dimensionality: we keep the *n* features that maximize the following objective function:

$$\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2\ \sigma_2^2} \qquad (4)$$

where μ_1 and μ_2 are the mean values for the feature considering Spanish and English input sentences respectively, and σ_1 and σ_2 are the respective covariances. A high value in this formula means that the feature is very discriminative. There is a very strong correlation among this separation measure and the final results in LID. We tested the system using 24, 30, and 35 features, keeping 30 features as the optimum. To get an idea of the information provided by this objective function, in Table 2 we can see the separation which is obtained with PPRLM and n-gram ranking for each n-gram considered applying equation (4). Discrimination for the ranking trigram is very similar to the PPRLM trigram, but now we can use 4-grams and 5-grams. The separation for the sentence acoustic score is 6.84, whereas for the 30 features of the acoustic score for each phoneme it ranges from 3.52 to 0.54.

Table 2. Comparison of feature discrimination

	PPRLM	Ranking
trigram	10.57	10.12
bigram	8.54	7.12
4-gram	-	6.61
5-gram	-	4.25
unigram	3.17	2.19

An alternative to this feature selection algorithm is to apply LDA to reduce the dimensionality, which is oriented to labeled samples, as we have. Unfortunately, results were slightly worse. LDA has one advantage: it projects into a space of dimension "number of classes -1", which is 1 in our case, so the Gaussian distribution is easily estimated. It would probably work better for a multiple class classification. This will be explored as future work.

One reason of the bad results is probably the "missing values" problem: we have an original vector with 68 components corresponding to phonemes, but several of them do not appear in a sentence. The easy solution is to substitute those missing values by their mean taken from the training database, but that implies some loss of information, and the projection of the test vector is worse. So, we still have to tackle this issue.

4.3. Inclusion of the duration for each phoneme

We considered that phoneme duration could also be different depending on the input language, so we thought that it could be easy to add just another feature vector to our Gaussian classifier. So, we modeled the Gaussian distribution for the average duration of each phoneme in our system. For each input sentence, we computed the average duration for each phoneme and the feature vector had as many features as the number of phonemes. The problem is that this duration produced by the recognizer is quite difficult to normalize. The "differential scores" approach that we should apply here would be to subtract the average duration for the competing language, but, as the phoneme sets are different for each language, this subtraction is not possible. We considered two normalizations: a) Subtract the average phoneme duration of the competing language; b) Subtract the phoneme duration of the competing language for the phoneme which had the largest part in common with the current one, so it will be the most probable "competing" phoneme. (b) was a better option.

We reduced the feature vector using the same feature selection technique as in the previous section, keeping this time 22 features as the optimum value.

5. LID results

5.1. Individual features

When mixing several sources of information differences are less evident. So, we will first show in Table 3 the results of each source independently. There are several interesting conclusions:

- The n-gram ranking provides a **15.4%** relative improvement over PPRLM.
- Phoneme acoustic score is 3% better than the Acoustic sentence score.
- Phoneme duration is the worst discriminative, so we still have a normalization problem with the technique.

Table 3. LID results for individual feature vectors

PPRLM	n-gram Ranking	Sentence Acoustic	Phoneme Acoustic	Phoneme Duration
3.69	3.12	8.14	7.90	24.67

5.2. Combination of several features

In Table 4 we can see the results when combining several feature vectors and the relative improvements over the PPRLM and the Ranking base systems from Table 3. We can extract the following comments:

- Rows 1 & 2: "PPRLM + Phoneme Acoustic" is better than "PPRLM + Sentence Acoustic", as the individual results predicted.
- Row 3: The fusion of PPRLM and duration only provides a low improvement, but it could be expected.
- Row 4 & 8: PPRLM / Ranking + both acoustic scores keeps improving the system, so these scores are complementary
- Rows 5-7: The fusion of the Ranking + additional features provides similar improvements to PPRLM, a bit lower probably because they begin from a much better system.
- Row 9: The fusion of PPRLM and Ranking provides a nice improvement. This is even surprising, as they use the same information source, the n-grams.
- Rows 10 & 11: The fusion of PPRLM + Ranking + Acoustic scores provides further improvements, which shows again that they all provide complementary information.

Table 4. LID results for feature vector combinations

Feature vectors	LID	Improv. PPRLM	Improv. Ranking
PPRLM + Sentence Acoustic	3.10	16.0%	-
PPRLM + Phoneme Acoustic	3.08	16.5%	-
PPRLM + Phoneme Duration	3.49	5.4%	-
PPRLM + both Acoustics	**3.00**	18.7%	
Ranking + Sentence Acoustic	2.78	-	10.9%
Ranking + Phoneme Acoustic	2.77	-	11.2%
Ranking + Phoneme Duration	3.07	-	1.6%
Ranking + both Acoustics	**2.63**	-	15.7%
PPRLM + Ranking	2.85	22.8%	8.7%
PPRLM +Ranking+S. Acoustic	2.66	27.9%	14.7%
PPRLM+Ranking+both Acoust.	2.54	31.2%	18.6%
All	**2.52**	31.7%	19.2%

5.3. Longer span of the ranking technique

We also checked the relevance of 4-grams and 5-grams in LID with this technique. In Table 5 we can see that the LID results considering only up to 4-gram or up to trigram are worse than using all n-grams, and the trigram ranking has similar results as PPRLM. So, we are clearly taking advantage of this longer span using this technique.

Table 5. Independent ranking for each n-gram

	Best result
All n-grams	3.12
Up to 4-gram	3.30
Up to trigram	3.59

6. Conclusions

We have demonstrated that the n-gram Frequency Ranking approach can clearly overcome PPRLM thanks to the longer span that can be modeled. Even the combination of this Ranking with more feature vectors keeps improving the results, showing that all the features proposed provide complementary information (phoneme duration being the worse). The acoustic score for each phoneme is a slightly better feature than the sentence acoustic score.

The measure of separation between pdf distributions (Section 4.2) is a good tool to anticipate which features are going to be actually discriminative for the LID task. LDA provides worse results, probably because of the "missing values" problem.

As future work, we will check these results with a bigger and more "standard" database.

7. Acknowledgements

This work has been partially funded by the Spanish Ministry of Education & Science under contracts DPI2007-66846-c02-02 (ROBONAUTA) and TIN2005-08660-C04-04 (EDECAN-UPM) and by UPM-DGUI-CAM under CCG07-UPM/TIC-1823 (ANETO).

8. References

[1] Zissman, M.A., "Comparison of four approaches to auto-matic language identification of telephone speech," IEEE Trans. Speech&Audio Proc., v. 4, pp. 31-44, 1996.

[2] Córdoba, R., et al. "Integration of acoustic information and PPRLM scores in a multiple-Gaussian classifier for Language Identification". IEEE Odyssey 2006.

[3] Cordoba, R., D'Haro, L.F., et al. "Language Identification using several sources of information with a multiple-Gaussian classifier". Interspeech 2007, pp. 2137- 2140. Belgium.

[4] Cordoba, R., D'Haro, L.F., et al. "Language Identification based on n-gram Frequency Ranking". Interspeech 2007, pp. 354- 357. Belgium.

[5] Navratil, J. 2001. "Spoken Language Recognition – A Step Toward Multilinguality in Speech Processing". IEEE Trans. Speech&Audio Proc., Vol. 9, pp. 678-685.

[6] Cavnar, W. B. and Trenkle, J. M., "N-Gram-Based Text Categorization", Proc. 3rd Symposium on Document Analysis & Information Retrieval, pp. 161-175, 1994.

[7] Gleason, T.P., M.A. Zissman. "Composite background models and score standardization for Language Identification Systems", ICASSP 2001, pp. 529-532.

[8] Gutierrez, J., J.L. Rouas, R. André-Obretch. "Fusing Language Identification Systems using performance confidence indexes". ICASSP 2004, pp. I-385-388.

[9] Li, J., S. Yaman, et al. "Language Recognition Based on Score Distribution Feature Vectors and Discriminative Classifier Fusion". IEEE Odyssey 2006.

PRE-LINGUA - UNA HERRAMIENTA DE APOYO PARA EL PRE-LENGUAJE

William Ricardo Rodríguez Dueñas, Eduardo Lleida Solano

Grupo de Tecnologías de las Comunicaciones
Instituto de Investigación en Ingeniería de Aragón
(wricardo,lleida)@unizar.es

RESUMEN

Este trabajo introduce $Pre-Lingua$, un conjunto de herramientas diseñadas para apoyar la labor diaria de logopedas en el desarrollo del pre-lenguaje en niños que sufren desórdenes en el habla. $Pre-Lingua$ que esta diseñada a manera de juegos trabaja en aspectos como la detección de actividad de voz, el control de la intensidad, tonalidad y respiración y finalmente la vocalización. Para conseguirlo hace uso de las tecnologías del habla y de un motor gráfico encargado de generar las animaciones en una interfaz muy sencilla y atractiva.

1. INTRODUCCIÓN

En la actualidad existen herramientas informáticas para terapia del lenguaje, habla y audición. Algunas de ellas como **SpeechViewer**, desarrollada por IBM trata desordenes de comunicación en diferentes edades, se puede elegir entre: Control de tono, intensidad, sonoridad, duración de la voz, análisis de espectros y pronunciación de fonemas. **Video Voice**, de Micro Video Corporation, ofrece juegos para el tono, amplitud y duración de la voz. **Dr. Speech**, es un sistema que cuenta con varios juegos interactivos, donde el niño recibe retroalimentación del cambio de tono, intensidad, y fonación[1]. Estas herramientas son en lengua inglesa y con licencia de pago, lo que dificulta la labor de logopedas e instituciones de habla hispana.

$Pre-Lingua$ es una herramienta de libre distribución para lengua española que hace parte del proyecto $Comunica$, un conjunto de aplicaciones desarrolladas para mejorar las capacidades de comunicación en personas con desórdenes en el habla.

El desarrollo del lenguaje durante el primer año de vida (pre-lenguaje) en un niño sano incluye aspectos como: la detección de actividad de voz, en donde el niño puede advertir la presencia de personas en su entorno y aprende que con su voz puede interactuar con ellas; el control de la intensidad de la voz donde el niño aprende a modular el volumen de su voz; el control de la respiración ya que es importante para una comunicación fluida; el control de la tonalidad ya que es requerido para una correcta producción del habla (prosodia); y finalmente la vocalización, en donde el niño empieza a generar sonidos articulados y lo preparan para la etapa fonológica en el desarrollo del lenguaje propiamente dicho a partir del primer año [2].

Desafortunadamente en algunos casos este desarrollo no es normal y es afectado por diferentes trastornos físicos o mentales, lo que limita seriamente sus habilidades para comunicarse, aprender una lengua e integrarse a la sociedad. Las Tecnologías del Habla (TH) que apoyan e investigan en campos como: la logopedia, estudios sobre fonética acústica y patológica, y lingüística entre otros, se convierten en una poderosa herramienta para que personas con trastornos del lenguaje se comuniquen de una mejor manera. Es así como el objetivo de este trabajo es el desarrollo de aplicaciones informáticas atractivas para niños a manera de juegos, que basadas en las TH permitan que personas discapacitadas con problemas en el desarrollo del pre-lenguaje, puedan comunicarse e interactuar de una mejor manera con su entorno y con ordenadores inclusive. Esta herramienta puede ser utilizada fácilmente por logopedas ya que no requiere configuraciones previas y además es de fácil uso. Aquí la evaluación logopédica es necesaria para la selección de candidatos debido a los múltiples factores que pueden intervenir en el desarrollo del pre-lenguaje.
El presente artículo muestra en el apartado 2 como se aplican las TH para intentar resolver éstos problemas, en el apartado 3 una descripción general de los juegos de $Pre-Lingua$ y en 4 resultados y conclusiones obtenidos hasta ahora, ya que el desarrollado aquí planteado es el diseño de la herramienta para poder posteriormente evaluarla en diferentes centros de educación especial.

2. TECNOLOGÍAS APLICADAS

Las TH aplicadas en $Pre-Lingua$ incluye un detector de actividad de voz (VAD), estimación de la energía de la señal, análisis LPC para estimar la frecuencia fundamental (Pitch) y los formantes F1 y F2 correspondientes a las vocales del castellano. También se utiliza el motor gráfico ALLEGRO que son rutinas en código C de uso libre y que se encargan de generar las animaciones gráficas en los juegos.

La figura 1 muestra en bloques el procesamiento real-

Este trabajo ha sido subvencionado por el MEC TIN 2005-08660-C04-01 y becas Banco Santander .

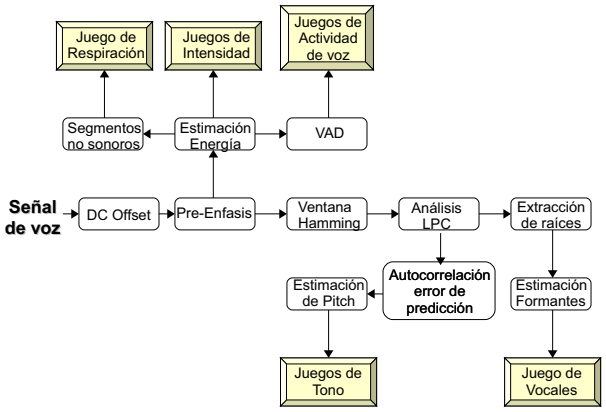


Figura 1. *Procesamiento sobre la señal de voz.*

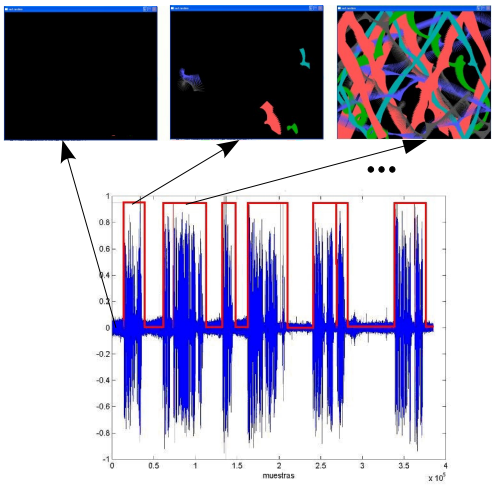


Figura 2. *VAD en la activación de Imágenes.*

izado por $Pre - Lingua$ sobre la señal de voz. La señal de voz tiene una etapa de pre-procesamiento donde se realiza compensación DC, pre-énfasis y un enventanado tipo Hamming. Posteriormente se realiza estimación de la energía de la señal, y análisis de predicción lineal LPC para la estimación de Pitch y formantes. Estos algoritmos entregan al motor gráfico los parámetros necesarios de control en los diferentes juegos, de manera que la realimentación para el usuario de los efectos de su voz es siempre gráfica y en tiempo real. A continuación se explica como se usan éstas tecnologías para cada aspecto del pre-lenguaje trabajado en $Pre - Lingua$.

2.1. Detección de la Actividad de Voz

Un niño con problemas de pre-lenguaje no diferencia los sonidos de su entorno de la voz humana y por consiguiente no advierte que puede usar su propia voz para comunicarse. La TH utilizada es un Voice Activity Detector (VAD) convencional basado en la energía de la señal. Éste entrega una señal binaria de alto nivel en los segmentos donde hay presencia de voz, y de bajo nivel en los segmentos de silencio [3]. La decisión del VAD solo es de alto nivel si la estimación de la energía de la señal supera un umbral pre-establecido y si existe frecuencia de Pitch. De esta manera se diferencia si el segmento analizado corresponde a voz o no voz. Como se aprecia en la figura 2, la salida binaria del VAD (línea roja) se encadena a diferentes animaciones gráficas en donde el objetivo es crear conciencia en el niño de que su voz genera cambios en pantalla. Por ejemplo mover figuras geométricas o permitir que un coche se mueva solo en presencia de voz.

2.2. Control de la Intensidad

Una vez el niño adquiere la habilidad para distinguir su propia voz, él puede aprender a modular la intensidad de la misma. Se utiliza el valor de la estimación de la energía y se lleva a un espacio de valores en píxeles para conseguir una proporcionalidad entre el valor de la intensidad y el movimiento de objetos en pantalla. En la figura 3, puede apreciarse como la evolución de la energía para un segmento sonoro se convierte en la posición vertical del objeto animado y el VAD proporciona el movimiento horizontal. Para inducir al niño a modular la intensidad de su voz hay juegos donde el objeto animado debe evadir obstáculos o desplazarse a través de un laberinto, allí la trayectoria es única y el niño debe variar la intensidad de la voz para llegar al final del juego.

2.3. Control de la Respiración

Hablar fluidamente requiere de una correcta respiración. Para enseñar al niño a manejar la potencia y el mantenimiento de la respiración se han diseñado juegos donde él debe soplar. El sistema analiza de nuevo la energía de la señal pero considera únicamente los segmentos no sonoros (sin pitch) como se muestra en la figura 4. Aquí

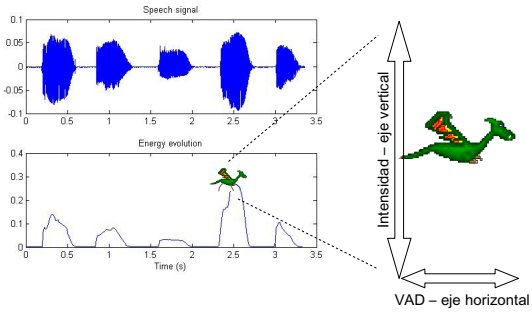


Figura 3. *Intensidad de la voz a posición vertical.*

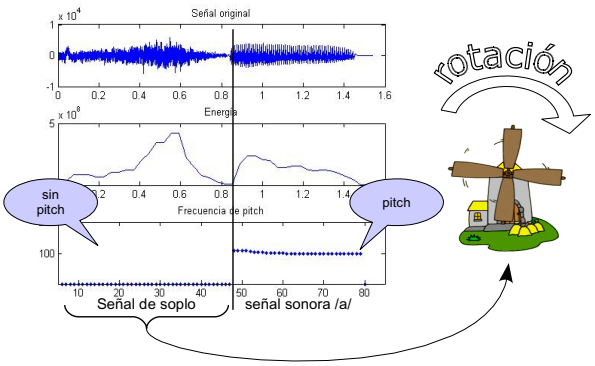


Figura 4. *Intensidad del soplo a rotación.*

la intensidad del soplo se transforma en la rotación de los molinos garantizando que se controle la respiración soplando y no gritando, ya que esta última acción incluye segmentos sonoros.

2.4. Control de la Tonalidad

Con una filosofía similar al control de la intensidad el control de la tonalidad busca que el niño aprenda a modular el tono de su voz. Aquí el análisis de predicción lineal LPC se requiere para separar la influencia del tracto vocal sobre el pulso glotal [4]. Utilizando la autocorrelación del error de predicción, el sistema estima el periodo de pitch y con su inverso la frecuencia de Pitch [5]. Este valor pasa por un filtro de mediana de orden 5 y luego es utilizado por el motor gráfico en los juegos como se muestra en la figura 5. En este juego el pitch controla la posición vertical y la intensidad proporciona velocidad al objeto. Para inducir el control del tono en el niño los juegos presentan diferentes escenarios y objetivos que lo obligan a variar la tonalidad.

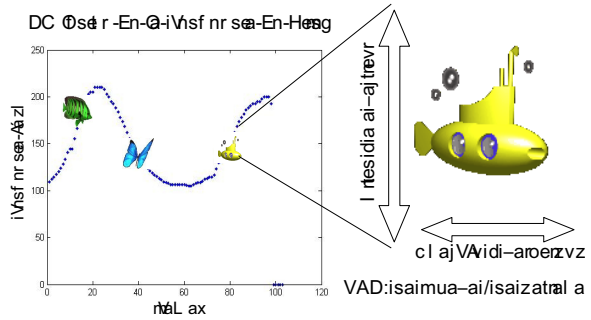


Figura 5. *Frecuencia de pitch a movimiento.*

2.5. Vocalización

En esta etapa del pre-lenguaje el niño debe empezar a producir sonidos vocálicos articulados modificando las características geométricas del tracto vocal. Se trabajan las cinco vocales del castellano /a/, /e/, /i/, /o/ y /u/ haciendo uso del triángulo vocálico. En éste la ubicación específica de cada vocal depende de los formantes F1 y F2 que son las frecuencias de resonancia que tienen lugar en el tracto vocal. Del análisis LPC (orden 12) se considera la influencia del tracto vocal obteniendo las raíces de los coeficientes del filtro predictor y transformándolos en frecuencia analógica. A partir de allí se toman los dos primeros valores que corresponden a los primeros formantes F1 y F2 para enviarlos a motor gráfico. Hay que tener en cuenta que los formantes están correlados con la edad, sexo, y estatura de cada niño [6], para lo cual la calibración inicial de cada triángulo se hace teniendo en cuenta estos parámetros. Basados en cálculos realizados en grabaciones de niños que apoyan este proyecto, el sistema estima aproximadamente la posición del triángulo vocálico de casa usuario, ingresando sexo, edad y estatura antes de iniciar el juego.

Idealmente el sistema debe estimar la posición de los formantes de cada usuario sin necesidad de ingresar datos tabulados. Situación que continua en desarrollo debido a las dificultades de trabajar con niños que tienen serios problemas fonológicos, ya que se les exige pronunciar sonidos muy específicos que permitan estimar características del tracto vocal, asumiendo que éste es un tubo acústico homogéneo.

3. JUEGOS EN PRE-LINGUA

$Pre-Lingua$ se divide en 5 categorías de juegos una para cada aspecto del pre-lenguaje. Las categorías y sus juegos se organizan de manera progresiva en lo que a dificultad respecta a manera de pirámide, aunque pueden ser utilizadas sin ningún orden específico. La figura 6 muestra la pantalla principal de $Pre-Lingua$. A continuación, se describen las categorías.

3.1. Juegos de Actividad de Voz

Esta categoría tiene diferentes juegos. La figura 7 (a), muestra por ejemplo un juego donde la actividad de voz hace que un coche se mueva, en otros un dragón vuela, o aparecen imágenes o figuras geométricas, todas ellas activadas con la presencia de voz.

3.2. Juegos de Intensidad

Como el objetivo aquí es la modulación de la intensidad, los juegos utilizan el mismo coche o el dragón de la categoría anterior. La figura 7 (b) muestra, que la intensidad se ha convertido en la posición vertical del dragón y la propia detección de voz imprime en el dragón un desplazamiento horizontal constante. Así pues el dragón

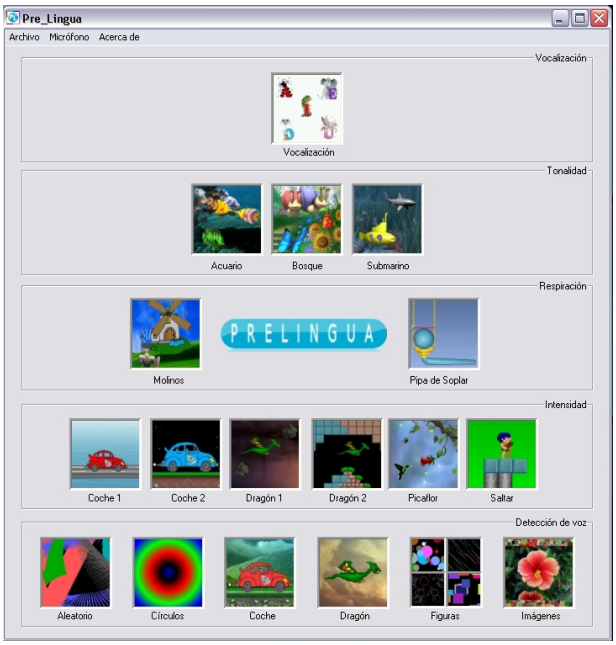


Figura 6. *Pantalla principal de Pre – Lingua.*

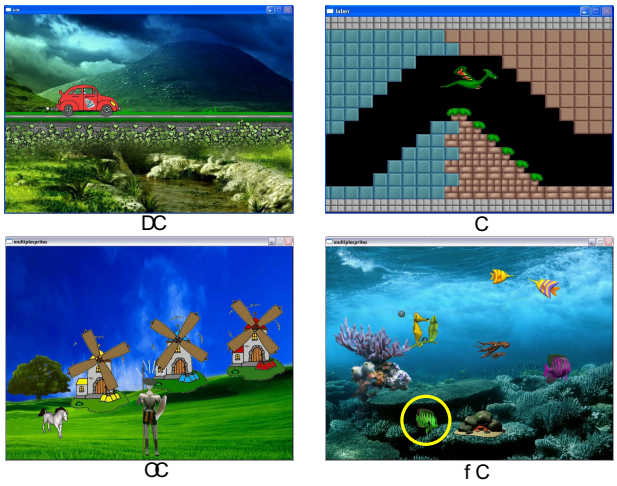


Figura 7. *Juegos en Pre – Lingua, Actividad de voz (a), Intensidad (b), Respiración (c) y Tonalidad (d).*

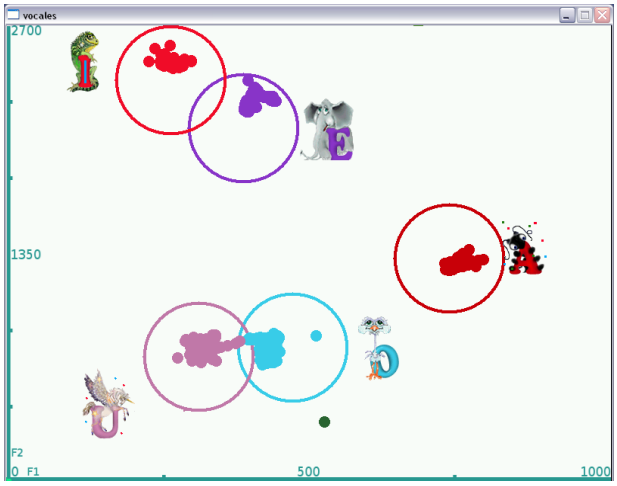


Figura 8. *Juego de Vocalización.*

debe volar a través del laberinto para terminar el juego. En otros juegos la intensidad de la voz se transforma en la velocidad de un coche o en el salto de un muñeco.

3.3. Juego de Respiración

En la figura 7 (c) puede apreciarse que el juego consiste en hacer que el personaje del *Quijote* haga mover los molinos, para conseguirlo el niño debe soplar hacia el micrófono y la intensidad de esta acción incrementa o decrementa la velocidad de rotación de los molinos. Otro juego simula la actividad logopédica de soplar una pipa, donde la intensidad del soplo varia la posición vertical de una esfera.

3.4. Juegos de Tonalidad

En esta categoría la modulación del tono permite variar la posición vertical de diferentes objetos animados, mientras que el VAD permite el movimiento horizontal. En la figura 7 (d) el pez verde (dentro del círculo amarillo) varia su posición vertical en función del tono, el movimiento horizontal es constante y activado por el VAD. El objetivo es seguir a los demás animales como el pulpo y otros peces que inicialmente se encuentran estáticos; al acercarse el pez del niño (pez verde) a los otros animales, estos se animan y se mueven en diferentes trayectorias, de manera que el niño debe seguirlos modificando la tonalidad.

3.5. Juego de Vocalización

Para iniciar el juego se ingresan datos de sexo, edad y estatura del niño. Posteriormente se presenta una imagen con cinco círculos (dianas) cuya área interna corresponde a la región donde caen los formantes para las cinco vocales. La actividad consiste en hacer diana pronunciando cada vocal. El sistema dibuja un punto en la coordenada formada por los formantes estimados en la parte de procesamiento de voz con el color respectivo de cada vocal. Adicionalmente para motivar al niño la imagen de un animalito se animará si la pronunciación cae correctamente dentro de la diana de la vocal seleccionada. Si el sistema detecta formantes que no pertenecen a ninguna vocal se dibujaran puntos de colores aleatorios y por supuesto ningún animal tendrá animación. En la figura 8 puede apreciarse el resultado de pronunciar las cinco vocales y sus correspondientes puntos en las dianas.

4. RESULTADOS Y CONCLUSIONES

$Pre-Lingua$ una herramienta de apoyo para la labor diaria de logopedas que consta de 18 juegos en 5 categorías, esta siendo utilizada en el Colegio Público de Educación Especial La Alborada en Zaragoza, en fase de pruebas. Los logopedas evalúan la herramienta como fácil de usar y muy atractiva para el usuario final.
Hasta el momento la herramienta ha tenido un alto grado de aceptación por parte de los terapistas y niños de la institución, obteniendo respuestas favorables incluso en niños que no eran candidatos iniciales para usar $Pre-Lingua$. En ellos se han visto avances en estimulación temprana como la captura de atención. Los logopedas ven un gran potencial para continuar investigando y mejorando la herramienta.
$Pre-Lingua$ que esta en fase de pruebas, puede descargarse libremente de www.vocaliza.es junto con otras herramientas del proyecto $Comunica$, todas orientadas a discapacidad en el habla.

5. BIBLIOGRAFÍA

[1] N. Garcia, "Tecnología de la voz utilizada en la terapia del lenguaje de niños con deficiencias auditivas," in *Departamento de Ciencias Básicas e Ingeniería, Universidad del Caribe*, Cancun, MEXICO, - 2004.

[2] M. Puyuelo, "Evaluación del lenguaje," vol. 1, pp. 9–17, 203–219, 1996.

[3] W.-R. Rodríguez, C. Vaquero, O. Saz, y E. Lleida, "Aplicación de las tecnologías del habla al desarrollo del prelenguaje y el lenguaje," in *Proceedings of the 2007 Congreso Latinoamericano de Ingeniería Biomédica (CLAIB)*, Isla Margarita, Venezuela, June 2007.

[4] R.-C. Snell y F. Milinazzo, "Formant location from lpc analysis data," *IEEE Transactions on Speech and Audio Processing*, vol. 1, pp. 129–134, 1993.

[5] L. Rabiner R. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978, Capítulo 4.

[6] S. P. Whiteside, "Sex-specific fundamental and formant frequency patterns in a cross-sectional study," in *Volume 110 The Journal of the Acoustical Society of America*, UK, July - December 2001.

PROCEDIMIENTO PARA LA MEDIDA Y LA MODIFICACIÓN DEL JITTER Y DEL SHIMMER APLICADO A LA SÍNTESIS DEL HABLA EXPRESIVA

Carlos Monzo, Ignasi Iriondo y Elisa Martínez

GPMM - Grup de Recerca en Processament Multimodal
Enginyeria i Arquitectura La Salle, Universitat Ramon Llull
Quatre Camins 2, 08022 Barcelona, Spain

{cmonzo, iriondo, elisa}@salle.url.edu

RESUMEN

En este trabajo se presenta un nuevo procedimiento para la medida de los parámetros de cualidad de voz (VoQ), el *jitter* y el *shimmer*. Este nuevo procedimiento tiene en consideración la prosodia del enunciado, de manera que su efecto se atenúa antes de realizar la medida de cada uno de los parámetros. El objetivo, además de realizar la medida de una forma más fiable, es el de modificar estos parámetros de forma que puedan ser utilizados en síntesis del habla expresiva, por ello, en paralelo a esta nuevo procedimiento de análisis, se presenta cómo llevar a cabo la modificación de ambos. Finalmente, se realiza una evaluación mediante una prueba perceptual CMOS sobre cuatro estilos expresivos: agresivo, alegre, sensual y triste; provenientes de la salida de un sistema de conversión de texto en habla con modelado prosódico, de modo que se hace un estudio de la utilidad de estos parámetros bajo diferentes situaciones.

1. INTRODUCCIÓN

El reconocimiento automático del habla y la conversión de texto en habla (CTH) son áreas de investigación en las que el habla expresiva se está usando con el objetivo de mejorar la naturalidad de la interacción persona-máquina. Ejemplos de esta investigación los encontramos en estudios sobre reconocimiento de emociones [1] o transformación de voz [2][3]. La prosodia y la cualidad de voz (de aquí en adelante VoQ) son parámetros utilizados en la representación del contenido emocional del habla tal y como se presenta en [1][4]. A pesar de que la VoQ ha sido menos estudiada que la prosodia, trabajos recientes proponen ambas informaciones para mejorar el modelado acústico del habla expresiva [5][6].

En este trabajo nos interesa la VoQ, que tradicionalmente ha sido analizada de manera independiente a la prosodia, ya sea en aplicaciones de patologías de voz [7], donde no se considera por la naturaleza y condiciones de medida, o bien en estudios sobre estilos expresivos [5][6], donde ya se observa que la prosodia se debe considerar, tal y como señala [8].

El objetivo principal de este trabajo, es el de presentar un nuevo procedimiento de análisis y modificación de los parámetros de VoQ, el *jitter* y el *shimmer*, teniendo en cuenta el efecto de la prosodia. En la bibliografía [9][10][11] se muestra como su medida se realiza sin considerar la variación prosódica debida a la expresividad, o emoción transmitida, de forma que es necesario en aplicaciones como síntesis del habla expresiva o reconocimiento de emociones, considerar su efecto para así tratar de cancelarlo y obtener una medida de VoQ sin interferencias. Complementariamente, se evalúa el efecto de añadir el *jitter* y el *shimmer* al habla generada por un CTH basado únicamente en modificación prosódica.

Este artículo está organizado como sigue. En el apartado 2 se introduce el material de voz usado en el diseño y evaluación. El apartado 3 explica el nuevo procedimiento de análisis para el *jitter* y el *shimmer*. Los apartados 4 y 5 presentan la modificación de estos parámetros, así como la evaluación y discusión de los resultados. Para terminar, el apartado 6 muestra las conclusiones alcanzadas.

2. MATERIAL DE VOZ

El material de voz usado para realizar los experimentos sobre los nuevos procedimientos de medida del *jitter* y del *shimmer*, es el mismo que utiliza el CTH desarrollado por el GPMM [12], se trata de cinco corpus de habla expresiva (o emocionada): neutro, agresivo, alegre, sensual y triste; en español y grabados por una locutora profesional. En [6] se encuentra una explicación detallada de ellos.

Los procedimientos de análisis y modificación del *jitter* y del *shimmer* se han aplicado sobre muestras de habla sintetizada, generadas a partir del corpus de habla neutra y con el modelo prosódico de la expresividad deseada [13].

Este trabajo ha sido subvencionado por el proyecto SALERO (IST-FP6-027122) de la Comisión Europea.

3. NUEVA METODOLOGÍA DE ANÁLISIS

Este apartado expone un nuevo procedimiento para la medida del *jitter* y del *shimmer*. Primero se presenta una descripción de cada uno de ellos y posteriormente se explica la propuesta. El cálculo habitual de estos parámetros, como el realizado por la herramienta Praat [10], no tiene en cuenta variaciones ni de F_0 ni de la energía debidas principalmente al efecto de la prosodia.

3.1. Jitter

Según [5], el *jitter* se corresponde a las variaciones de F_0 que existen en el tramo de habla analizado, representadas como un ruido por modulación en frecuencia.

El procedimiento que se propone, parte de la información de marcas de F_0 calculadas únicamente en las zonas sonoras de la señal de voz, usando para ello el algoritmo RAPT [14]. A partir de ellas se calcula la curva de F_0 en cada una de las zonas sonoras y se lleva a cabo una transformación logarítmica utilizando semitonos [15], consiguiéndose así una normalización relativa al tono medio y una mejor representación de la percepción subjetiva de las variaciones de tono. La transformación de hercios a semitonos y su inversa se muestra en las ecuaciones (1) y (2) respectivamente, donde 'ref' es la frecuencia de referencia. Para este trabajo, se ha tomado como referencia la F_0 media del locutor para la expresividad deseada, calculada a partir del correspondiente corpus de síntesis.

$$\mathrm{Hz} = 2^{\mathrm{St}/12} \cdot \mathrm{ref} \tag{1}$$

$$\mathrm{St} = 12 \cdot \left[\ln(\mathrm{Hz}/\mathrm{ref})/\ln 2\right] \tag{2}$$

A partir de los valores transformados en semitonos, se realiza una detección de los tramos de crecimiento y decrecimiento del contorno de F_0 a partir de un análisis de la pendiente. Para cada tramo obtenido se lleva a cabo una regresión lineal, que se resta de la curva inicial de F_0, con el fin de tratar de anular el efecto de la prosodia (véase ejemplo en la Figura 1).

Para terminar, se calcula la variación de F_0 ($\Delta F0$) entre periodos consecutivos ($F0$) tal y como muestra la ecuación (3) y, finalmente, se calcula el valor del *jitter* para cada una de las tramas según la ecuación (4):

$$\Delta F0_i(j) = F0_i(j+1) - F0_i(j) \tag{3}$$

$$\mathrm{jitter_i} = \frac{1}{N} \cdot \sum_{j=1}^{N} \Delta F0_i(j)^2 \tag{4}$$

Donde j = 1:(núm de marcas de F_0 en la trama) - 1, i = trama bajo análisis y N = longitud de $\Delta F0_i$.

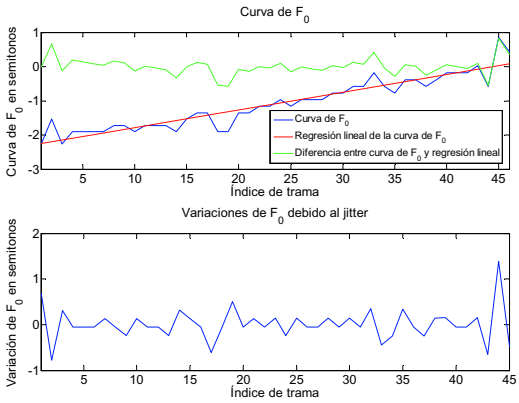


Figura 1. *Extracción de la variabilidad de F_0*

3.2. Shimmer

El *shimmer* computa las variaciones de amplitud de la forma de onda tal y como se presenta en [5]. Describe un ruido por modulación en amplitud.

El nuevo procedimiento propuesto está inspirado en el expuesto para el *jitter*, por tanto parte de las marcas de F_0 de las zonas sonoras. Se calcula la curva de amplitudes pico a pico máximas, por periodo de F_0, en cada una de las zonas sonoras, llevando a cabo por último una transformación logarítmica, aplicando el logaritmo natural.

Una vez se dispone de los valores transformados, se elimina el efecto prosódico de la energía igual que se hizo para la F_0 en el apartado 3.1 (véase la Figura 2).

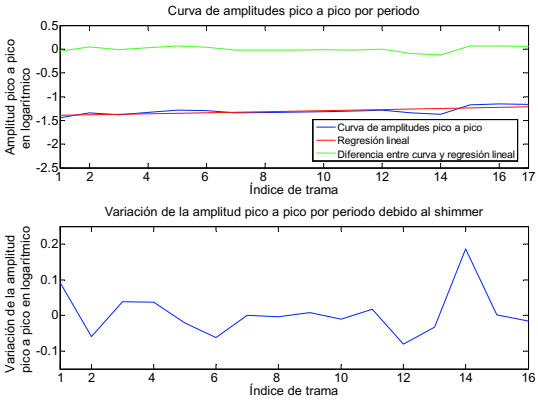


Figura 2. *Extracción de la variabilidad de amplitud*

En (5) se calcula la variación de amplitud pico a pico ($\Delta cpap$) en periodos consecutivos ($cpap$), presentando en (6) el cálculo de *shimmer* por trama:

$$\Delta cpap_i(j) = cpap_i(j+1) - cpap_i(j) \tag{5}$$

$$\mathrm{shimmer_i} = \frac{1}{N} \cdot \sum_{j=1}^{N} \Delta cpap_i(j)^2 \tag{6}$$

Donde j = 1:(núm de periodos de F_0 en la trama) - 1, i = trama bajo análisis y $\boldsymbol{N}$ = longitud de $\Delta cpap_i$.

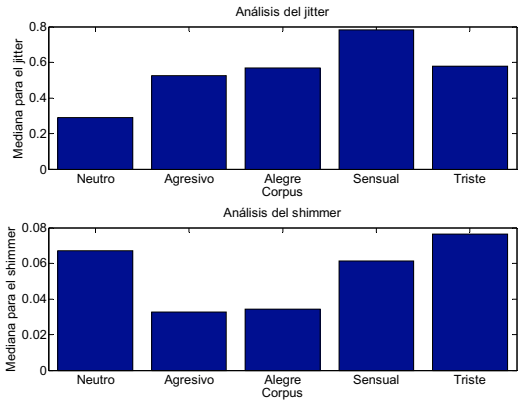


Figura 3. *Análisis del jitter y del shimmer sobre los 5 corpus expresivos*

4. EXPERIMENTOS

En este apartado se muestra como, a partir del procedimiento de análisis presentado para la medida del *jitter* y del *shimmer*, estos parámetros pueden ser modificados. A partir de aquí se evalúa cómo afectan éstos a la identificación de expresividades generadas por un CTH, por un lado, únicamente con modelado prosódico [13], y por otro con modificación de VoQ.

4.1. Modificación del jitter y del shimmer

La modificación para ambos sigue el mismo procedimiento, basándose en la inserción de ruido blanco sobre una curva de F_0 o de amplitudes pico a pico limpias de *jitter* o de *shimmer*. Para ello, a partir de la prosodia se calcula la regresión lineal de aquellos tramos donde la curva de interés mantenga su tendencia, y sobre ella se realizará la adición del ruido blanco.

El ruido blanco que se añade tiene como potencia el valor esperado del *jitter* y del *shimmer* adecuado a la expresividad que se desea simular. Estos valores se conocen a partir del proceso de análisis sobre los corpus expresivos, presentados en el apartado 2, correspondiéndose al valor de mediana obtenido a partir de estadística descriptiva sobre cada uno de los corpus.

Los resultados obtenidos para los diferentes parámetros *jitter* y *shimmer*, usando el procedimiento expuesto, se muestran en la Figura 3. El utilizar la mediana y no la media, se debe a que de este modo, después de analizar las distintas distribuciones, se ha visto que se evitan valores atípicos que puedan desviar el valor medio de la medida.

4.2. Evaluación

Dado el nuevo procedimiento de análisis y modificación del *jitter* y del *shimmer*, el siguiente paso ha sido evaluar cómo pueden contribuir, cada uno por separado o bien de forma conjunta, a la síntesis del habla expresiva.

La prueba parte de 5 enunciados sintetizados usando el CTH presentado en [12] con 4 expresividades diferentes: agresiva, alegre, sensual y triste; a partir de aplicar modelos prosódicos para cada una de ellas [13] sobre un enunciado originariamente neutro. Una vez generadas se les aplicará la modificación del *jitter* y del *shimmer* para su posterior evaluación. Ésta se realiza mediante una prueba perceptual CMOS [16] usando la interfaz web presentada en [17]. Si se desea profundizar sobre la expresividad obtenida usando únicamente modelado prosódico se recomienda la lectura de [18].

Los enunciados se muestran en parejas, comparando la original sintetizada usando modelos prosódicos con la modificada usando el *jitter*, el *shimmer* o ambos parámetros, dando lugar a 60 comparaciones. Cada uno de los evaluadores, 13 en total, eligió si la intensidad de la expresividad presentada en el ejemplo era "*mucho más*", "*más*", "*poco más*" o "*igual*" que la del otro, con una puntuación de: 3, 2, 1, 0, -1, -2 y -3.

5. RESULTADOS Y DISCUSIÓN

En cuanto a los resultados obtenidos, los valores positivos se han reservado para aquellos casos donde, por usar VoQ, la expresividad se muestra con mayor intensidad, mientras que los negativos indican que es la original con solo modelado prosódico (véase la Figura 4).

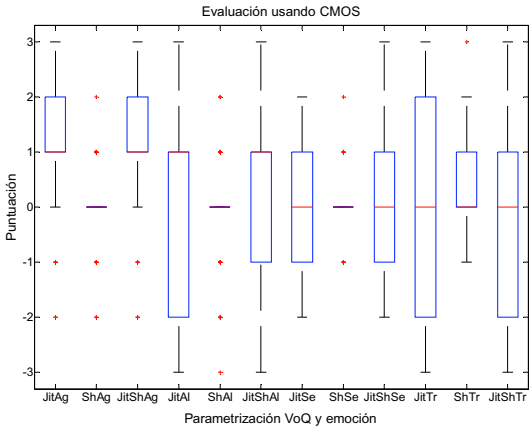


Figura 4. *Resultados de la prueba CMOS sobre la VoQ*

Por otro lado, el valor de CMOS medido por configuración, calculado como su valor medio, se presenta en la Tabla 1, señalando en cursiva a aquellos donde la VoQ mejora la percepción de la expresividad y en negrita cuando su efecto es más representativo.

Tabla 1. *CMOS para 3 configuraciones y 4 expresividades*

	Agresiva	Alegre	Sensual	Triste
Jitter	***1.06***	0.00	-0.12	-0.06
Shimmer	*0.12*	-0.06	*0.08*	*0.29*
Sh + Jit	***1.14***	*0.18*	-0.03	-0.31

Como se puede observar en la Figura 4, la expresividad para la que el efecto de la VoQ intensifica en mayor grado su percepción es la "Agresiva", con un valor superior a 1 en la escala CMOS (véase la Tabla 1). Por otro lado, se tiene que la "Alegre" presenta buenos resultados, mediana igual a 1, a pesar de su dispersión. El resto de expresividades, la "Sensual" y la "Triste", dan resultados con una elevada dispersión, hecho que hace pensar en su utilidad solamente en ciertos casos. Estos resultados son interesantes en tanto que la "Agresiva" y la "Alegre" daban los peores resultados en estudios que usaban solamente prosodia [18], por tanto, los parámetros *jitter* y *shimmer*, la complementará al generar estas expresividades.

Por último, en cuanto al parámetro de VoQ que contribuye al incremento en la intensidad de la expresividad presentada, es el *jitter* para el caso de la "Agresiva", el *shimmer* para la "Triste" y una combinación de ambos para la "Alegre" y la "Sensual".

6. CONCLUSIONES

En este trabajo se ha presentado un nuevo procedimiento para la medida de los parámetros de VoQ, el *jitter* y el *shimmer*. Esta metodología trata de evitar la dependencia con la prosodia de forma que puede ser utilizada en el análisis del habla expresiva.

Visto el procedimiento de análisis, se ha presentado cómo realizar la modificación de estos parámetros, mostrando su utilidad en síntesis del habla expresiva.

Por último, con el objetivo de evaluar la utilidad y dependencia de cada parámetro con una expresividad diferente, se ha llevado a cabo una prueba perceptual CMOS, donde la utilidad de los parámetros de VoQ en la síntesis del habla expresiva ha quedado justificada.

De los resultados obtenidos, se plantea como trabajo futuro, realizar un análisis de la dependencia del lugar del enunciado donde el evaluador centra su atención, pudiéndose así aplicar las modificaciones de VoQ de forma más específica.

7. BIBLIOGRAFÍA

[1] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, y J. Taylor. "Emotion recognition in human-computer interaction", IEEE Signal Processing Magazine, vol. 18, no. 1, pp. 32-80, 2001.

[2] C. Drioli, G. Tisato, P. Cosi, y F. Tesser, "Emotions and voice quality: experiments with sinusoidal modeling", VOQUAL'03, pp. 127-132, Geneva, 2003.

[3] O. Turk, M. Schröder, B. Bozkurt, y L.M. Arslan, "Voice quality interpolation for emotional text-to-speech synthesis", INTERSPEECH, pp. 797-800, Lisbon, 2005.

[4] C. Gobl, y A. Ní Chasaide, "The role of voice quality in communicating emotion, mood and attitude". Speech Communication, 40, 189-212. 2003.

[5] C. Monzo, F. Alías, I. Iriondo, X. Gonzalvo, y S. Planet, "Discriminating Expressive Speech Styles by Voice Quality Parameterization", ICPhS, pp. 2081-2084, Saarbrücken, 2007.

[6] I. Iriondo, S. Planet, J.C. Socoró, F. Alías, C. Monzo, y E. Martínez, "Expressive speech corpus validation by mapping subjective perception to automatic classification based on prosody and voice quality", ICPhS, pp. 2125-2128, Saarbrücken, 2007.

[7] F. Núñez, P. Corte, C. Suárez, B. Señaris, y G. Sequeiros, "Evaluación perceptual de la disfonía: correlación con los parámetros acústicos y fiabilidad", Acta otorrinolaringológica española: Órgano Oficial de la Sociedad Española de Otorrinolaringología y Patología Cérvico-Facial, vol. 55, no. 6, pp. 282-287, 2004.

[8] M. Swerts, y R. Veldhuis, "The effect of speech melody on voice quality", Speech Communication, vol. 33, pp. 297-303, 2001.

[9] R.E. Slyh, W.T. Nelson, y E.G. Hansen, "Analysis of mrate, shimmer, jitter, and F_0 contour features across stress and speaking style in the SUSAS database", ICASSP '99, vol. 4, pp. 2091-2094, Phoenix, 1999.

[10] P. Boersma, "Praat, a system for doing phonetics by computer", Glot International, vol. 5, no. 9-10, pp. 341-345, 2001.

[11] A. Verma, y A. Kumar, "Introducing Roughness in Individuality Transformation through Jitter Modeling and Modification", ICASSP'05, vol. 1, pp. 5- 8, ISSN: 1520-6149 ISBN: 0-7803-8874-7, Philadelphia, 2005.

[12] F. Alías, e I. Iriondo, "La evolución de la Síntesis del Habla en Ingeniería la Salle", 2JTH02, Granada, 2002.

[13] I. Iriondo, J.C. Socoró, L. Formiga, X. Gonzalvo, F. Alías, y P. Miralles, "Modelado y estimación de la prosodia mediante Razonamiento Basado en Casos", 4JTH06, pp. 183-188, ISBN 84-96214-82-6, Zaragoza, 2006.

[14] D. Talkin, "A Robust Algorithm for Pitch Tracking (RAPT)", Speech Coding and Synthesis, W. B. Kleijn and K. K. Paliwal, Eds., chapter 14, pp. 495–518. Elsevier Science, Amsterdam, 1995.

[15] G. Fant, A. Krucenberg, K. Gustafson, y J. Liljencrants, "A new approach to intonation analysis and synthesis of Swedish", Proceedings of Fonetik, pp. 161-64, Stockholm, 2002.

[16] ITU-P.800, "Methods for subjective determination of transmission quality", Recommendation P.800 International Telecommunication Union (ITU), 1996.

[17] S. Planet, I. Iriondo, E. Martínez, y J.A. Montero, "TRUE: an online testing platform for multimedia evaluation", LREC'08. Marrakech, 2008.

[18] I. Iriondo, "Producción de un corpus oral y modelado prosódico para la síntesis del habla expresiva", Tesis Doctoral, Barcelona, 2008.

T-NORM Y DESAJUSTE LÉXICO Y ACÚSTICO EN RECONOCIMIENTO DE LOCUTOR DEPENDIENTE DE TEXTO

Daniel Hernández López[1], Doroteo Torre Toledano[1], Cristina Esteve Elizalde[1], Joaquín González Rodríguez[1], Rubén Fernández Pozo[2] y Luis Hernández Gómez[2]

[1]ATVS Biometric Recognition Group, Universidad Autónoma de Madrid, España
[2]GAPS, SSR, Universidad Politécnica de Madrid, España

RESUMEN

Este trabajo presenta un estudio extenso sobre T-norm aplicado a Reconocimiento de Locutor Dependiente de Texto, analizando también los problemas del desajuste léxico y acústico. Veremos cómo varían los resultados teniendo en cuenta la dependencia de género y realizando T-norm a nivel de frase, fonema y estado con cohortes de impostores de distintos tamaños. El estudio demuestra que implementar T-norm por fonema o estado puede llegar a conseguir mejoras relativas de hasta un 16% y que realizar una selección de cohorte basada en el género puede mejorar más aún los resultados con respecto al caso independiente de género.

1. INTRODUCCIÓN

El Reconocimiento Automático de Locutor es una disciplina de la biometría que consiste reconocer la identidad de una persona (locutor) a través de la voz. Dentro de ésta hay dos grandes vertientes, el Reconocimiento de Locutor Independiente de Texto y el Reconocimiento de Locutor Dependiente de Texto. La segunda de ellas parece haber quedado en segundo plano comparada con la primera, muy probablemente debido a la ausencia de evaluaciones competitivas como las hay para Reconocimiento de Locutor Independiente de Texto [1].

El Reconocimiento de Locutor Dependiente de Texto tiene la particularidad de que el sistema dispone, tanto para entrenamiento como para test, de las trascripciones de la locución. Esto significa que mediante un diccionario fonético podemos disponer de la trascripción fonética de lo que se dice en la locución, lo que hace que se consigan buenos resultados con menor cantidad de habla que en Reconocimiento de Locutor Independiente de Texto. Como es habitual en este tipo de sistemas, en el nuestro utilizamos Modelos Ocultos de Harkov (HMMs) [2] para modelar las características fonéticas de los locutores. Utilizar HMMs permite tener modelos independientes de cada fonema para cada locutor, donde cada uno de los fonemas estará modelado como una serie de probabilidades de transición entre estados, y cada estado estará representado mediante un Modelo de Mezclas de Gaussianas (GMM) [3]. Con estas herramientas y disponiendo de la trascripción fonética, se puede realizar un reconocimiento fonético, utilizando el algoritmo de Viterbi, que proporcione una trascripción fonética con los instantes de comienzo y fin de cada uno de los fonemas y de sus correspondientes estados.

Esta serie de características suponen varias ventajas al Reconocimiento de Locutor Dependiente de Texto frente al Independiente de Texto, pero sin duda la mayor de ellas es poder trabajar, tanto en entrenamiento como en reconocimiento, con niveles por debajo de la frase (palabra, fonema y estado). Dicha ventaja ha sido utilizada múltiples veces en esta disciplina tanto en entrenamiento como en reconocimiento. Entonces ¿porqué no utilizarla en T-norm?

En este trabajo veremos cómo se pueden mejorar los resultados finales del sistema mediante la conocida técnica de T-norm. Hasta ahora esta técnica ha sido muy utilizada en Reconocimiento de Locutor Independiente de Texto y, aunque en menor medida, también en Reconocimiento de Locutor Dependiente de Texto. Sin embargo en todos los casos en que se ha utilizado, T-norm ha sido aplicado a la puntuación global de la locución de test. En el caso de Reconocimiento de Locutor Independiente de Texto parece lógico que se haga así, pero en el caso de Reconocimiento de Locutor Dependiente de Texto parece mejor aprovecharse de la ventaja de poder trabajar con niveles inferiores. Además estudiaremos cómo influye el género y el tamaño de la cohorte de impostores de T-norm.

El resto del artículo está organizado de la siguiente manera: en la Sección 2 describiremos el sistema del que se parte y que ha evolucionado a lo largo de los experimentos, en la Sección 3 explicamos como se implementa T-norm para los experimentos realizados, en la Sección 4 se muestran las bases de datos utilizadas para los experimentos de las Secciones 5 y 6 y por último se presentan las conclusiones en la Sección 7.

2. DESCRIPCIÓN DEL SISTEMA DE PARTIDA

Se parte de un sistema de Reconocimiento de Locutor Dependiente de Texto basado en HMMs. La parametrización que se ha aplicado al audio usado en

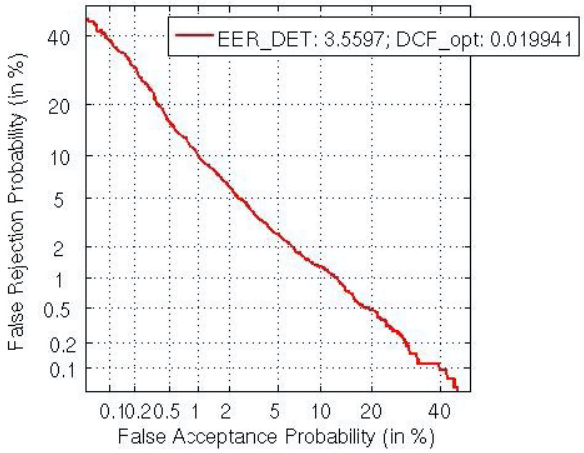


Figura 1. *Curva DET para el sistema de partida.*

entrenamiento y test se basa en la extracción de coeficientes cepstrales mediante filtros de Mel (MFCC, *Mel Frequency Cepstral Coefficients*), tomados en formato $(13+\Delta+\Delta\Delta)$. El sistema de partida realiza alineamiento no completamente forzado de la trascripción tanto para entrenamiento como para verificación (no es totalmente forzado porque se incluyen silencios opcionales entre palabras). Esto es porque aunque se tenga una trascripción textual de lo que se ha pronunciado en la locución no se sabe si hay silencio entre palabras ni de qué duración es éste. De esta forma, realizado un reconocimiento fonético con un modelo de silencio opcional se mejora el alineamiento temporal, lo cual favorece tanto a la etapa de entrenamiento como posteriormente la de reconocimiento.

Dada la trascripción fonética correctamente alineada en el tiempo y el audio parametrizado, el sistema realiza la adaptación al locutor del modelo acústico independiente del locutor. Para ello la adaptación se realiza en tres fases.

En una primera fase se adaptan las medias de las Gaussianas de los Modelos de Mezclas de Gaussianas de cada uno de los estados de cada fonema de forma global. Esto quiere decir que se adaptan todas las medias de forma conjunta. Esta adaptación se hace según el algoritmo MLLR [4] (*Maximum Likelihood Linear Regression*) de forma global, sin clases de regresión. De esta adaptación se obtiene el modelo de transformación lineal, que consiste en una matriz para transformar los modelos fonéticos independientes del locutor en modelos adaptados al locutor. De esta forma no es necesario guardar un modelo de cada locutor, sino simplemente el modelo de transformación. Posteriormente se adaptan los modelos resultantes empleando MLLR, ahora con 2 clases de regresión, obteniendo un nuevo modelo de transformación lineal. Finalmente se aplica adaptación MAP (*Maximum A Posteriori*) [4] a los modelos después de haber sido transformados con el modelo de adaptación MLLR global y posteriormente con 2 clases de regresión.

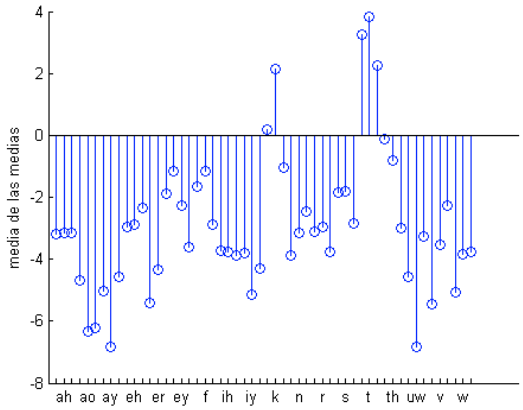


Figura 2. *Medias de las puntuaciones medias de los fonemas (compuestos por 3 estados) y estados en tests de impostores.*

Una vez adaptados los modelos se procede a realizar la fase de evaluación con intentos tanto *target* (donde la locución a verificar es del locutor representado por el modelo) como *non-target* (donde la locución a verificar es de un locutor distinto al representado por el modelo). Se obtienen puntuaciones para cada estado de cada fonema enfrentando la locución de test con el modelo adaptado al locutor y restándola de la puntuación obtenida de enfrentarla al modelo independiente del locutor.

Por último se eliminan las puntuaciones obtenidas por los silencios y se promedia la puntuación general de la locución de test. El resultado obtenido de esta forma para la base de datos YOHO (descrita en la Sección 4), se representa en forma de curva DET en la Figura 1.

3. T-NORM A DISTINTOS NIVELES

Lo que se propone en este artículo es un estudio sobre T-norm, que básicamente consiste en tomar las puntuaciones obtenidas por una cohorte de impostores y calcular la media (μ) y la desviación típica (σ) de dichas puntuaciones. De esta forma se calcula la nueva puntuación ($score_{T-norm}$) como la puntuación obtenida por el locutor que realiza el intento de acceso ($score$) menos la media, dividido entre la desviación típica, como se muestra en la Fórmula 1.

$$score_{T-norm} = \frac{score - \mu}{\sigma} \qquad (1)$$

La clave de este estudio consiste en que se realizará este proceso no sólo a nivel de locución (como suele ser habitual) sino también a nivel de fonema y de estado. Por otra parte se implementará también T-norm dependiente de género. Esto significa que la cohorte de impostores estará compuesta por locutores del mismo género que el locutor *target* para experimentos de este tipo.

La idea de realizar este tipo de T-norm surge de un estudio analítico de las puntuaciones. Como podemos observar en la Figura 2 las puntuaciones de impostor de los estados de un mismo fonema tienen una cierta correlación mientras que entre fonemas las puntuaciones son muy dispares. Esto nos induce a pensar que realizar T-norm a nivel de fonema o estado puede reportarnos buenos resultados debido a que de este modo alinearemos las puntuaciones obtenidas por cada fonema y estado, que parecen desalineadas (Fig. 2).

4. DESCRIPCIÓN DE LAS BASES DE DATOS

4.1. YOHO

Se ha usado la base de datos YOHO [5] para este experimento. Esta base de datos tiene 138 locutores, de los cuales 106 son hombres y 32 son mujeres. Cada locutor presenta 96 locuciones de entrenamiento repartidas en 4 sesiones y 40 locuciones de test repartidas en 10 sesiones. Se han utilizado 6 locuciones de entrenamiento de la primera sesión para realizar el entrenamiento y todas las locuciones de test del locutor como intentos *target* para la etapa de verificación, tomando una locución al azar de cada uno de los demás locutores como intentos *non-target*. Cabe destacar que el léxico de esta base de datos consiste en frases de pares de dígitos (p.e. 32-98-64) y que no hay ninguna relación entre los dígitos pronunciados en entrenamiento y test, con lo cual tenemos un importante desajuste léxico.

4.2. BioSec

Para otro experimento realizado se ha usado la base de datos BioSec Baseline [6]. En esta base de datos hay 150 locutores cuyo idioma nativo es el castellano. Cada locutor ha grabado 2 sesiones con 4 locuciones cada una de un número aleatorio asignado al usuario (el mismo para todas las locuciones de las 2 sesiones). Se han utilizado las 4 frases de la primera sesión para entrenar los modelos acústicos del locutor (se entrena un modelo con cada frase) y las 4 de la segunda sesión como intentos *target* para la fase de test, siendo los intentos *non-target* la primera frase de la primera sesión del resto de impostores (sin enfrentamientos simétricos). Todas las locuciones descritas anteriormente se han realizado de forma idéntica para 4 escenarios. Castellano grabado con un micrófono de unos auriculares (cercano), castellano grabado con un micrófono integrado en una webcam (lejano) y otros 2 escenarios equivalentes en inglés.

5. EXPERIMENTOS CON T-NORM EN FUNCIÓN DEL NIVEL, COHORTE Y GÉNERO

Para poder implementar T-norm, se ha realizado un reconocimiento de cada locución de test con los modelos de locutores de la cohorte de impostores por cada enfrentamiento tanto *target* como *non-target*. De esta forma se han realizado 3 tipos de T-norm en función de la cohorte de impostores: una con una cohorte fija de 20 locutores, 10 hombres y 10 mujeres, a la que llamaremos TN10; otra con una cohorte variable de 60 locutores, 30 hombres y 30 mujeres, a la que llamaremos TN30; una última con una cohorte masculina variable que incluye como impostores todos aquellos locutores que no sean ni el *target* ni el *non-target* (en el caso de que se trate de una prueba *non-target*), a la que llamaremos TNMale.

Para el caso sin T-norm y TN10 hemos realizado experimentos tanto dependientes de género como independientes de género, mientras que para TN30 y TNMale únicamente hemos realizado experimentos dependientes de género. Para los experimentos independientes de género anteriormente descritos se han obtenido los resultados expresados en EER (*Equal Error Rate*) mostrados en la Tabla 1.

T-norm\Nivel	Frase	Fonema	Estado
No	3.56		
TN10	3.91	2.98	3.04

Tabla 1. *EERs (%) obtenidas para distintos tipos de T-norm independiente de género en función del nivel.*

Como podemos ver en la Tabla 1 resulta mucho mejor realizar T-norm a nivel de estado o fonema que a nivel de frase, de hecho podemos ver que es incluso mejor no implementar T-norm que hacerlo a nivel de frase para este experimento en concreto. A continuación vemos en la Tabla 2 como evolucionan los resultados al incrementar el número de impostores de la cohorte y realizar una selección por género de la cohorte de impostores a utilizar.

T-norm	Género	Frase	Fonema	Estado
No	Masc	3.54		
	Fem	3.72		
TN10	Masc	3.32	2.64	2.80
	Fem	3.57	3.15	2.45
	Ambos	3.64	2.97	2.91
TN30	Masc	2.69	2.48	2.46
	Fem	3.99	3.67	3.67
	Ambos	3.10	2.98	2.96
TNMale		2.57	2.41	2.53

Tabla 2. *EERs (%) obtenidas para distintos tipos de T-norm en función del nivel y género.*

En la Tabla 2 vemos cómo varían las tasas de error obtenidas en función del género y el número de impostores de la cohorte. En líneas generales podemos observar que parece ser que cuanto mayor es la cohorte de impostores mejor funciona el sistema, debido probablemente a que al tener un número mayor de impostores tenemos más probabilidades de encontrarnos con modelos próximos al del locutor *target*. Sin

embargo esto no se cumple para todos los casos y es debido, muy probablemente, a que también nos encontraremos más modelos que se alejen del modelo del locutor *target*. Por otra parte también vemos que se generaliza la suposición de que es mejor realizar T-norm a nivel de fonema o estado que a nivel de frase. Además vemos que no hay mucha diferencia entre realizarlo a nivel de estado o fonema, ya que hay casos en los que resulta mejor uno que otro y viceversa.

6. OTROS EXPERIMENTOS

A fin de extender nuestra experimentación al idioma castellano realizamos también experimentos con la base de datos BioSec Baseline [7]. Esta base de datos, aparte de permitirnos comparar resultados en inglés y castellano con el mismo entorno experimental nos permite comparar la influencia del canal de grabación (micrófono de habla cercana frente a micrófono de habla lejana) y la influencia de la coincidencia léxica entre entrenamiento y test (cosa que ocurre en BioSec pero no en YOHO). Otra diferencia con los resultados presentados anteriormente es que en los resultados con BioSec se ha empleado únicamente MLLR y no MAP.

Canal\Idioma	Castellano	Inglés
Mic. cercano	1.68	2.17
Mic. lejano	17.24	12.72

Tabla 3. *EER (%) obtenidas para distintos tipos de micrófono e idioma.*

Como podemos observar en la Tabla 3 la calidad del micrófono supone una gran contribución a la eficiencia del sistema de Reconocimiento de Locutor Dependiente de Texto. Vemos que los resultados obtenidos con el micrófono de la webcam son mucho peores que con el micrófono cercano integrado en los auriculares. Se ha de indicar que en estos experimentos no se han utilizado técnicas de compensación de canal de ningún tipo (salvo CMN).

Sin desajuste	7.02
SNR	7.47
Canal	9.76
Léxico (2 dígitos en común)	8.23
Léxico (1 dígito en común)	13.4
Léxico (0 dígitos en común)	36.3

Tabla 4. *EER (%) obtenida para distintos tipos de desajuste para el estudio realizado en [8].*

Por otra parte si nos fijamos en los resultados para micrófono cercano observamos que son mucho mejores para esta base de datos que con YOHO (utilizando la misma técnica sólo con MLLR en YOHO el EER resultante es de 4.82%). La principal diferencia entre ambas bases de datos es el desajuste léxico existente en YOHO e inexistente en BioSec. El problema del desajuste léxico ya se ha analizado con anterioridad [8] y se ha comprobado (ver Tabla 4 con resultados publicados en [8]) que el desajuste léxico puede ser el tipo más perjudicial de desajuste, incluso peor que el de canal.

7. CONCLUSIONES

Dados los resultados obtenidos en los diferentes experimentos se puede concluir que el desajuste léxico tiene una gran influencia en la eficiencia de un sistema de Reconocimiento de Locutor Dependiente de Texto. La principal razón es que en este campo se entrenan modelos de unidades léxicas por debajo de la locución completa (palabra, fonema, tri-fonema, estado…). Y el hecho de que se intente reconocer al locutor con modelos de unidades léxicas que hemos podido no entrenar previamente (desajuste léxico), o que hemos entrenado en contextos léxicos distintos, hace que los resultados empeoren de forma muy abultada.

Demostrado esto, el principal objetivo de la técnica de T-Norm a nivel de fonema y estado era tratar de reducir la influencia del desajuste léxico, para así ponderar la influencia de cada fonema en el proceso de verificación. Aunque los resultados obtenidos con la normalización a nivel de estado y fonema son positivos, superando los resultados a nivel de frase, el problema del desajuste léxico sigue sin estar resuelto y sigue teniendo una influencia importante en los resultados.

8. BIBLIOGRAFÍA

[1] "National institute of standard and technology. Speaker Recognition Evaluation Home Page", http://www.nist.gov/speech/tests/sre/

[2] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition", Proceedings of the IEEE, vol 77, no 2, pp. 257-286, Febrero 1989.

[3] D. A. Reynolds, T. F. Quatieri, R. B. Dunn, "Speaker verification using adapted gaussian mixture models", Digital Signal Processing, vol 10, no 1, pp. 19-41, Enero 2000.

[4] D. T. Toledano, D. Hernandez-Lopez, C. Esteve-Elizalde, J. Gonzalez-Rodriguez, R. Fernandez, L. Hernandez, "MAP and sub-word level T-norm for text-dependent speaker recognition", to appear in Interspeech 2008.

[5] J. P. Campbell, "Testing with the YOHO CD-ROM voice verification corpus", Proc. ICASSP, vol 1, pp. 341-344, 1995.

[6] J. Fierrez, J. Ortega-Garcia, D. T. Toledano, J. Gonzalez-Rodriguez, "Biosec baseline corpus: A multimodal biometric database", Pattern Recognition, vol 40, no 4, Abril 2007.

[7] D. T. Toledano, D. Hernandez-Lopez, C. Esteve-Elizalde, J. Fierrez, J. Ortega-Garcia, D. Ramos, J. Gonzalez-Rodriguez, "BioSec Multimodal Biometric Database in Text-Dependent Speaker Recognition", Proc. LREC, Mayo 2008.

[8] D. Boies, M. Hébert, L. P. Heck, "Study of the effect of lexical mismatch in text-dependent speaker verification", Proc. Odyssey Speaker Recognition Workshop, vol 1, pp. 135-140, Junio 2004.

Turning Wikipedia into a resource for language research

Alberto Montero-Asenjo, Carlos A. Iglesias

Grupo de Sistemas Inteligentes (GSI)
Universidad Politécnica de Madrid, Spain
{amontero,cif}@gsi.dit.upm.es

Abstract

Wikipedia is a valuable resource whose usage goes beyond the encyclopedia itself. In this paper the proposal is to use Wikipedia as a large source of text, suitable for language research, explaining the followed procedure to turn Spanish Wikipedia raw data into a suitable text source, considering the format of source data (wiki syntax), the conversion from written text to individual sentences or the conversion from acronyms or numbers to the way they are said. The case explained here is specific in some parts to the Spanish wikipedia, but the ideas and some steps of the followed procedure can be generalised to any language or text source.

1. Introduction

Language resources (corpus) are usually collected and distributed by dedicated organisations and the cost to the public is usually high or it has restrictions on their applicability. Examples of such corpus for Spanish are CREA [1], *Corpus de la lengua española contemporánea* [2], Argentina [3] or ARTHUS [4]. Up to some extent, price and availability restrictions are caused by the fact that usually this kind of resources are built every time from the ground, and efforts to collect, revise and tag (at several levels) the corpus is huge. But there already are large sources of text data, some of those even free (as in free speech, not as in beer) that could be potentially used to build larger and better databases for language research, without big investments.

Wikipedia [5] is collaborative effort to build a free multilingual encyclopedia. Its name is a portmanteau of the words wiki (a type of collaborative website) and encyclopedia. It was launched in 2001 by Jimmy Wales and Larry Sanger and currently is operated by the non-profit Wikimedia Foundation. It is one of the largest, fastest growing and most popular general reference work currently available on the Internet (accorgind to Wikipedia webpage [5]).

As of December 2007, Wikipedia had approximately 9.25 million articles in 253 languages, comprising a combined total of over 1.74 billion words for all Wikipedias. The English Wikipedia edition passed the 2,000,000 article mark on September 9th 2007, and as of 21 January 2008 it had over 2,185,000 articles consisting of over 950,000,000 words. Wikipedia's articles have been written collaboratively by volunteers around the world, and the vast majority of its articles can be edited by anyone with access to the Internet. Having steadily risen in popularity since its inception, it currently ranks among the top ten most-visited websites worldwide (these figures have been taken from [6]).

Spanish Wikipedia [7] is a much smaller project than the English one. It was founded a few months later than the general project (on May 2001) and at the beginning of 2008 it had more than 300,000 articles and more than 600,000 user from most of the Spanish-speaking countries.

Wikipedia has been used in other scenarios than the encyclopedic search, and it has been previously used as a research resource in fields like semantic research, knowledge extraction or natural language processing [8, 9, 10, 11, 12], where knowledge embedded in Wikipedia was the most valuable resource. This paper proposes not using the knowledge but the text expressing that knowledge, as a representation of language, and up to some extent, speech.

Wikipedia main strengths are its nature of free resource, and thus available to anyone, and its big size (as stated before, Spanish Wikipedia has more than 300,000 articles), thus allowing for a wide variety of words, topics and writing styles. The main weakness is the unsupervised nature, thus not ensuring quality, and requiring some quality control and improvements steps.

Next section (2) is fully devoted to explain the procedure to turn Wikipedia raw data into useful text, section 3 refers to public availability of the generated resources, section 4 draws the main conclusions derived from this work and section 5 is about future lines.

2. Data processing

Data processing consists in a set of steps to convert Wikipedia data into useful text. Figure 1 represents an overview of the process. Basically it has four steps: convert wiki markup to plain text, split paragraphs into sentences (being aware of certain aspects), rewrite sentences as they would be read and, finally, remove incorrect words from vocabulary (and sentences having those words).

2.1. Data source

Every few months, Wikipedia is dumped to a large XML file per language and made publicly available. A dump from Spanish Wikipedia dated by 06/07/2007 was used as raw source data. This dump is not currently available as old dumps are removed (every dump requires about 1Gb of disk space). Latest dumps can be found at [13]. For the processing of the raw XML dump, Perl module Parse::MediaWikiDump (available at CPAN [14]), and data was splitted into articles storing each one in a separate file.

A sample piece of text taken from article "Tebas (Grecia)"("Tebas (Greece)") will be used to illustrate the followed

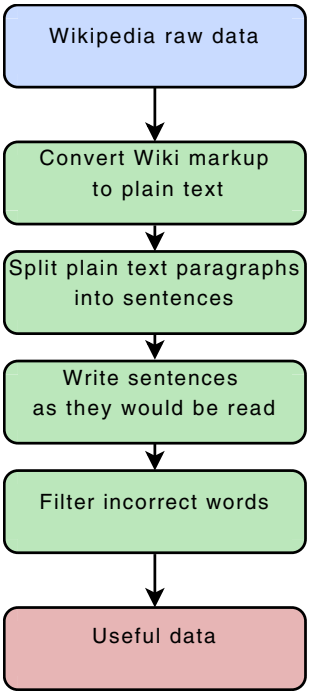


Figure 1: Data processing overview

process, showing the transformations suffered by the text on every step.

Sample
En la actualidad, el lugar de la antigua ciudadela, [[Cadmea]], se encuentra ocupado por la ciudad de Thíva (”Θήβα”) que fue reconstruida después del [[terremoto]] de [[1893]]. La ciudad actual tiene 24.400 habitantes ([[2001]]), llamados ”tebanos”..

2.2. From wiki markup to plain text

Wikipedia XML dumps have data as users wrote it, so it has not only the useful text but many other symbols and references corresponding to wiki syntax. This extra markup must be removed in order to extract valuable text. For this task, Perl module Text::MediawikiFormat (available at CPAN [14]). Despite the simpleness of the wiki markup, there are inevitable syntactic errors not parseable by Text::MediawikiFormat, so it was necessary an extra filtering stage to remove, for example, unmatched brackets or extra '='. This step was also used to beauty text by removing extra spaces and other minor changes. The effect on the sample text denoted above is shown below.

Sample
En la actualidad, el lugar de la antigua ciudadela, Cadmea, se encuentra ocupado por la ciudad de Thíva ($\Theta\eta\beta\alpha$) que fue reconstruida después del terremoto de 1893. La ciudad actual tiene 24.400 habitantes (2001), llamados tebanos.

2.3. Sentences division

Since the corpus is intended to be applied to speech recognition, it is needed to convert it into sentences, instead of written formated text, such as paragraphs, lists, text in parenthesis, enumerations after semicolons and others. These cases have to be addressed in order to use that text. The following points may serve as an example of the followed approach:

- Abbreviations or acronyms are translated into the full words, in order to be process the corpus as a speech corpus.
- Paragraphs were divided into sentences. Dots are the main sentence separator (as well as line or paragraph end), but with some considerations like dots being number separators or part of an acronym (processed previously).
- Text inside parenthesis is consider as different sentences, thus generating two. The first one is the original one without the text inside parenthesis and the second one the text inside parenthesis.

Sample
En la actualidad, el lugar de la antigua ciudadela, Cadmea, se encuentra ocupado por la ciudad de Thíva que fue reconstruida después del terremoto de 1893. $\Theta\eta\beta\alpha$. La ciudad actual tiene 24.400 habitantes , llamados tebanos. 2001.

2.4. Sentences as they would be read

Written text and spoken speech are closely related, but they are not the same. As an example one may consider numbers (either in arabic or roman format). In written text “2001” may appear, while in spoken speech it will be said as “two thousand and one”. But there are many other examples, as mathematical operations, where “+” must be replaced by “plus” or acronyms, which are usually spoken by spelling letters. Finally, capital letters were converted to lower case and commas and other unrecognised symbols were removed. After this step the previous example will remain as follows:

Sample
en la actualidad el lugar de la antigua ciudadela cadmea se encuentra ocupado por la ciudad de thíva que fue reconstruida después del terremoto de mil ochocientos noventa y tres la ciudad actual tiene veinticuatro mil cuatrocientos habitantes, llamados tebanos dos mil uno

2.5. Vocabulary filtering

After the steps explained above a huge amount of text was available. Main figures are shown below.

Number of articles: 69,541

Sentences: 3,280,428

Vocabulary size: 549,962

The dynamic range of the histogram of occurrences is so high that a singe picture cannot show all information. The word having most occurrences is 'de' (in English *of*, *from*), appearing 2,526,038 times. Near half a million words appear less than 30 times. Figure 2 shows a crop of the histogram of vocabulary, considering only words appearing less than 50 times, covering almost 94% of the words. Horizontal axis represents number of occurrences and the vertical one the number of words having that number of occurrences.

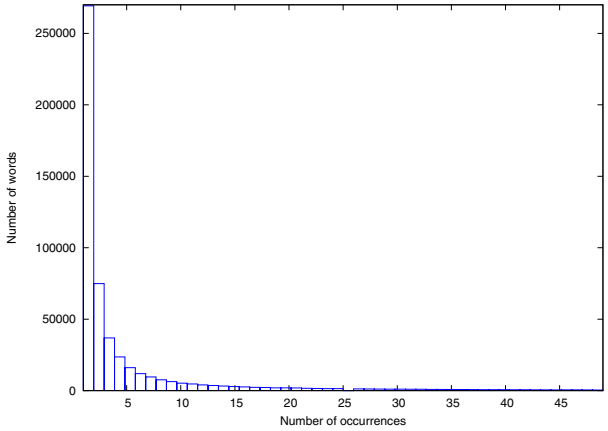


Figure 2: Word occurrences histogram before vocabulary cleaning

The dictionary of Real Academia Española (RAE, the official Spanish language authority), in the 2001 edition [15] has about 90.000 entries, considering near a 10% of archaisms, not including neither all verbal forms nor plural and gender dependent forms for nouns and adjectives. Despite the fact that many words can be composed by adding prefixes and suffixes to standard words, given this figures, more than half a million words seems a huge number.

Due to misspelled words and foreign terms (as propper names, technical words or etymological terms, to cite only a few) it was expected to have a large amount of words, where correct words will occur many times and incorrect ones only a few, making a simple threshold-based decision good enough. But reality is that correct and incorrect words are much more coupled in number of occurrences than expected. This situations makes vocabulary filtering and cleaning a difficult task.

The filtering process consisted in an iterative process of data examination and word removal. When a word was removed, all the sentences where it appeared were removed. Articles with no sentences were erased. The followed heuristics have been defined:

1. Remove words with only one occurrence. These words are considered to be foreign words, misspelled ones or too rare in common Spanish.
2. Remove words with double consonants (bb, cc, dd, ...) and less than 3 occurrences.
3. Remove words with only one occurrence.

Step		Articles	Sentences	Words
Intial	Abs	69,541	3,280,428	549,962
1	Abs	69,541	2,998,212	275,196
	%	100	91.4	50.04
2	Abs	68,679	2,981,025	263,943
	%	98.76	90.87	47.99
3	Abs	68,679	2,959,262	241,180
	%	98.76	90.21	43.85
4	Abs	68,650	2,958,498	241,105
	%	98.72	90.19	43.84
5	Abs	51,925	1,294,040	114,068
	%	74.67	39.45	20.74

Table 1: Remaining data evolution after filtering stages

4. Remove words with the same letter repeated 3 o more times consecutively (aaa, bbb, ccc, ...)

This steps, despite the simplicity, greatly reduced the amount of selected data, keeping only a half of the original vocabulary (241,105 remaining words).

The reduction of available data was very large, but a closer look to the remaining words revealed that there were many terms not valid in Spanish, but difficult to discriminate based on occurrences. A more powerful filtering scheme was needed, and it was achieved by manual inspection of words, identifying words and word patterns not present in Spanish and removing them. For example there is no Spanish words ending with '-ly' (typical in English adverbs) and words ending with '-lae' or '-mae' or starting with 'phy-' are usually Latin words found in technical terms (as species names).

At this point other problems were revealed, not related to words but to character encoding. Wikipedia is primarily UTF-8 encoded, but we found many words having other codification schemes (ISO 8859 1) which made the filtering process a bit harder and we took the decision to remove non UTF-8 encoded words (altough some of them may have been correctly re-encoded automatically and preserved).

Manual revision of near a quarter of million words is a very expensive and time consuming process, so we decide to achieve it iteratively, inspecting a subset on each iteration. The benefits of this approach was that after removing a word and its associated sentences, other potential candidates for removal are automatically removed, thus decreasing the total number of words to inspect. After 8 iterations, over 20,000 words and word patterns were identified and removed from vocabulary.

5. Iteratively inspect vocabulary and select words and word patterns to remove.

Table 1 and figure 3 summarizes the effect of the different filtering stages. In the table you can see the absolute amounts of reamining data and the percentages of the initial dat, while the plot shows the evolution of percentages. As you can see, the most aggresive stage is the last one.

The most common word is still 'de' appearing 367,013 times, and 31,729 words appear only once in the whole remaining text. Figure 4 shows again a crop of the histogram of the remaining data restricted to words appearing less than 50 times, which represents 93% of the words.

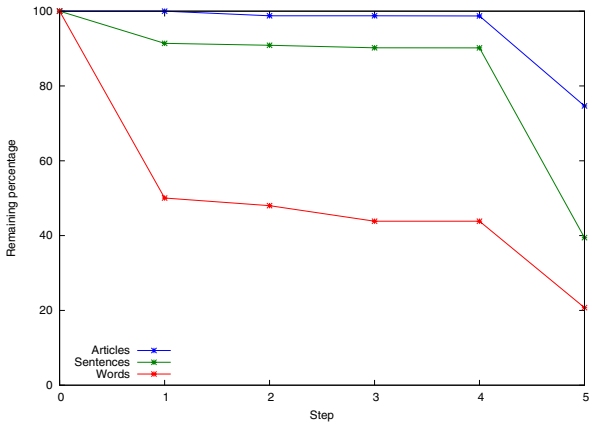


Figure 3: Remaining data evolution after filtering stages

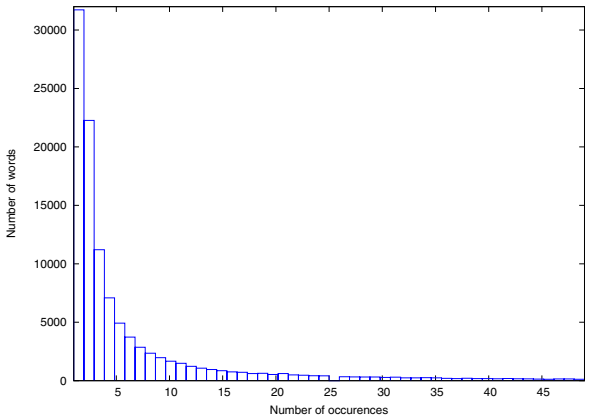


Figure 4: Word occurrences histogram after vocabulary cleaning

3. Data availability

As mentioned above, this work starts from free resources and has made use of many free software tools. To maintain this spirit and to allow others to make improvements and new researches, the generated data as well as the involved scripts have been made public and can be found at `http://wp4lr.sourceforge.net/`. All material can be used and redistributed under the same terms as Wikipedia itself. Any suggestion, improvement or bug detection will be welcomed.

4. Conclusions

Wikipedia is a large source of data but in a format that is not the most usual in the language research community. This paper presents an effort to make that source a valuable resource.

The has been no measures about the goodness of the generated data, as it is difficult to make meaningful comparisons. What other source can be compared to Wikipedia in terms of size and topics covered? How to make such a comparison? Perplexity measures perhaps? And, what should be measured at the end, the data or the process? We have prefer to keep the original goal that was to have a large amount of text for our current researches.

The process to utilise Wikipedia can be automated and re-utilised across languages and text sources up to some extent, but the most difficult and time consumming step (vocabulary filtering) is Spanish specific.

Due to the fact of public availability and continuous improvement of articles, it could be expected to have a mid to high quality resource. But the fact is that many misspelled and badly encoded words were found. Some criticisms have been published about the quality of Wikipedia at semantic level (accuracy, political and ideological bias), and the results found here may be a source for another level of criticism. This fact along with the unsupervised nature of the proposed procedure makes necessary a quality control.

For the purpose of this article, initial quality of data may have reduced the amount of required work and increased the size of available data (as sentences with unappropriated words were fully removed), but even after the extensive filtering stage the amount of data is still very large and suitable for the purposes it was conceived.

5. Future lines

One of the weak points of the followed procedure is related to the vocabulary filtering stage, as it is quite expensive, language specific and its quality is difficult to assess. A way to improve these aspects may be the use of already made dictionaries, to move out sentences containing words not covered by the external vocabulary. Obviously this dictionary has to be large enough as to cover all (or at least mostly) of the vocabulary present in the Wikipedia. Such kind of resources are available for Spanish [1, 16], but as the number of queries that have to be made (more than half a million) is quite large, the process must carefully designed to not overload those sites, and count with the agreement of the site.

6. References

[1] R. A. E. B. de datos (CREA) [on line], "Corpus de referencia del español actual," http://corpus.rae.es/creanet.html, 2008.

[2] S. LABORATORIO DE LINGÜÍSTICA INFORMÁTICA, Universidad Autónoma de Madrid, "Corpus de referencia de la lengua española contemporánea," http://www.lllf.uam.es/corpus/corpus.html.

[3] "Corpus lingüístico de referencia de la lengua española en argentina," http://www.lllf.uam.es/~fmarcos/informes/corpus/coarginl.html.

[4] S. Universidad de Santiago, "Archivo de textos hispánicos de la universidad de santiago," http://gramatica.usc.es/EspArthus.html.

[5] J. Wales and L. Sanger, "Wikipedia, the free encyclopedia," http://www.wikipedia.org, 2001.

[6] "Wikipedia entry for Ẅikipedia," http://en.wikipedia.org/wiki/Wikipedia.

[7] "Spanish wikipedia," http://es.wikipedia.org.

[8] T. Zesch, I. Gurevych, and M. Mühlhäuser, "Analyzing and Accessing Wikipedia as a Lexical Semantic Resource," in *Data Structures for Linguistic Resources and*

Applications, G. Rehm, A. Witt, and L. Lemnitzer, Eds. Tuebingen, Germany: Gunter Narr, Tübingen, 2007, pp. 197–205.

[9] T. Zesch and I. Gurevych, "Analysis of the Wikipedia Category Graph for NLP Applications," in *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT 2007)*, 2007, pp. 1–8.

[10] T. Zesch, I. Gurevych, and M. Mühlhäuser, "Comparing Wikipedia and German Wordnet by Evaluating Semantic Relatedness on Multiple Datasets," in *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2007)*, 2007, pp. 205–208.

[11] F. Wu and D. S. Weld, "Autonomously semantifying wikipedia," in *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. New York, NY, USA: ACM, 2007, pp. 41–50.

[12] S. P. Ponzetto and M. Strube, "Deriving a large-scale taxonomy from wikipedia," in *AAAI*. AAAI Press, 2007, pp. 1440–1445.

[13] "Spanish Wikipedia dumps," http://download.wikimedia.org/eswiki/latest/.

[14] "Comprehensive Perl Archive Network," http://www.cpan.org.

[15] "Spanish RAE dictionary figures," http://buscon.rae.es/draeI/html/drae/cifras.htm.

[16] "Corpus del español," http://www.corpusdelespanol.org/.

USING PITCH AND FORMANTS FOR ORDER ADAPTATION OF FRACTIONAL FOURIER TRANSFORM IN SPEECH SIGNAL PROCESSING

Hui Yin[1,2], Climent Nadeu[1], Volker Hohmann[1,3]

1. TALP Research Center, Universitat Politècnica de Catalunya, Barcelona, Spain
2. Dept. of Electronic Engineering, Beijing Institute of Technology, Beijing 100081, China
3. Medical Physics, Universität Oldenburg, Germany

ABSTRACT

Fractional Fourier transform (FrFT) has been proposed to improve the time-frequency resolution in signal analysis and processing. However, selecting the FrFT transform order for the proper analysis of multi-component signals like speech is still debated. In this work, we investigated several order adaptation methods based on the pitch and formants of voiced speech. This study is motivated by the fact that speech is not stationary even in a short time interval, and the idea is shown using an AM-FM speech model. First, FFT and FrFT based spectrograms of an artificially-generated vowel are compared to indicate the merit of the methods. Second, a tonal vowel discrimination test is designed to compare the performances of the various proposed methods using MFCC features implemented with FrFT.

1. INTRODUCTION

Speech is a non-stationary signal. Traditional speech processing methods generally treat speech as short-time stationary, i.e., process speech in 20~30ms frames. In practice, intonation and coarticulation introduce combined spectro-temporal fluctuations to speech even for the typical frame sizes used in the front-end analysis. Modeling speech signals as frequency modulation signals accords better with speech characteristics from both production and perception views.

From the speech production view, traditional linear source-filter theory lacks the ability to explain the refined structure of speech in a pitch period. Maragos et al. therefore proposed an AM-FM modulation model for speech analysis, synthesis and coding [1]. From the perception view, neurophysiological studies show that the auditory system of mammals is sensitive to FM-modulated (chirpy) sounds. This fact explains the human sensitivity to non-stationary acoustic events with changing pitch (police and ambulance siren) [2].

This research was partially supported by the Spanish project SAPIRE (TEC2007-65470), and a research grant to V.H. from the Spanish Ministry of Education and Science. H.Y. was partially supported by the National Nature Science Foundation of China under Grant NSFC 60605015.

Fractional Fourier transform (FrFT) can be considered as a generalization of the traditional Fourier transform [3]. Since FrFT can be considered as a decomposition of the signal in terms of chirps, FrFT is especially suitable for the processing of chirp-like signals [4]. The chirp rate (temporal derivative of instantaneous frequency) of the FrFT kernel functions is set by one free parameter, the transform order. The determination of the optimal transform orders is always critical. In this paper we show that the representation of the time-varying properties of speech may benefit from using the values of pitch and formants to set the order of the FrFT. Different order adaptation methods based on pitch and formants are proposed in this paper.

In tonal languages as Mandarin, the time evolution of pitch inside a syllable (the tone) is relevant for the meaning. Consequently, there are relatively fast changes of pitch which are usual and informative. As the use of the FrFT might help to better track the dynamic properties of speech harmonics, we have carried out a classification experiment using a small set of Mandarin vowels, where the classes correspond to the four basic types of tones, and the discrimination ability is measured using the MFCC features implemented with FrFT.

The rest of the paper is organized as follows. In section 2, the AM-FM model of speech is described, and the motivation of the proposed method is given. In section 3, the definition and some basic properties of FrFT are briefly introduced. In section 4, different order adaptation methods are described. One method is illustrated using FFT and FrFT based spectrograms of an artificially-generated vowel. In Section 5, a tonal vowel discrimination test is designed, and the results are given and analyzed. Some conclusions and future work are given in section 6.

2. THE AM-FM MODEL OF SPEECH

Considering the fluctuation of pitch and the harmonic structure, voiced speech can be modeled as an AM-FM signal

$$x(t) = \sum_{n=1}^{\infty} a_n(t)\cos(n(\omega_0 t + \int_0^t q(\tau)d\tau) + \theta_n) \quad (1)$$

where $q(t)$ is the frequency modulation function. Making the reasonable simplification that the frequency is changing linearly within the frame, i.e. $q(t) = kt$, where *k* is the chirp rate of the pitch (referred to as *pitch rate* in the rest of the paper), we can obtain:

$$x(t) = \sum_{n=1}^{\infty} a_n(t) \cos(\underbrace{n(\omega_0 t + \frac{1}{2} k t^2) + \theta_n}_{\varphi_n(t)}) \quad (2)$$

The chirp rate of the *n*-th harmonic is the second derivative of the phase function

$$\frac{d^2 \varphi_n(t)}{dt^2} = q_n = nk, \quad (3)$$

which means that the chirp rate of the *n*-th harmonic is *n* times the pitch rate.

3. DEFINITION OF THE FRACTIONAL FOURIER TRANSFORM

The FrFT of signal $x(t)$ is represented as:

$$X_\alpha(u) = F_p[x(t)] = \int_{-\infty}^{\infty} x(t) K_\alpha(t,u) dt, \quad (4)$$

where *p* is a real number which is called the order of the FrFT, $\alpha = p\pi/2$ is the transform angle, $F_p[\bullet]$ denotes the FrFT operator, and $K_\alpha(t,u)$ is the kernel of the FrFT:

$$K_\alpha(t,u) = \begin{cases} \sqrt{\frac{1 - j\cot\alpha}{2\pi}} \exp\left(j \frac{t^2 + u^2}{2} \cot\alpha - jut \csc\alpha \right), & \alpha \neq n\pi \\ \delta(t-u), & \alpha = 2n\pi \\ \delta(t+u), & \alpha = (2n \pm 1)\pi \end{cases} \quad (5)$$

The inverse FrFT is

$$x(t) = F_{-p}[X_\alpha(u)] = \int_{-\infty}^{\infty} X_\alpha(u) K_{-\alpha}(t,u) du \quad (6)$$

Eq.(6) indicates that the signal $x(t)$ can be interpreted as a decomposition to a basis formed by the orthonormal Linear Frequency Modulated (LFM) functions in the *u* domain, which means an LFM signal with a chirp rate corresponding to the transform order *p* can be transformed into an impulse in a certain fractional domain. Therefore, the FrFT has excellent localization performance for LFM signals.

4. ORDER SELECTION METHODS

To test the proposed order selection methods informally, we produced an artificial vowel [i:] with time-varying pitch. The excitation of the vowel is a pulse train with linearly decreasing frequency from 450Hz to 100Hz, and the formants of the vowel are 384Hz, 2800Hz, and 3440Hz, which are extracted from a real female vowel. The sampling rate is 8000Hz.

We experimented three different classes of order adaptation methods based on the pitch and formants. They will be explained in detail and the spectrograms of the artificial vowel based on these methods are shown.

4.1. N times of pitch rate

Since the chirp rates for different harmonics are different, the FrFT is emphasizing the N-th harmonic when setting the transform order according to N times of the pitch rate *k*. The transform angle is determined by:

$$\alpha = \text{acot}(-2\pi * k * N). \quad (7)$$

When the order is set according to N times of the pitch rate, the N-th harmonic and its neighbors will be emphasized, i.e. they have better concentration performance than the FFT-based spectrogram. On the other hand, the representation of harmonics whose chirp rates are not close to 10 times of pitch rates will be smeared. This is also true for the formants, because their frequency variations are generally smaller than the harmonics, i.e., the chirp rates of the formants are generally much smaller than N times of pitch rate when N gets larger.

4.2. Pitch and formants

The sub-band energies that are usually employed to compute the speech recognition features, e.g. in the widely used MFCC, are a representation of the envelope Since the FT-based spectral harmonics are an intermediate step in the computation of the envelope, a more precise representation of the harmonics in relevant regions of the spectral envelope may help to get more accurate formant estimates and also more discriminative speech features. This is the motivation for the order adaptation method based on pitch and formants that is introduced in the following. As in (11), the transform angle is determined by M times of the pitch rate *k*:

$$\alpha = \text{acot}(-2\pi * k * M). \quad (8)$$

M will be computed from the frequency of a formant and the pitch frequency as

$$M = f_{formant} / f_{pitch}. \quad (9)$$

Here, M is different for different analysis frames.

4.3. Multi-order multiplication

Since different optimal orders are needed for different harmonics, we can calculate the FrFT with the orders corresponding to 1, 2, 3… times of the pitch rate and multiply them together. This method can obtain a compromise among several harmonics. Alternatively, in our experiments, multi-order multiplication was also applied to the N FrFT spectrograms that target the first N formants according to the technique described in section 4.2. The resulting multiplied FrFT spectrogram

is shown in figure 1 for N=3 (right panel). In this case, formant smearing is limited, while still enhancing the harmonics going through the formant resonances.

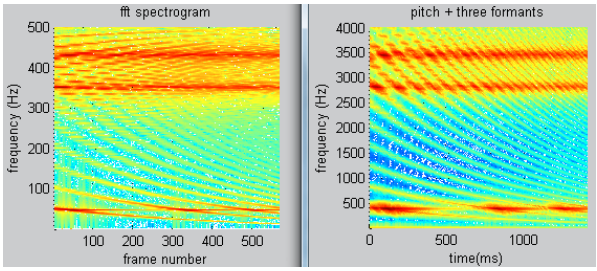


Figure 1: Left panel: *FFT-based spectrogram. Right panel: FrFT-based spectrogram with multi-order multiplication. The order multipliers M1,M2 and M3 (see eq. 9) correspond to the three formant frequencies.*

5. TONAL VOWEL DISCRIMINATION TEST

In Mandarin, there are four basic lexical tones and a neutral tone. The number of tonal syllables is about 1300, and it is reduced to about 410 when tone discriminations are discarded [5]. Fundamental frequency or pitch is the major acoustic feature to distinguish the four basic tones. In order to test the performance of the order adaptation methods, we designed a tonal vowel discrimination test. Since the proposed FrFT order adaptation methods may show a more accurate representation of the time-varying characteristics of the harmonics than the FT, we decided to test them in tone recognition for tonal languages.

5.1. Experiment design

We recorded the five Mandarin vowels [a], [i](yi), [u](wu), [e], [o](wo) with four tones: the flat tone (tone 1), the rising tone (tone 2), the falling and rising tone (tone 3), and the falling tone (tone 4). Each tone of each vowel from a female voice is recorded five times. The utterances are sampled at 8kHz, with a 16bit quantization. We use 16-dimensional standard MFCC features as the baseline. The features based on the FrFT are computed with the same processing used for the MFCCs, but substituting the Fourier transform by the FrFT (we will refer to them as FrFT-MFCC) [6].The performance of FrFT-MFCC using different order adaptation methods is compared with the baseline. Speech signals are analyzed using a frame length of 25ms and a frame shift of 10ms.

Because the recorded utterances have variable lengths, we use Dynamic Time Warping (DTW) to calculate the distances between all the utterances for the individual vowels. Thus, five 20x20 distance matrices are obtained (4 tones, 5 times). The discriminative ability of features can be analyzed using the Fisher score, which is defined as the ratio between the between-class variance and the within-class variance. Here, we take the distances calculated by DTW to compute a similar score (also referred to as Fisher Score):

$$F = \frac{\frac{1}{N_1}\sum_{m=1}^{5}\sum_{n=1}^{5}\sum_{i=1}^{4}\sum_{j\neq i,\,j=1}^{4} dist(v_i^m, v_j^n)}{\frac{1}{N_2}\sum_{m=1}^{5}\sum_{n=1}^{5}\sum_{i=1}^{4} dist(v_i^m, v_i^n)} \quad (10)$$

v_i^m represents the token m of a vowel with tone i. N_1 and N_2 are the total numbers of the between-class and within-class tokens respectively. $dist(\bullet)$ represents the Euclidean Distance. By this analysis, the discriminability across different tones of the same vowel is assessed. The discrimination among different vowels is also assessed here for comparison.

5.2. Pitch rate and formant calculation

The speech is processed in overlapping frames. Each frame is further divided into several non-overlapping sub-frames. One pitch value is detected for one sub-frame. These pitch values are obtained using a robust pitch tracking algorithm described in [7]. In order to get the pitch rate of a frame, we first calculate the median value of the sub-frame pitch values for this frame to set a threshold, if any sub-frame pitch value is larger than twice this threshold, then it is divided by 2. If any pitch value is smaller than half the threshold, it is multiplied by 2. By this, octave confusions are largely eliminated. Then, a straight line was fitted to all the corrected pitch values in this frame. The pitch rate is taken as the slope of this fitted line.

The formants are determined as the frequencies of the LPC-based spectral peaks. The order for LPC analysis is set to be twice the number of formants used in the multi-order FrFT analysis. Note that when the number of formants exceeds 4, they may be not real formants but envelop peaks.

5.3. Experimental results

The Fisher scores for different vowels using the various methods are given in Table 1. In this experiment, the frame length is 25ms and frame shift is 10ms. Every frame is divided into 5 subframes. The experimental results show that FrFT analysis increases the tone discriminability for most of the order selection methods proposed here. We can see that:

(1) The average Fisher score over all vowels using MFCC is 4.43. This indicates that MFCC already has a good discriminability for different tones, but the FrFT-MFCC can get even better results, especially for the multi-order multiplication method with N=1*2*..*5, which obtains nearly 50% improvement. For comparison, the Fisher score for the discrimination of different vowels of the same tone is 12.20 on average across tones. This indicates that the discrimination of tones is more difficult than the discrimination of vowels,

as expected, and that the improvement of tone-discrimination by using the FrFT might provide a large benefit for speech analysis and recognition applications.

(2) When using a single N value for the N times of pitch rate method, the increases of the scores are moderate. Just as stated before, the formants may be dispersed when N gets larger, because the chirp rate of formants is not close to that value. There is always an optimal value of N. Generally N=1~3 can obtain a good compromise between tracking the dynamic speech harmonics and preserving the concentration of the formants.

	a	i	e	o	u	Average
MFCC	2.77	3.94	5.28	4.59	5.56	4.43
N=1	2.63	4.48	6.24	4.90	6.61	4.97
N=2	2.58	4.15	6.07	4.78	6.48	4.81
N=3	2.55	3.95	5.90	4.68	6.38	4.69
N=5	2.49	3.71	5.61	4.52	6.19	4.5
Pitch +MP	2.46	4.76	6	4.77	6.55	4.91
Pitch +2MP	2.25	3.91	5.53	6.94	8.74	5.47
Pitch +3MP	2.27	4.53	5.67	6.23	11.2	5.99
Pitch +5MP	2.44	4.52	5.85	6.00	12.0	6.16
Pitch+ 10MP	2.11	4.21	6.85	4.13	12.7	5.99
N=1*2	2.4	4.63	5.67	6.91	9	5.72
N=1*2*3	2.36	5.41	5.71	6.17	11.6	6.25
N=1*2*..*5	2.46	5.01	5.86	5.96	12.4	6.34
N=1*2*..*10	2.13	4.08	6.83	4.1	12.5	5.93

Table 1: *Fisher scores using MFCC and all variants of the FrFT-MFCC method. MP denotes the main peaks of the LPC spectrum, and Pitch + xMP refers to the technique presented in Section 4.2. When x>1, the transforms are multiplied as explained in Section 4.3 (right panel in Fig. 1).*

(3) The pitch + "formants" method can obtain significantly better results than the method only based on the pitch. Different vowels have their different optimal numbers of formants, e.g. for [u], even using 10 formants its maximum is still not achieved, but for [i], the maximum is achieved using one main formant, and for [o], two formants. The pitch + 5MP method can obtain good results on average for all vowels except [a].

(4) For the vowel [a], the FrFT-MFCC always performs worse than MFCC. This is possibly because the first formant of [a] is much higher than in the other vowels. A higher formant needs a larger N, but a larger N will smear the formant, so a good compromise can't be achieved.

(5) The multi-order multiplication method with different number of N's can significantly increase the scores for vowels [i] [e], [o] and [u] compared with MFCC. These four vowels achieve their best results with different numbers of order multipliers. Here, they are 3, 10, 1, 10 respectively. The best average result of all is obtained using the multi-order multiplication method with N=1*2*..*5.

(6) Compared with the pitch + MP method, the pitch + 2MP method improves the discriminability of FrFT-MFCC for vowels [o], [u], but not for the other three vowels, especially for [i]. The reason for this might be the frequencies of the first two formants of [o] and [u] are low and close, so a significant improvement can be obtained; but it's the opposite for [i], whose first formant is quite low and the second formant is rather high. The smearing effect prevails in the combination of the corresponding two orders. When more "formants" are taken, such situation is somewhat alleviated.

6. CONCLUSIONS AND FURTHER WORK

In this paper, we have proposed several order adaptation methods for FrFT in speech signal analysis and processing, which are based on the pitch and the formants (or just envelope peaks) of voiced speech. The FrFT results with the proposed order selection methods have some improvement over its FFT counterpart. We have also done some preliminary work applying the N times of pitch rate method to speech recognition, and the results show some improvement over the MFCC baseline [8]. Considering the effectiveness of the FFT analysis on formant determination and of the FrFT analysis on emphasizing harmonics, one possible approach is to combine the FFT and FrFT to get an improved representation of speech features for speech analysis and recognition.

10. REFERENCES

[1] P. Maragos, T. Quatieri, and J. F. Kaiser, “On Amplitude and Frequency Demodulation Using Energy Operators,” *IEEE Transaction on Acoustics, Speech and Signal Processing*, *41*(4), pp. 1532-1550, April 1993.

[2] M. Képesi, L. Weruaga, “High-resolution noise-robust spectral-based pitch estimation”, *Interspeech*, Lisbon, Portugal, 313-316, 2005.

[3] Namias V. The fractional order Fourier transform and its application to quantum mechanics. *J Inst Math Appl*, *25*, 1980, 241-265.

[4] Qi Lin, Tao Ran, Zhou Si-yong, “Detection and parameter estimation of multicomponent LFM signal based on the fractional Fourier transform”, *Science in China, 47*(2), 184-198, 2004

[5] Y.-R. Chao, ed., *A Grammar of Spoken Chinese*, University of California Press, Berkeley, 1968.

[6] Yin Hui, Xie Xiang, Kuang Jingming, "Adaptive-Order Fractional Fourier Transform Features for Speech Recognition", *Interspeech*, Brisbane, Australia, 2008

[7] D. Talkin, "A Robust Algorithm for Pitch Tracking (RAPT)" in "Speech Coding & Synthesis", W B Kleijn, K K Paliwal eds, Elsevier ISBN 0444821694, 1995.

[8] Hui Yin, Climent Nadeu, V. Hohmann, et al. "Order adaptation of the fractional Fourier transform using the intraframe pitch change rate for speech recognition". ISCSLP, Kunming, China, 2008.

iATROS: A SPEECH AND HANDWRITING RECOGNITION SYSTEM

Míriam Luján-Mares, Vicent Tamarit, Vicent Alabau,
Carlos-D. Martínez-Hinarejos, Moisés Pastor, Alberto Sanchis, Alejandro Toselli

Instituto Tecnológico de Informática, Universidad Politécnica de Valencia,
Camino de Vera, s/n, 46022, Valencia, Spain

ABSTRACT

Speech technologies have developed in the last twenty years and now allow the implementation of real-world speech applications. Furthermore, handwriting recognition has gained attention in the last years due to their multiple applications and the opportunity of reusing the consolidated speech technology for that problem. In this work we present the implementation of the modules of a flexible recognition system, the iATROS system, which allows speech and handwriting input. The iATROS system is developed in a modular manner, with a core recognition engine and several utility functions that can be used in the construction of speech and handwriting-based applications, including multimodal and interactive applications. We show the capabilities and features of the modules and present a few schemes on how the modules can be used to build applications.

1. INTRODUCTION

In the last twenty years, the speech recognition systems have become widely available to research scientists and nowadays they are quite present in real world. Some free speech recognizers based on Hidden Markov Models (HMM), like Sphinx [1] or HTK [2], are available for the speech processing research community, which uses and modifies them to experiment with different techniques to enhance the speech recognition performance. These recognizers can be used on the construction of speech-based applications, but with some limitations due to the difficulty of integration with other software applications and possible license restrictions.

Parallel to the speech recognition development, text recognition has gained interest in the last years for its applications: automatic processing of forms [3], handwriting transcription [4], transcription of ancient books [5], etc. A few years ago, handwriting text recognition started to base on the same technology as speech recognizers (HMM-based). Therefore, many speech recognizers have been adapted by the handwriting text recognition researchers to cope with this new task.

In this work we present a new recognizer which allows the recognition of both speech and handwriting signals, the iATROS[1] recognizer. iATROS is composed of two preprocessing and feature extraction modules (for speech signal and handwriting images) and a core recognition module. The preprocessing and feature extraction modules provide feature vectors to the recognition module, that using HMM models and language models performs the search for the best recognition hypothesis. All the modules are implemented in C.

Since the iATROS system accepts both speech and handwriting signal, it is possible to build multimodal applications based on this system. The flexibility of the core recognition module allows the implementation of many applications based on this system.

The paper has the following content: Section 2 presents the speech preprocessing and feature extraction; Section 3 presents the handwriting images preprocessing and feature extraction; Section 4 describes the basic recognition process for the core recognizer; Section 5 describes a few examples on how to develop applications based on the iATROS system; Section 6 presents some concluding remarks and future plans to improve and use the iATROS system.

2. SPEECH PROCESSING

The iATROS sound system is based on the ALSA[2] sound modules. The software package includes record software for both online and offline recognition. The source code includes functions to load, save and play sounds. The software can read three different sound formats: raw data, AD files (defined by the PRHLT group) and WAV files without compression. The output for a recorded sound is only raw data, because we think is the most compatible format.

iATROS includes a feature extraction program based on the mel cepstral coeficients [6]. The architecture of the preprocess module is quite simple and reproduces the

WORK PARTIALLY SUPPORTED BY THE SPANISH RESEARCH PROGRAMME CONSOLIDER INGENIO 2010: MIPRCV (CSD2007-00018), BY SPANISH MEC AND FEDER UNDER PROJECT TIN2006-15694-C02-01, BY THE GENERALITAT VALENCIANA UNDER GRANT GVPRE/2008/331 RESEARCH PROJECT "UPENNSPANISH" AND BY VIDI-UPV UNDER GRANT FPI-PAID06 AND PROJECT 20070315.

[1] iATROS stands for improved Automatically Trainable Recognizer of Speech.

[2] Advanced Linux Sound Architecture

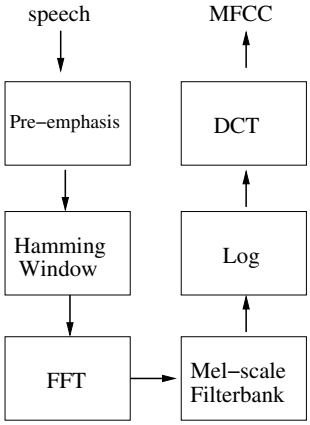


Figure 1. Process diagram for each frame of audio signal.

typical extraction process used in speech recognition. The audio signal is processed by moving a window over it; this portion of signal covered by the window is called *frame*. For each frame we compute its cepstrum coefficients using the modules shown in Figure 1. These modules are separate functions, with no dependencies between them except the input and output data. This modularization allows to easily modify the feature extraction process by changing the modules or adding new ones.

The functions use a structure which stores all the information needed in the process. These parameters are initially loaded from a configuration file. The values that affect the feature extraction and can be modified by the user are: size of the preprocess window, sample frequency, audio channels, number of coding bits, subsample frequency, length of the FFT, pre-emphasis factor, number of cepstrals, silence threshold, and duration of silence.

One of the most important parts of a feature extraction system is the Fast Fourier Transform. We used the FFTW3 library [7]. This library is free software and is one of the most efficient ways to compute the FFT. Another important piece in the preprocess pipe is the estimation of the Mel Filter Bank. Software like Sphinx [1] computes the filters in real time, but we decided to do that work offline because the automatical estimation is complicated (manual tuning is usually requiered). Moreover, it is not necessary to compute the filters for each run, since the filters only depend on the sample frequency.

The feature vector is formed by the cepstrum coefficients and an extra element, the frame energy. This value is a global measure for the frame and is computed as the first element of the Discrete Cosine Transform. The output of the feature extraction module is in plain text format with a header that indicates the number of cepstrum features and vectors, as well as other parameters.

3. HANDWRITING PROCESSING

The process starts from a PGM image. The following steps take place in the text preprocessing module. First, a conventional noise reduction method, and skew correction are applied on the whole document image. Its output is then fed to the text line extraction process, which divides it into separate text lines images. Finally, slope and slant correction, and size normalization are applied on each of these separate lines. More detailed description of this preprocessing can be found in [8, 9].

As our recognition system is based on HMM, each preprocessed line image is represented as a sequence of feature vectors. To do this, the feature extraction module applies a grid to divide line image into $N \times M$ squared cells ($N = 20$ is an usual value and M must satisfy the condition $M/N = \textit{original image aspect ratio}$). From each cell, three features are calculated: normalized gray level, horizontal gray level derivative and vertical gray level derivative. The image context is taken into account in this process. The way these three features are determined is described in [10]. Columns of cells (also called *frames*) are processed from left to right and a feature vector is constructed for each frame by stacking the three features computed in its constituent cells. Hence, at the end of this process, a sequence of M $(3N)$-dimensional feature vectors (N normalized gray-level components and N horizontal and N vertical derivatives components) is obtained. In Figure 2 is shown graphically an example of feature vectors sequence x.

4. DECODING PROCESS

The decoding process is performed by using the Viterbi algorithm. In this process, the most likely sentence is searched in a network that integrates the morphological (HMM), lexical and syntactic models. The network is composed of states. In each state, three types of transitions can be distinguished, according to the model that is involved.

The states pertaining to a recognition stage are stored in a heap, whose size is determined by a configuration parameter. A hash table is used to allow an efficient search for the states. Each state has the following essential information on the current:

- State of the language model
- History
- State of the morphological model
- State of the lexical model

The search uses two types of pruning:

- Histogram pruning: this pruning is provided by the size of the heap that stores the current stage; when the heap is full, the probability of the new generated state p_n is compared to the probability of the state in the heap with lowest probability p_l; if $p_n \leq p_l$, the new state is not introduced into the heap; if $p_l < p_n$, the state with p_l gets lost; therefore, an implicit pruning is performed.

Figure 2. Graphical representation of the feature extraction of the text image *"antiguos ciudadanos que en castilla se llamaban"*. The first corresponds with the normalized grey level features, whereas the second and third with the horizontal and vertical derivatives features respectively.

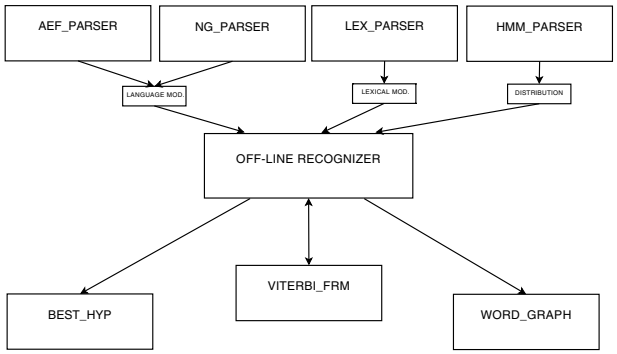


Figure 3. An example of organization of the iATROS modules.

- Beam-search: when a new state presents a probability that is lower than the probability of the current best state divided by the beam factor, the new state is not introduced into the next stage.

5. BUILDING APPLICATIONS: SOME EXAMPLES

The core of the recognizer is the main recognition function (viterbi_frame). This function receives as basic input the feature vector, the set of models (morphological, lexical and language) and the current recognition stage (heap of states), and produces as output the new stage using the search presented in Section 4. Apart from that function, iATROS provides one parser for each type of model, a function to calculate the word-graph and a function to calculate the best hypothesis (Figure 3).

The parsers have the current features:

- Language model parser: supports N-grams (in ARPA format) and Finite State Models (FSM); the language model is loaded in the same structure for both types of language models.
- Lexical model parser: supports FSM with alternative pronunciations.
- Morphological models: supports continuous density HMM in the HTK format, with gaussians as output distributions in the states; it is possible, with little effort, to modify the format, the parser and the recogniser to support HMM with other non-gaussian output distributions in the states.

The main advantage of this recognizer is that it is very easy to build new applications, since the main function of the recognizer has a stable and well defined interface, as well as the auxiliary functions. Therefore, new speech and handwritten text applications can be implemented by using the basic iATROS functions and implementing auxiliary functions that process the results provided by the iATROS functions.

Some examples of applications that can be build based on iATROS are presented in the following subsections.

5.1. Off-line recognition

To carry out off-line recognition only some steps are necessary:

- Read the configuration file.
- Load the models: morphological, lexical and syntactic models.
- For each sentence to be recognized:
  - Analyze frame to frame.
  - Optionally: obtain the word-graph.
  - Return the best hypothesis.
- Free memory and end processes.

This application is actually implemented in a small piece of code.

5.2. On-line speech recognition

For the on-line speech recognition task, the audio system must be initialized and used to feed the recognizer with frames. The on-line recognizer follows this scheme:

- Read the configuration file.
- Load the models: morphological, lexical and syntactic models.
- Init audio system.
- While user does not finish the process:
  - Wait for audio input.
  - While input cepstra are present, analyze frame to frame.
  - Return the best hypothesis.
- Free memory and end processes.

5.3. Combining handwritten text and speech recognition

This application is actually not implemented, but it is shown as an easy example of construction of a multimodal application based on iATROS. In this case, an image representing a text is presented to the user, who utters the corresponding words to the recognizer. The recognizer uses both types of inputs (multimodal input) to enhance the recognition. The scheme for this kind of application is the following:

- Read the configuration.
- Load the models: morphological (only text), acoustic (only speech), lexical and syntactic models (common).
- For each sentence to be recognised:
  - Analyze text input frame to frame.
  - Obtain word-graph for text recognition.
  - Init audio system.
  - Wait for audio input.
  - Analyze speech input frame to frame.
  - Obtain word-graph for speech recognition.
  - Process text and speech word-graphs to return the best common hypothesis.
- Free memory and end processes

6. CONCLUDING REMARKS

In this article we have introduced the basic architecture and modules than form the iATROS system. We presented the steps that are used in speech and handwritten text process to obtain the feature vectors that can be processed by the decoder. We showed the basic features of the decoder, as well as the different models and formats that can be used.

We presented some applications based on the iATROS system: two basic recognizers (on-line and off-line), whose implementation is quite simple (only short programs are requiered to manage the iATROS functions) and a multimodal recognizer that allows both handwritten text and speech. This last recognizer should be implemented as future work, but the presented scheme shows that its construction is not difficult at all.

Future work is directed to the implementation of new applications based on iATROS, as well as its use in new tasks and experiments. Some improvements on temporal complexity and the addition of new features (e.g., to allow not-gaussian output distributions) will be done at the internal level. The implementation of new applications could be used to improve the core system with new utility functions (e.g., recognition with confidence measures, speaker adaptation, etc.).

7. REFERENCES

[1] Xuedong Huang, Fileno Alleva, Hsiao-Wuen Hon, Mei-Yuh Hwang, and Ronald Rosenfeld, "The SPHINX-II speech recognition system: an overview," *Computer Speech and Language*, vol. 7, no. 2, pp. 137–148, 1993.

[2] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book for HTK V3.2*, Cambridge University Press, Cambridge, UK, 2004.

[3] A.C. Downton, A. Amiri, L. Du, and S.M. Lucas, "A configurable toolkit approach to handwritten forms recognition," in *IEE Coll. on Doc. Image Processing and Multimedia Environments*, Nov 1995.

[4] Alessandro Vinciarelli, Samy Bengio, and Horst Bunke, "Offline recognition of unconstrained handwritten texts using hmms and statistical language models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 709–720, 2004.

[5] V. Romero, A. H. Toselli, L. Rodríguez, and E. VidalA, "Computer Assisted Transcription for Ancient Text Images," in *International Conference on Image Analysis and Recognition (ICIAR 2007)*, vol. 4633 of *LNCS*, pp. 1182–1193. Springer-Verlag, Montreal (Canada), August 2007.

[6] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition*, Prentice Hall PTR, 1993.

[7] Matteo Frigo and Steven G. Johnson, "The design and implementation of FFTW3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005, special issue on "Program Generation, Optimization, and Platform Adaptation".

[8] Moisés Pastor, Alejandro Toselli, and Enrique Vidal, "Projection profile based algorithm for slant removal," in *International Conference on Image Analysis and Recognition (ICIAR'04)*, Porto, Portugal, Sept. 2004, Lecture Notes in Computer Science, pp. 183–190, Springer-Verlag.

[9] V. Romero, M. Pastor, A. H. Toselli, and E. Vidal, "Criteria for handwritten off-line text size normalization," in *Procc. of The Sixth IASTED international Conference on Visualization, Imaging, and Image Processing (VIIP 06)*, Palma de Mallorca, Spain, August 2006.

[10] A. H. Toselli, A. Juan, D. Keysers, J. González, I. Salvador, H. Ney, E. Vidal, and F. Casacuberta, "Integrated Handwriting Recognition and Interpretation using Finite-State Models," *Int. Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 4, pp. 519–539, June 2004.

VALORACIÓN DEL CIERRE NEO-GLÓTICO EN VOZ ESOFÁGICA

Roberto Fernández-Baíllo[1], Pedro Gómez[1], Bartolomé Scola[2], Carlos Ramírez[2]

[1]Laboratorio de Comunicación Oral (GIPASI) Universidad Politécnica de Madrid, Campus de Montegancedo, s/n 28660 Boadilla del Monte, Madrid. e-mail: roberto@junipera.datsi.fi.upm.es

[2]Servicio ORL, Hospital Provincial Universitario Gregorio Marañón, C/ Doctor Esquerdo. Madrid.

RESUMEN

Las características de la voz esofágica hacen que su estudio a través de un análisis acústico tradicional sea complicado y limitado. Estas limitaciones son mayores cuando se trabaja con pacientes que no tienen un gran dominio de la técnica. Sin embargo, el rehabilitador necesita obtener información sobre la mecánica desarrollada por el paciente para la producción de la voz esofágica. Ya que el mecanismo de producción en voz esofágica a diferencia de la voz laríngea no es universal ni tan transparente. Cada paciente, debido a los cambios anatómicos que afectan al esfínter cricofaríngeo (ECF) y a las pérdidas funcionales derivadas de la cirugía, desarrolla diferentes estrategias para producir voz. Por todo ello, es fundamental que los clínicos puedan contar con nuevos instrumentos para valorar la calidad de la voz esofágica, que a su vez le aproximen al conocimiento de la dinámica del ECF. El presente trabajo realiza una descripción de la voz de cuatro pacientes laringectomizados basada en el estudio del perfil de la onda neo-glótica. Se muestran los resultados obtenidos tras analizar las fases de abierto-cerrado y la tensión del cuerpo muscular a nivel del ECF.

1. INTRODUCTION

El paciente laringectomizado es aquel que debido a un proceso cancerígeno ha sido sometido a una intervención quirúrgica cuyo resultado ha sido la extirpación total de la laringe. Como consecuencia de la operación el paciente sufre una serie de modificaciones anatómicas que conllevan la alteración y/o pérdida de determinadas funciones. Sin duda, la limitación principal a la que se enfrentan estos pacientes es a la pérdida de la voz y por tanto de la comunicación oral. Es por ello, que el proceso de tratamiento y rehabilitación post-cirugía tiene como principal objetivo el restablecimiento de la comunicación. En estos casos el rehabilitador enseña a al paciente un nuevo modelo de producción de voz, llamado voz erigmofónica o esofágica. Está voz sigue el mismo principio general descrito para la voz laríngea, se trata de aprovechar el cierre ocasionado por un esfínter muscular para impulsar una columna de aire hacia el tracto vocal (1). En voz esofágica se utiliza el ECF como fuente de vibración (neo-glotis) y se impulsa el aire previamente almacenado en el esófago. Una de las limitaciones que presenta la voz esofágica radica en la cantidad de aire que puede ser almacenar en el esófago, aproximadamente unos 50ml (2) en habla no esofágica, cantidad que durante el habla esofágica puede aumentar a 94ml (3). La limitación al almacenar energía ocasiona que el habla esofágica sea lenta, ya que continuamente debe ser interrumpida para coger aire. Otra de las restricciones importantes de la neo-fonación radica en que el ECF tiene una dinámica muy distinta a las cuerdas vocales. En el caso del ECF el esfínter muscular es de tipo constrictor y el cierre lo consigue por un fenómeno de "estrangulamiento" de la luz descrita por la neo-glotis. Además hay que considerar la capacidad para controlar el cierre y el grado de tensión muscular es relativo. Por último, es importante también considerar que la mucosa a nivel esofágico es diferente a la mucosa de los pliegues vocales. Ambas comparten un epitelio estratificado en su capa superficial, pero la organización a nivel más profundo difiere.

La calidad de la voz esofágica está muy relacionada con el mecanismo utilizado para la producción de la voz (4). Actualmente, los estudios de la dinámica del ECF se realizan mediante métodos basados en técnicas radiográficas, manometría (5) y videofluoroscopia (6). Estos métodos tienen varias limitaciones, son invasivos y ninguno de ellos aporta conjuntamente datos de dinámica de la neoglótis y de calidad de voz.

El presente trabajo se basa en un método de filtrado inverso utilizado para la estimación de la fuente glótica en voz laríngea (7) con el objetivo de establecer el correlato dinámico del ECF. Para ello se asume que la ubicación ECF es similar a la de los pliegues vocales y que en ambos durante la fonación se produce un fenómeno de onda mucosa de similares características.

2. MÉTODO

2.1 Obtención de muestras.

El estudió se llevó a cabo con cuatro registros de voz esofágica enviados por el Servicio de ORL del Hospital Gregorio Marañón para su análisis. Se conservó el etiquetado de los pacientes tal y como fueron derivados por el centro hospitalario. La única información aportada era referente al sexo y la técnica esofágica. Todos los pacientes conseguían una voz esofágica a través del

método de deglución, excepto uno de ellos que utilizaba una estrategia basada en la inyección (8) (9). Igualmente se realizó una clasificación de los pacientes en función de la aceptabilidad de la voz esofágica tras una valoración perceptual por un experto en voz del componente de tensión y ruido según la escala GRABS (10) *(Ver tabla I)*. El protocolo de registro incluía una grabación de la vocal /a/ mantenida. Todos los pacientes tuvieron un tiempo de fonación inferior a 1,5 seg. Posteriormente se seleccionaron 0,2 seg. para el procesamiento y extracción del correlato de fuente neo-glótica y onda mucosa (7). Tanto para la grabación como para el procesamiento de la señal se utilizó el software GLOTTEX® (11) (12) *(Figura 2)*.

Tabla I. Clasificación de los pacientes.

Nº Traza	Técnica	Tensión	Ruido
ES-1	Deglución	Alta	Bajo
ES-3	Deglución	Baja	Alto
ES-4	Deglución	Muy Alta	Alto
ES-7	Inyección	Alta	Bajo

2.2 Estimación de la fuente neo-glótica y del correlato de onda mucosa (GLOTTEX®).

El modelo de producción de la voz esofágica es similar al de la voz laríngea. De tal forma que la voz esofágica se define como el resultado irradiado por los labios de una onda generada por la vibración del ECF a su paso de una columna de aire procedente del esófago y que ha sido modificada en los órganos de la resonancia.

El esfínter ECF es un anillo muscular establecido entre el cricoides (límite inferior de la laringe) y el músculo constrictor inferior de la faringe. En función de lo dicho se puede concluir que a priori la longitud del tracto vocal en el paciente laringectomizado será similar a la del sujeto con voz laríngea.

Por tanto, en el paciente laringectomizado podemos aplicar el mismo filtrado de señal que se utiliza para voz normal (7), ya que el modelo de tracto vocal no sufre grandes variaciones. El resultado que obtenido después de eliminar la influencia del tracto vocal es una señal neo-glótica resultante de la vibración del ECF.

Si bien la dinámica cierre es muy diferente entre los pliegues vocales y el ECF, ya que este actúa como un músculo orbicular. Una vez conseguido el cierre el comportamiento dinámico de ambos sistemas es muy similar. En ambos casos se diferencia un cuerpo muscular y un componente de cubierta. La vibración, tanto los pliegues vocales como el ECF, se produce por el paso de una columna de aire que desplaza los componentes superiores e inferiores de la cubierta con distinta fase. Así pues el mismo modelo de 3-masas (7) utilizado para aproximarnos a la dinámica de los pliegues vocales es válido para explicar el comportamiento del ECF. El modelo de cuerpo-cubierta nos permite conocer las tensiones establecidas en cada uno de dichos elementos durante la producción de la voz. De tal forma que tensiones elevadas en cubierta disminuyen el componente de onda mucosa y tensiones altas en cuerpo se relacionan con elevada presión de cierre.

El procesamiento de la señal de voz basada en la metodología expuesta mediante el software GLOTTEX, genera una onda correlato de la fuente glótica (o neoglótica en caso de voz esofágica) en la que se diferencian dos componentes (Fig. 1): una señal de duración de un ciclo de fonación completo (intervalo entre dos cierres glóticos consecutivos, incluyendo la fase de cierre y la de apertura) denominada por Titze (13) como la onda acústica promedio, y una componente que conserva los contenidos de alta frecuencia, que se denomina componente dinámica de cubierta o también correlato de onda mucosa. En la *figura 1* se muestra un ejemplo para voz laríngea y en la *figura 2* los resultados obtenidos con voz esofágica.

El software utilizado permite la posibilidad de extraer una serie de puntos singulares en el perfil de la onda

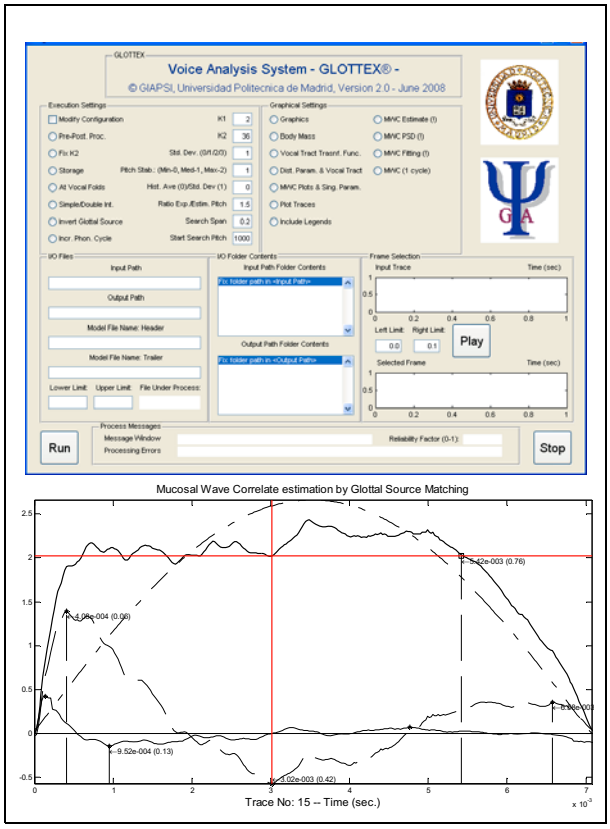


Figura 1. La imagen superior muestra la pantalla principal del software utilizado para la captura y procesamiento de la señal. La imagen inferior se corresponde con el perfil de onda glótica para una voz laríngea no patológica masculina. En la imagen aparecen marcados los puntos de referencia para el estudio de las fases de abierto-cerrado. Se observa que el modelo se aproxima a lo descrito por el modelo L-F (14).

glótica. Posibilitando así calcular las fases de abierto-cierre tomando como referencia lo esperado para un modelo L-F (14) *(Figura 1)*. El software también ofrece una serie de parámetros relacionados con las masas, rigidez y las pérdidas de energías ocurridas en el cuerpo y la cubierta (7) *(Figura 2).*

3. RESULTADOS Y DISCUSIÓN

El perfil de onda glótica para la voz laríngea no patológica se caracteriza por ser próximo al descrito en el modelo L-F (14). En estas curvas el valor máximo de amplitud se encuentra en fase de abierto (Pho), siendo la amplitud en fase de cerrado (Phc) de tendencia uniforme e inferior a la reflejada en Pho. Esta es una condición que está relacionada con la calidad del cierre. El estudio del perfil de la fuente neo-glótica en los pacientes con voz esofágica derivó en distintos resultados los cuales estaban relacionados con la técnica utilizada en la producción de la voz erigmofónica.

Paciente	Tono (HZ)		Masa (g)	Rigidez (g/seg^2)
ES-1	60	B	0.12	15000
		C	0.09	15000
ES-3	61	B	0.04	6500
		C	0.05	3800
ES-4	67	B	0.5	100000
		C	0.08	55000
ES-7	75	B	0.03	7500
		C	0.02	2500

Tabla II. Muestra los valores medios obtenidos en la estimación del tono, masa y rigidez. Se muestran separadamente los valores de masa y rigidez para el cuerpo del esfínter neo-glótico (B) y para la cubierta o mucosa (C).

El estudio de la onda neo-glótica en el paciente ES-1 se aprecia la irregularidad existente en la producción de la voz. Se obtienen dos ciclos consecutivos que son diferentes (Ver figura 2). El segundo ciclo (C2) es algo más próximo al modelo L-F (14), mientras que el primer ciclo se aleja del modelo normativo y se caracteriza por un marcado defecto de cierre. La irregularidad en la producción es una característica intrínseca al propio mecanismo de producción de la voz esofágica. Así el control de la técnica va dirigido a lograr un cierre efectivo y regular. En los pacientes ES-3 y ES-4 se aprecia una mayor simetría entre ciclos consecutivos (C1 y C2).

Una de las características más notorias de la voz erigmofónica es el tono. El cual adquiere unos valores muy bajos con independencia del género. Este es un hecho difícil de superar para el paciente y el rehabilitador ya que es consecuencia directa de la anatomía del ECF. El ECF aporta una mayor masa tanto de cuerpo como de cubierta. Además hay que considerar que el mecanismo de cierre es orbicular lo cual ocasiona que la neo-glotis tenga una mayor superficie de contacto en sentido cráneo-caudal. Con independencia de estas limitaciones anatómicas hay que considerar que la mecánica de cierre, y por tanto la masa involucrada en el mismo, es dependiente de la técnica de voz. En la tabla II se muestran los valores medios del tono para los pacientes estudiados. Se puede apreciar como todos tienen un registro muy grave que está por debajo del umbral de género. Los paciente ES-1, ES-3 Y ES-4 tienen unos valores próximos tono localizados entre 60-70Hz. Todos ellos utilizan la técnica de la deglución, la cual si es efectiva implica una mayor aportación de masa por parte del ECF. El paciente ES-4 consigue un tono de 67 Hz, el más elevado de los tres, pero para ello ha tenido que desarrollar una rigidez máxima. El paciente ES-7 utiliza la técnica de la inyección y presenta el valor de tono más elevado (75Hz). Esto es debido a la propia mecánica de la técnica que aprovecha la tensión articulatoria para introducir el aire en el esófago. El resultado es un cierre más relajado y con un tono más alto al involucra menos masa muscular y mucosa a nivel del ECF. Es decir, al igual que en la voz fonada existe una relación directa entre tono, masa y tipo de cierre glótico.

El estudio de la rigidez revela datos que corroboran todo lo anteriormente expuesto. Los cierres más absolutos implican una mayor tensión tanto en cuerpo como en cubierta. La estimación de la sección del tracto vocal (Ver figura 2) es un tipo de estudio muy interesante para poder acercarnos a la biomecánica particular del paciente durante la producción de la voz esofágica.

En el paciente ES-1 el estudio de la sección del tracto vocal permite estimar la localización de la neo-glotis 20 cm desde los labios. El segmento del tubo esofágico utilizado para la fonación es de unos 13 cm. con una sección transversal del 15% en relación a la apertura máxima.

Los resultados para ES-2 y ES-3 son muy parecidos a los obtenidos en ES-1. Únicamente destacar que en estos casos el valor de la sección del tubo esofágico aumenta tomando valores del 20-30%. En el caso de ES-4 que es la voz calificada como más tensa, tanto por la valoración acústica como por el resultado de la estimación de la rigidez en cuerpo y cubierta, se aprecia que la zona de vibración, neo-glotis, es más extensa verticalmente lo cual viene a confirmar lo anteriormente expuesto al hablar del tono. Los resultados para el paciente ES-7 son diferentes. La estimación de la zona neogótica es más difusa y la participación del tubo esofágico durante la producción de la voz es menor. Esto es debido a que por la técnica vocal utilizada, como ya se ha comentado, no necesita un cierre tan tenso, ni almacenar las mismas cantidades de aire que los otros pacientes ya que se aprovecha para ello de las oclusiones articulatorias.

4. CONCLUSION

En voz esofágica hay que buscar aquellas conductas vocales que implican un cierre efectivo y que no implique una excesiva rigidez a nivel del ECF. Favoreciendo el

uso de aquellas técnicas vocales que favorecen el incremento del tono y la agilidad en la producción.

El estudio de la fuente neo-glotica aporta importantes datos sobre la biomecánica del ECF. Estos datos son de gran importancia para la evaluación de la voz en este tipo de paciente y fundamentalmente para la planificación y desarrollo de programas de rehabilitación y pedagogía vocal.

AGRADECIMIENTOS

Este trabajo ha sido realizado gracias al **TIC2003-08756**, **TEC2006-12887-C02-00** del Plan Nacional de I+D+i, Ministerio de Educación y Ciencia, **CCG06-UPM/TIC-0028** del Plan Regional de Investigación Científica e Investigación Tecnológica de la Comunidad de Madrid, y el proyecto **HESPERIA (http.//www.proyecto-hesperia.org)** del Programa CENIT, Centro para el Desarrollo Tecnológico Industrial, Ministerio de Industria.

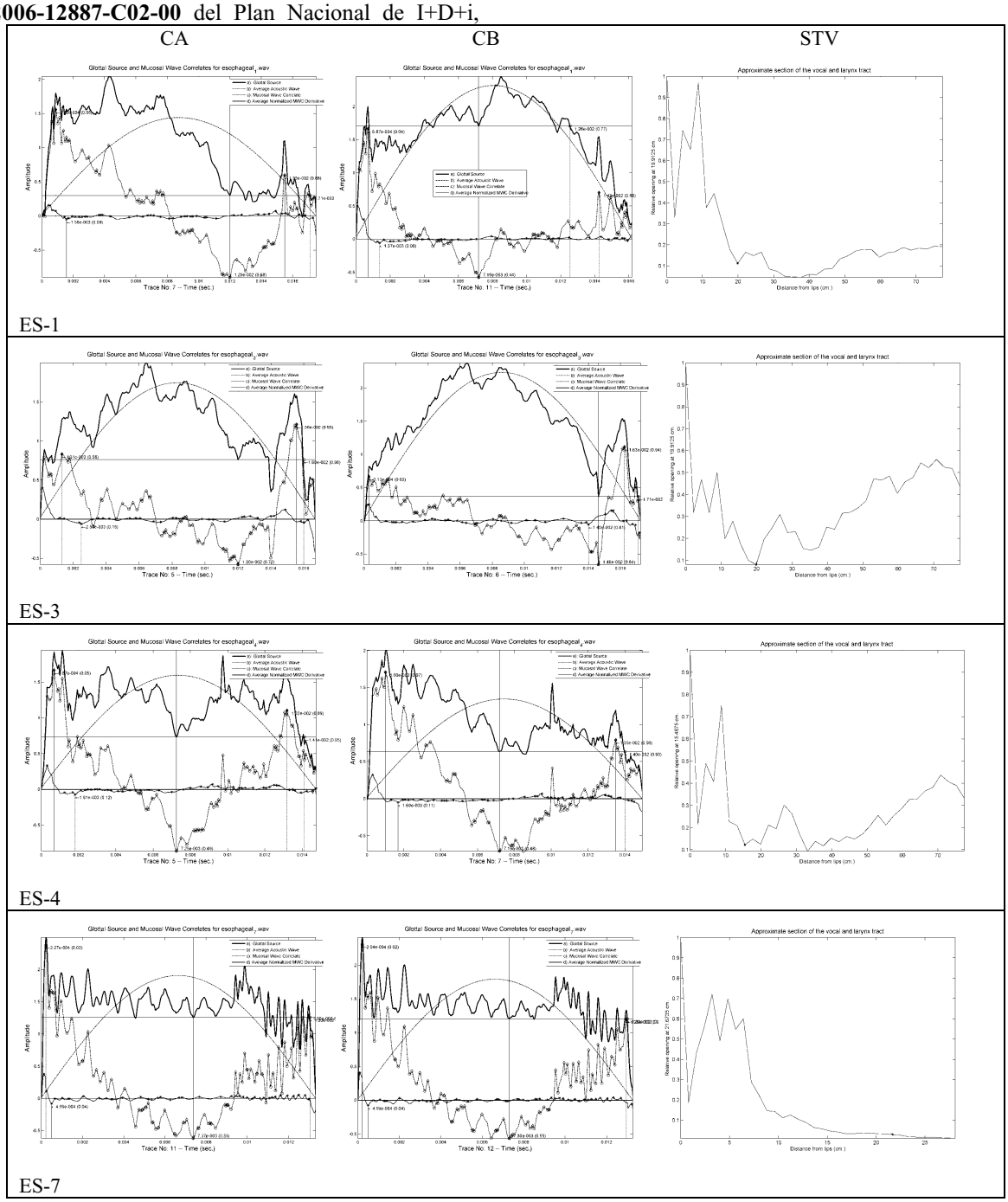


Figura 2. Se muestran los perfiles de onda neo-gloticas obtenidos para cada paciente en dos ciclos consecutivos (C1 y C2). La columna STV muestra los perfiles correspondientes a la estimación de la sección del tracto vocal.

BIBLIOGRAFÍA

[1] Fernández-Baillo R., Fernández Camacho F.J., Nieto Altuzarra., Gómez Vilda P. Biomechanical Model of Voice Production. *Libro de Actas del XXIV Internacional Congress AELFA 2004*. Madrid. pp. 590.

[2] Isshiki N, Snidecor JC. Air intake and usage in esophageal speech. Acta Otolaryngol, vol 9, pp. 559-5574. 1965.

[3] Snidecor JC, Isshiki N. Air volume and air flow relationships of sic male asophageal speakers. J Speech Hear Disord, vol 30, pp. 205-216. 1965

[4] Gatenby RA., et all. Esophageal Speech: Double-contrast evaluation of the pharyngo-esophageal segment. Radiology, 157, pp. 127-131. 1985.

[5] Kilman WJ, Goyal RK. Disorders of pharyngeal and upper esophageal sphinter motor function. Arch Intern Med. Vol. 136, pp. 592-601.1976.

[6] Sloane PM., Griffin JM.., O'Dwyer TP. Esophageal insuffation and videofluroscopy for evaluation of esophageal speech in laryngectomy patients: clinical implications. Radiology, vol 181, pp. 433-437. 1991.

[7] Gómez, P., Fernández-Baillo., et al., Evaluation of voice pathology based on the estimation of vocal fold biomechanical parameters, Journal of Voice. Vol. 21, No. 4, 2007, pp. 450-476.

[8] Gonçalves MI ,Behlau M. Laringectomia total: perspectivas dareabilitaçao vocal. In Lopes Filho, O. Tratado de fonoaudiología.S,,o Paulo, Roca, 1997.

[9] Vazquez de la Iglesia F, et all. Voz esofágica. Rev Med Univ Navarra. Vol 50, pp 56-64. 2006.

[10]Hirano M, Hibi S, Yoshida T, Hirade Y, Kasuya H, Kikuchi Y. Acoustic análisis of pathological voice: Some results of clinical application. *Acta Otolaryngologica*. Vol. 105, No 5-6, pp. 432-438, 1998.

[11]Fernández-Baillo R., Gómez P. Métodos de análisis y evaluación acústica de la voz normal y patológica. In Proccedings The lenguaje of health care. Alicante. October. 2007.

[12] Gómez-Vilda, P., Fernández-Baillo, R., Rodellar-Biarge, V., Lluis, c.N., Álvarez-Marquina, A., Mazaira-Fernández, L.M., Martínez-Olalla, R., Godino-Llorente, J.I., Glottal Source Biometrical Signature for Voice Pathology Detection, Speech Communication (2008)

[13] Fant G., Liljentcrants J., Lin Q., "A four-parameter model of glottal flow", STL-QSPR, Vol. 4, pp 1-13. 1985.

[14] Fernández-Baíllo, R., Gómez, P., Ramírez, C., Scola, B., "Pre-post surgery evaluation based on the profile of the glottal source", Proc. of MAVEBA'07, Florence, December 13-15, 2007.

[13] Titze IR. Summary Statement. *Workshopon Acoustic Voice Analysis*. National Center of Voice and Speech. 1994.

SESIÓN ESPECIAL 1
LA EVALUACIÓN ALBAYZIN 08

ADAPTACIÓN DEL CTH-URL PARA LA COMPETICIÓN ALBAYZIN 2008

Carlos Monzo, Lluís Formiga, Jordi Adell, Ignasi Iriondo, Francesc Alías y Joan Claudi Socoró

GPMM - Grup de Recerca en Processament Multimodal
Enginyeria i Arquitectura La Salle - Universitat Ramon Llull
C/ Quatre Camins 2, 08022 Barcelona, Spain

RESUMEN

En esta comunicación describimos el sistema de síntesis de voz presentado a la competición Albayzin 2008. Es un sistema que sigue un esquema clásico de concatenación de unidades basado en corpus. Cabe destacar que los costes de selección se han ajustado mediante un método basado en algoritmos genéticos y que no se ha utilizado ningún sistema de predicción prosódica. Se construyeron dos sistemas preliminares que diferían en el algoritmo de generación de forma de onda escogiendo el que se presenta a la competición mediante un test perceptual.

1. INTRODUCCIÓN

La investigación sobre síntesis de voz en la Escola Tècnica Superior d'Enginyeria Electrònica i Informàtica La Salle se inició en los ochenta con trabajos sobre síntesis articulatoria y por formantes [1, 2, 3].

Más tarde se optó por la síntesis concatenativa basada en difonemas. Se implementó un sintetizador en catalán basado en la técnica *TD-PSOLA* [4, 5], que ha sido la base para los sistemas desarrollados posteriormente hasta la actualidad. Desde entonces se ha mejorado la selección de textos a ser usados durante el proceso de grabación, la creación de reglas para la transcripción fonética (especialmente para castellano), la segmentación de las unidades y el marcado de *pitch*. Por otro lado, se han realizado investigaciones en el campo del modelado prosódico[6], ajuste de pesos para la función de selección [7, 8] y nuevas parametrizaciones basadas en cualidad de la voz [9].

La investigación del grupo los últimos años se ha basado en disminuir el coste de producción, segmentación y puesta a punto de los corpus de voz desarrollando nuevas herramientas de etiquetado automático. Dichas técnicas de rápida puesta a punto han sido desarrolladas dentro del proyecto europeo SALERO (*Semantic AudiovisuaL Entertainment Reusable Objects*). Dicho proyecto trata de conseguir un flujo producción para juegos, películas y televisión en diferentes medios de manera rápida, cualitativa y económica mediante la combinación de gráficos por ordenador, tecnologías de lenguaje y síntesis, tecnologías de web semántica así como búsqueda y recuperación basada en contenido.

El sistema presentado en esta evaluación está compuesto por tres módulos: el de transcripción fonética, selección de unidades y generación de forma de onda. La transcripción fonética se realiza mediante reglas. La selección de unidades se lleva a cabo mediante un algoritmo de programación dinámica y la generación de forma de onda mediante concatenación y modificación (en algunos casos) de la señal. La versión presentada en esta competición prescinde excepcionalmente del módulo de estimación prosódica [6] y considera la prosodia inherente en el corpus de voz de las unidades seleccionadas parcialmente siguiendo lo descrito en [10]. De esta forma el algoritmo de selección debe asegurar que la prosodia generada sea natural. Para ello debe ser capaz de escoger las unidades de forma que un elevado porcentaje sean seleccionadas consecutivamente, lo que conlleva aprovechar la variabilidad prosódica implícita en el corpus.

En las secciones 2, 4 y 5 se describen los módulos del sistema de síntesis. Además, en la sección 3 se explica el tratamiento del corpus para la generación del inventario de unidades y en la sección 6 el procedimiento seguido para selección del sistema definitivo.

Este trabajo ha sido subvencionado parcialmente por el proyecto SALERO (IST FP6-027122) de la Comisión Europea.

2. TRANSCRIPCIÓN FONÉTICA

El *phone-set* utilizado en nuestro sintetizador deriva de SAMPA [11]. El módulo de transcripción fonética consiste en un motor de reglas [12]. Las reglas actúan sobre una estructura de datos, que consiste en una lista de los pares grafema-fonema en un enunciado. Es posible utilizar una regla de inserción (I) o de borrado (D). La reglas se aplican sólo cuando la evaluación (E) de una cualidad de un fonema da un resultado positivo.

$$E(gr ==' h') \rightarrow D(gr) \quad (1)$$
$$E(gr ==' x') \rightarrow I(/ks/) \quad (2)$$

La regla (1) indica que se debe borrar el grafema $(gr)\,'h'$ y la (2) que el grafema $'x'$ se transforma en los fonemas $/ks/$ por lo que se produce una inserción. Para excepciones el sistema incluye un diccionario que se consulta antes de aplicar el motor de reglas.

3. CREACIÓN DEL INVENTARIO DE UNIDADES

3.1. Segmentación y Etiquetado

3.1.1. Segmentación por fonemas más detección de silencios

La segmentación es el proceso por el cual se etiqueta el corpus de voz indicando los límites temporales a nivel de fonema. Dentro del grupo de investigación se ha evolucionado el proceso de segmentación, incorporando mejoras respecto del primer marcador realizado [13], tanto en lo que se refiere a la calidad de marcado como en la facilidad de uso gracias a la creación de interfaces de usuario e independizándolo de la lengua de interés. Actualmente, el entrenamiento necesario y el posterior proceso de etiquetado está basados en el uso de modelos ocultos de *Markov* (HMM). Para ello se dispone de código propietario desarrollado en *Matlab®* que hace uso a su vez de la herramienta HTK (*Hidden Markov Model Toolkit*) [14].

El corpus que se ha suministrado es un corpus locutado por una voz femenina, de estilo neutro que consta de 776 archivos

de alrededor 6000 palabras y En lo que se refiere al análisis del corpus subministrado para la competición, esencialmente se ha realizado el control de la aparición y omisión de silencios respecto a los indicados en cada uno de los enunciados, independizando así el hecho que el locutor realizara las pausas de modo acorde al que los textos exigían. Por otro lado, los sonidos oclusivos se tratan de forma especial, de tal manera que el golpe de voz (*burst*) y el silencio previo se modelan como unidades diferentes.

3.1.2. Marcado de pitch

Se ha realizado en dos fases. En la primera se sitúan las marcas sobre zonas sonoras, utilizando para ello un marcador basado en el algoritmo RAPT [15]. En segundo lugar se aplica un postprocesado sobre las marcas llamado *Pitch Marks Filtering Algorithm* (PMFA) [16], con el que se consigue un marcado fiable, minimizando a su vez la existencia de inserciones y omisiones. Finalmente, se obtiene como resultado el marcado de zonas sordas y sonoras sin transiciones bruscas.

3.2. *Prunning* de la base de datos

Una vez creada la base de datos, con la información de duraciones (segmentación) y de F_0 media (*pitch*), se realiza un análisis estadístico para cada una de las unidades de forma que se descarten, durante la síntesis, aquellos casos que sean considerados erróneos (p.ej. *outliers*). El criterio considerado se basa en el hecho que todos aquellos valores, de duración y F_0, que estén fuera del margen ± 1.5 veces el valor de mediana calculado sobre todo el corpus para cada unidad serán descartados. A partir de este procedimiento, por tanto, se dispone de una lista (*black list*) donde se identifica la unidad que no se desea utilizar durante la síntesis, el archivo a la que pertenece y su posición dentro del enunciado.

En cuanto a los resultados obtenidos sobre las duraciones y F_0 medio se obtuvieron los siguiente porcentajes: 2.71% de valores descartados respecto a duración media y 1.10% respecto a la F_0 media.

4. OPTIMIZACIÓN DE COSTES DE SELECCIÓN

La selección de unidades se realiza mediante un algoritmo de programación dinámica que minimiza una función de coste para una serie de i unidades seleccionadas del corpus [17]. Al no haber predicción prosódica se omiten los costes de *target* y solo se tienen en cuenta los costes de concatenación. Según se puede observar en la ecuación (3), la matriz de costes es ponderada por un vector de pesos que intenta correlar los costes con la calidad final de la señal.

$$Csel_{\{U_1,U_2,\cdots,U_N\}} = \sum_{i=1}^{N-1}\sum_{j=1}^{3}\omega_j SC_j(i,i+1) \qquad (3)$$

En la ecuación (3) se presenta un ejemplo de cálculo de la función de coste, donde SC_{ij} corresponde al subcoste de seleccionar la unidad i según la parametrización $j = \{$PIT C, ENE C, MFCC C$\}$ y ω_j a su peso correspondiente.

4.1. Costes de concatenación y su normalización

Los costes de concatenación con los que se trabaja son:

- PIT C: Subcoste de *pitch* de concatenación. Determina la diferencia de F0 en el punto de concatenación de la unidad.
- ENE C: Subcoste de energía de concatenación. Es la diferencia de nivel energético de las unidades a concatenar en el punto de concatenación mencionado anteriormente.
- MFCC C: subcoste espectral de concatenación. Su cálculo se basa en la estimación del espectro mediante su parametrización cepstral en la escala Mel (en inglés, *Mel Frequency Cepstral Coefficients (MFCC)*. Se utilizan 24 coeficientes cepstrales más sus derivadas calculadas sobre una ventana de 20ms en el punto de concatenación.

La normalización de los costes para que sean comparables se basa en la aplicación de una función sigmoidea, debido a que la cantidad de valores atípicos no permiten una normalización MAX-MIN (la normalización sigmoidea pondera la parte lineal central de la distribución de valores). En las ecuaciones (4) y (5) se detalla la normalización siendo P_{ij}^R y $P_{(i+1)j}^L$ los subcostes de las unidades i e $i+1$ según la parametrización j y en su última (R - *right*) y primera (L - *left*) tramas, respectivamente. Adicionalmente $SC_j(u_i, u_{i+1})$ representa el coste de concatenación j de las unidades i e $i+1$ una vez normalizado. Dicha normalización se basa en el cálculo de las diferencias de los parámetros normalizándolas respecto a la desviación del subcoste (σ_{X^c}) sobre una función sigmoidea [18].

$$X^c(u_i,u_{i+1}) = \sum_{1}^{N}|P_{ij}^R - P_{(i+1)j}^L| \qquad (4)$$

$$SC_j(u_i,u_{i+1}) = 1 - e^{-\left(\frac{X^c(u_i,u_{i+1})}{\sigma_{X^c}}\right)^2} \qquad (5)$$

Asimismo, las estadísticas en el proceso de normalización se obtienen para cada una de las unidades analizadas, y se aplican evitando el problema del impacto de sesgar los subcostes debido a una normalización global de los mismos considerando todas las unidades del corpus.

Los pesos utilizados se han obtenido mediante una técnica de regresión que utiliza algoritmos genéticos por torneo (tGA) [19]. En este caso, se utilizaron los pesos normalizados que se calcularon para un corpus propio en catalán, de 20 minutos de duración y 1.207 unidades.

5. GENERACIÓN DE FORMA DE ONDA

Para la competición Albayzín 2008 se pusieron a punto dos sistemas de generación de forma de onda para la arquitectura del CTH explicada anteriormente. El primer sistema implementaba una síntesis basada en *Overlap and Add* en el dominio temporal (*TD-PSOLA*) [20] mientras que el segundo simplemente realizaba concatenación directa de la señal según sus marcas de *pitch* (*RAW*).

La modificación de la señal mediante *TD-PSOLA* se ha aplicado solamente a las unidades sonoras ralentizando su velocidad un 10% (aumentando su duración). Para alargar unidades se hace una interpolación de las tramas.

La concatenación *RAW* no hace ningún tipo de ventaneo y se concatena la señal en el paso por cero más cercano a la marca de *pitch*. La ventaja de este tipo de concatenación se basa en mantener la propia naturalidad de la señal. Dicho sistema, asume que ante una selección de unidades óptima, no es necesaria ninguna normalización o modificación de la señal y así se puede aprovechar la velocidad, volumen y entonación de la grabación original [21].

6. SELECCIÓN DEL SISTEMA PRESENTADO

Debido a la restricción de presentar un sólo sistema por equipo participante, se hizo una selección del sistema de síntesis más aceptado a nivel perceptual. A este efecto se escogieron al azar cinco frases de las 350 de test que se tenían que generar, se sintetizaron con *TD-PSOLA* y con *RAW* para que el usuario final escogiera entre ellas (según la naturalidad y inteligibilidad conseguidas).

15 usuarios, tanto expertos como no expertos en tecnologías del habla, realizaron la prueba cuyos resultados se pueden observar en la figura 1.

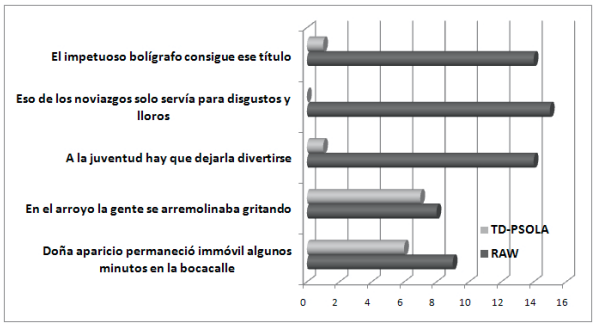


Figura 1. Porcentaje de preferencia del test perceptual para las cinco frases escogidas del test.

Analizando los resultados se puede observar que el número de votaciones entre *RAW* y *TD-PSOLA* es muy parecido en dos de las cinco frases donde siempre acaba ganando por la mínima *RAW*. En las otras frases *RAW* peresenta una abrumadora mayoría. Por este motivo, se decidió presentar a la evaluación el sistema *RAW*.

7. CONCLUSIONES

Desde el punto de vista de la creación de la voz se puede concluir que el tiempo de puesta a punto ha sido bastante rápido y eficiente. Para cuantificar el tiempo destinado para ello, serían en torno a 20 horas de parte de un técnico y las necesarias, en función de la longitud del corpus y de la potencia de cálculo de la máquina, para su parametrización. Cabe decir que el tiempo y coste destinado ha sido el más óptimo según nuestra experiencia, ya que para cada nueva voz desarrollada las herramientas para su etiquetado han sido mejoradas y los procesos más automatizados según el proyecto SALERO detallado en la introducción.

Cabe deducir, debido al resultado de las pruebas perceptuales, que al no disponer de modelo prosódico, la técnica de *TD-PSOLA* no mejora, y en algunos casos empeora, la calidad perceptual de la señal.

Es la segunda voz en castellano utilizada para un dominio genérico con nuestro conversor texto-habla. Nuestro primer conversor texto-habla utilizó una voz expresiva con distintas emociones el cual ha dado lugar a diferentes publicaciones y proyectos [22, 6, 23].

8. TRABAJO FUTURO

La estrategia de futuro de nuestro CTH pasa por incorporar técnicas de modificación de la señal basadas en modelos harmónicos más ruido, mejorar el procesado de la señal según *TD-PSOLA* teniendo en cuenta la especifidad de cada unidad, incorporar costes lingüísticos en coste de *target*, incorporar el trabajo prototipo de selección de unidades basado en medidas perceptuales detallado en [7, 8] y incorporar el modelo prosódico expresivo detallado en [6].

9. BIBLIOGRAFÍA

[1] Josep Martí, *Estudi acústic del català i síntesi automàtica per ordinador*, Ph.D. thesis, Universidad de Valencia, Valencia, España, 1985.

[2] Josep Martí, *Reconocimiento automático del habla*, chapter Síntesis del habla: Evolución histórica y situación actual, Boixareu Marcombo, 1987.

[3] Josep Martí, "Estado actual de la síntesis de voz," in *Estudios de Fonética Experimental*, 1990, vol. 4, pp. 147–168.

[4] Joan Camps, Gerard Bailly, y Josep Martí, "Synthèse à partir du texte pour le catalan," in *Proc. 19èmes Journées d'Études sur la Parole*, Bruselas, Francia, 1992, pp. 329–333.

[5] Roger Guaus, Francesc Gudayol, y Josep Martí, "Conversión textovoz mediante síntesis PSOLA," in *Jornadas Nacionales de Acústica*, Barcelona, España, 1996, pp. 355–358.

[6] Ignasi Iriondo, Joan Claudi Socoró, y Francesc Alías, "Prosody Modelling of Spanish for Expressive Speech Synthesis," in *Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Honolulu, EUA, Abril 2007, vol. 4, pp. 821–824.

[7] Lluís Formiga y Francesc Alías, "Extracting User Preferences by GTM for aiGA Weight Tuning in Unit Selection Text-to-Speech Synthesis," in *Computational and Ambiental Intelligence - Proceedings on 9th International Work-Conference on Artificial Neural Networks (IWANN)*, 2007.

[8] Francesc Alías, Xavier Llorà, Lluís Formiga, Kumara Sastry, y David E. Goldberg, "Efficient interactive weight tuning for TTS synthesis: reducing user fatigue by improving user consistency," in *Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toulouse, Francia, 2006, vol. I, pp. 865–868.

[9] Carlos Monzo, Francesc Alías, Ignasi Iriondo, Xavier Gonzalvo, y Santiago Planet, "Discriminating Expressive Speech Styles by Voice Quality Parameterization," in *Proc. of the 16th International Congress of Phonetic Sciences(ICPhS)*, Saarbrücken, Alemania, Abril 2007, pp. 2081–2084.

[10] Francesc Alías, Ignasi Iriondo, Lluís Formiga, Xavier Gonzalvo, Carlos Monzo, y Xavier Sevillano, "High quality Spanish restricted-domain TTS oriented to a weather forecast application," in *Proc. of the 9th International Conference on Speech Communication and Technology (Eurospeech)*, Lisboa, Portugal, 2005, pp. 2573–2576.

[11] John C. Wells, *SAMPA computer readable phonetic alphabetHandbook of Standards and Resources for Spoken Language Systems*, chapter SAMPA computer readable phonetic alphabet, pp. Part IV, section B, Berlin and New York: Mouton de Gruyter, 1997.

[12] Ignasi Iriondo, *Producción de un corpus oral y modelado prosódico para la síntesis del habla expresiva*, Ph.D. thesis, Universitat Ramón Llull, 2008.

[13] Francesc Alías y Ignasi Iriondo, "Segmentador de fonemas en catalán basado en DHMM," in *Actas del 16th Simposium Nacional de la Unión Científica (URSI)*, Madrid, España, 2001, pp. 149–150.

[14] "HTK," in *Recuperado el 19 de 09 de 2008, de http://htk.eng.cam.ac.uk*, 2008, pp. 149–150.

[15] David Talkin, "A Robust Algorithm for Pitch Tracking (RAPT)," in *W. B. Kleijn y K. K. Paliwal (eds.). Speech Coding and Synthesis. Amsterdam, NL: Elsevier Science*, 1995, pp. 495–518.

[16] Francesc Alías, Carlos Monzo, y Joan C. Socoró, "A Pitch Marks Filtering Algorithm based on Restricted Dynamic Programming," in *Proc. of International Conference on Speech and Language Processing (ICSLP)*.

[17] Andrew Hunt y Alan W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Atlanta, EUA, 1996, vol. 1, pp. 373–376.

[18] Albert Febrer, *Síntesi de la parla per concatenació basada en la selecció*, Ph.D. thesis, Universitat Politècnica de Catalunya, Gener 2001.

[19] Francesc Alías y Xavier Llorà, "Evolutionary weight tuning based on diphone pairs for unit selection speech synthesis," in *Proc. of the 8th European Conference on Speech Communication and Technology (EuroSpeech)*, Geneve, Suiza, 2003, pp. 1333–1336.

[20] Eric Moulines y Francis Charpentier, "Pitch-Sinchronous waveform processing techniques for text-to-speech synthesis using diphones," in *Speech Communication*, 1990, vol. 9, pp. 453–467.

[21] Marcello Balestri, Alberto Pacchiotti, Silvia Quazza, Pier Luigi Salza, y Stefano Sandri, "Choose the best to modify the least: a new generation concatenative synthesis system," in *Proc. of the 6th European Conference on Speech Communication and Technology (EuroSpeech)*, 1999, pp. 2291–2294.

[22] Luigi Ceccaroni, Paloma Martínez, Josefa Z. Hernández, y Xavier Verdaguer, "IntegraTV-4all: an interactive television for all," in *Proc. of 1st International Symposium on Ubiquitous Computing and Ambient Intelligence (UCAmI'05)*, 2005.

[23] Francesc Alías, Xavier Sevillano, Joan Claudi Socoró, y Xavier Gonzalvo, "Towards high quality next-generation Text-to-Speech synthesis: a Multidomain approach by automatic domain classification," *IEEE Transactions on Audio, Speech and Language Processing (Special issue on New Approaches to Statistical Speech and Text Processing)*, vol. 16 (7), pp. 1340–1354, 2008.

ATVS-UAM ALBAYZIN-VL08 SYSTEM DESCRIPTION

Doroteo T. Toledano, Ismael Mateos-Garcia, Alejandro Abejon-Gonzalez, Daniel Ramos, Juan Bonillo and Joaquin Gonzalez-Rodriguez

ATVS Biometric Recognition Group, Universidad Autonoma de Madrid, Spain

{doroteo.torre, ismael.mateos, alejandro.abejon, daniel.ramos, juan.bonillo joaquin.gonzalez} @uam.es

Abstract

ATVS submission to ALBAYZIN-VL08 will consist of different combinations of a set of acoustic and phonotactic subsystems that our group has developed during the last years. Most of these subsystems have already been evaluated on NIST LRE 07 evaluation. At the time of writing this system description some of the details of our submission are still undefined. Therefore we will briefly describe our systems and the intended combinations to be submitted, but these settings should not be taken as final in any way. As acoustic subsystems we will use a GMM SuperVectors and a GLDS-SVM subsystem, while the phonotactic subsystem will be a PhoneSVM system. We are still deciding the best fusion strategy and the best combination of subsystems at the time of writing. Output scores will be submitted in the form of log-likelihood ratio (logLR) scores in an application independent way. Open-set detection thresholds will be set to the Bayes thresholds in all cases, and the same logLR sets will probably be submitted to the closed- and open-set conditions.

1. Introduction

ATVS-UAM submission to ALBAYZIN-VL08 consists of different combinations of a set of acoustic and phonotactic subsystems that our group has developed during the last years. The two acoustic subsystems are based on two different techniques: SVM-GLDS (language recognition using SVMs with Generalized Linear Discriminant Sequence kernel) and GMM SuperVectors (also named as GMM-SVM in this document, is language recognition using SVMs that take as input the means of Gaussian Mixture Models). The phonotactic system will be a Phone-SVM subsystem (phone recognition and n-gram modeling followed by Support Vector Machine classification).

A particularity of all of ATVS subsystems is that no transcribed speech is needed to train language models. This makes them particularly useful for situations where few language resources are available or when transcription of materials for training the language models is difficult or very expensive. For this reason, our subsystems are better fitted for the restricted training condition of the evaluation. The Phone-SVM subsystem, however, requires phonetic recognizers trained on different corpora, and therefore cannot be included in our submission to the restricted training condition.

The same individual subsystems will be used to perform language recognition for test segments of 3, 10 and 30s. These subsystems will be fused together in some way. At this time we are experimenting with several fusion strategies ranging from sum fusion to anchor model fusion. The scores will be submitted as calibrated Log-Likelihood Ratios.

The rest of this system description is organized as follows: Section 2 describes the acoustic subsystems, Section 3 the phonotactic subsystems, Section 4 the fusion strategies and Section 5 the calibration process.

2. Acoustics systems

We use two different acoustic systems, both of which are based on SVMs using different features. In this section we describe these two acoustic systems and the training material used for training them.

2.1. Individual Sub-systems: SVM-GLDS

The first individual sub-system is, in fact, the fusion of two acoustic systems based on SVM [1,2] using different features. Systems use a kernel expansion on the whole observation sequence, and a separating hyperplane is computed between the target language features and the background model. Both ATVS acoustic SVM-GLDS subsystems use a polynomial expansion of degree three [3] followed by a Generalized Linear Discriminant Sequence kernel as described in [4].

2.2. Individual Sub-systems: GMM-SVM (or GMM SuperVectors)

This subsystem is based on using an SVM classifier over the GMM models space. The language model is constructed by MAP (Maximum A-Posteriori) adaptation of the means of the UBM (Universal Background). A GMM super vector is constructed by stacking the means of the adapted mixture components. Then the SVM classifier is used to train and separating hyperplane in the vector space defined by the super-vectors.

2.3. Training data used

To fulfil the restricted condition training of the ALBAYZIN-VL08 evaluation these subsystems have been trained and adjusted using exclusively the training (and development) data supplied by the organization of the evaluation. In this way, these acoustic systems can (and will) be used for the restricted training condition. We have downsampled the speech materials provided by the organization to 8 kHz because our systems were developed to work on telephone speech. This will limit the performance of our systems.

3. Phonotactic Systems

Our phonotactic system consists of a Phone-SVM system using 7 phonetic recognizers in 7 different languages. In this section we describe this system in more detail and discuss the training material used.

3.1. Individual Sub-systems: Phone-SVM

Each of the seven different Phone-SVM subsystems is based on the following steps. First a voice activity detector segments

the test utterance into speech and non-speech segments. The speech segments are recognized with one open-loop phonetic decoder. The best decoding is used to estimate count-based 1-grams, 2-grams and 3-grams. All these parameters are reshaped as a single vector that is taken as the input of an SVM that classifies the test segment as corresponding (or not) to one language.

The process described above is repeated for the seven different open-loop phonetic recognizers used. All decoders are based on Hidden Markov Models (HMMs) trained using HTK and used for decoding with SPHINX. The phonetic HMMs are three-state left-to-right models with no skips, being the output pdf of each state modelled as a weighted mixture of Gaussians.

For each test utterance, the systems make n-grams with the transcription produced by the phonetic decoders. Support Vector Machines (SVMs) take the n-grams as input vectors [1,2].

3.2. Training data used

Most of the seven decoders were trained on SpeechDat-like corpora, containing over 10 hours of training material per language and covering hundreds of different speakers. The languages of these phonetic decoders are English, German, French, Arabic, Basque and Russian. We have also included a 7th phonetic decoder in Spanish trained on Albayzin [5] downsampled to 8 kHz, which contains about 4 hours of speech for training. We developed these phonetic recognizers for telephone speech (mainly for NIST LRE evaluations) and will use them with the test materials downsampled to 8 kHz. This will limit the performance of our systems.

Although the models that we will use for detecting the language for the ALBAYZIN-VL08 evaluation will be trained entirely on the training and development materials provided by the organization, the use of previously trained phonetic decoders in this system makes it usable only in the unrestricted training condition.

4. Fusion

In order to combine the results of the subsystems presented above different fusion techniques are currently being explored, from classical techniques, like sum fusion, to novel ones like anchor-models fusion [6-8]. Prior to any processing, the scores of each of the individual sub-systems are normalized using a test-segment dependent normalization. This normalization is also currently under study, so we have not yet decided the final configuration.

5. Calibration and decision

In order to take the actual decision we will follow a per-language detection approach in order to calibrate the output log-likelihood-ratios (logLR). Therefore, each score *for each of the target languages* will be mapped to a logLR assuming a target-language-vs.-all configuration, in the following way:

$$\log(LR) = \log\left(\frac{P(\text{score}|\text{target language})}{P(\text{score}|\text{any other non-target language})}\right)$$

After calibrating logLR values, the logarithm of the Bayes threshold will be used in order to take decisions. If the calibration process is correctly performed, this is equivalent to choosing the minimum-cost threshold for each target language detection sub-system.

6. Conclusions

For the ALBAYZIN-VL08 evaluation, we have built mainly on previously developed subsystems that we have used in NIST LRE 07, trying to adapt them for the particular task and languages proposed in the ALBAYZIN-VL08 evaluation. Our systems have been developed with the requirement of easy training for new languages, so it has been relatively straightforward to train them for the languages of the ALBAYZIN-VL08 evaluation. However, once the subsystems have been trained we still have to fine tune the fusion and the calibration. This work is still in progress at the time of writing.

Acknowledgments

We thank the organizers of the ALBAYZIN-VL08 evaluation for their hard work in preparing this evaluation and the corresponding training, development and test materials. This work was funded by the Spanish Ministry of Science and Technology under project TEC2006-13170-C02-01.

7. References

[1] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, “Support vector machines for speaker and language recognition,” Computer Speech and Language, vol. 20, no. 2-3, pp. 210–229, 2006.

[2] W. Wan and W. Campbell, “Support vector machines for speaker verification and identification,” in Proc. of IEEE International Workshop on Neural Networks for Signal Processing, 2000, pp. 775–784.

[3] Pedro A. Torres-Carrasquillo, Elliot Singer, Mary A. Kohler, Richard J. Greene, Douglas A. Reynolds and J.R. Deller Jr, “Approaches to Language Identification using Gaussian Mixture. Models and Shifted Delta Cepstrum”, ICSLP, 2002.

[4] W. Campbell, “Generalized linear discriminant sequence kernels for speaker recognition,” in Proc. of ICASSP, 2002, pp. 161–164.

[5] A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J. Mariño, C. Nadeu, “ALBAYZÍN Speech Database: Design of the Phonetic Corpus,” in *proceedings of the 3rd European Conference on Speech Communication and Technology (EUROSPEECH)*. Berlin, Germany, 21-23 September 1993. Vol. 1. pp. 175-178.

[6] N. Brümmer et al. “Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006.” IEEE Transactions on Audio, Speech and Signal Processing, 15(7) pp. 2072-2084, 2007.

[7] Mikael Collet, Yassine Mami, Delphine Charlet, Frederic Bimbot, "Probabilistic Anchor Models Approach for Speaker Verification", in INTERSPEECH 2005.

[8] Elad Noor1, Hagai Aronowitz "Efficient Language Identification using Anchor Models and Support Vector Machines", in Odyssey 2006 ISBN: 1-4244-0472-X pp 1-6.

DESCRIPCIÓN DE LOS SISTEMAS PRESENTADOS POR IXA-EHU A LA EVALUACIÓN ALBAYCIN'08

Gorka Labaka, Arantza Díaz de Ilarraza, Kepa Sarasola

Grupo IXA
Universidad del País Vasco
{jiblaing, jipdisaa, jipsagak}@ehu.es

RESUMEN

En este artículo describimos los sistemas presentados por el grupo IXA-EHU a la evaluación ALBAYCIN'08. Dada las características de los pares de lenguas a tratar y la naturaleza aglutinativa del euskara hemos procedido a la segmentación de las palabras en morfemas para, de este modo, facilitar el alineamiento. Además este proceso habilita la posibilidad de aprender pseudosintagmas (secuencias de palabras sin estructura sintáctica) que pueden estar compuestos además de por palabras, por morfemas considerados de manera independiente a la palabra a la que van unidos; por ejemplo, en el caso de 'etxe-ra' noa (voy a casa), el pseudosintagma '-ra noa' se puede alinear con 'voy a'.

Además de la segmentación hemos incorporado a la tabla de traducción pares de pseudosintagmas que han sido extraídos utilizando técnicas de traducción basada en ejemplos de MaTrEx [1]. Estos nuevos pseudosintagmas, a diferencia de los extraídos por las técnicas estadísticas, coinciden con sintagmas desde un punto de vista lingüístico. Al ampliar la tabla de traducción con estos nuevos pseudosintagmas, se amplia la cantidad de pseudosintagmas disponibles por el decodificador, además de favorecer aquellas traducciones sintácticamente correctas.

1. INTRODUCCIÓN

En este artículo describimos los sistemas presentados por el grupo IXA de la Universidad del País Vasco a la evaluación Albaycin'08.

El alto nivel de flexión del euskara, junto con el hecho de que no haya gran cantidad de corpus paralelo accesible, complica la tarea de traducción español-euskara convertiendola en un reto interesante.

Para hacer frente a su alto nivel de flexión hemos segmentado las palabras en euskara dividiéndolas en morfemas, de modo similar al realizado en otros trabajos para otros pares de lenguas de gran flexión, como en el caso del inglés-checo [2] y el inglés-turco [3].

El articulo está organizado del siguiente modo: en la sección 2 explicamos las distintas técnicas que usaremos en nuestros sistemas; la sección 3 está dedicada a mostrar los sistemas que hemos evaluado y que se basan en combinaciones de las técnicas previamente explicadas; en la sección 4 resumimos los resultados conseguidos por cada uno de los sistemas; finalmente comentamos las conclusiones extraídas de esos resultados (sección 5).

Este trabajo ha sido subvencionado por Gobierno Vasco, mediante la ayuda predoctoral concedida a Gorka Labaka (código BFI05.326)

2. TÉCNICAS UTILIZADAS

En esta sección explicamos las técnicas que hemos utilizado en la implementación de los sistemas que presentamos a la evaluación Albaycin'08.

2.1. Segmentación del texto en euskara

Dada la naturaleza aglutinante de la lengua y, continuando con el trabajo presentado en un publicación anterior [4], hemos llevado a cabo la segmentación del texto en euskera. De esta manera, tendremos en tokens independientes los morfemas que conforman una palabra.

Para llevar a cabo esta segmentación hemos analizado el texto en euskara utilizando EusTagger[5] y cada palabra se ha separado en como máximo tres tokenes: los prefijos, el lema y los sufijos. De este modo, para una palabra como 'etxekoa' (el de la casa) se crean dos tokenes: 'etxe' y '+koa'. En una primera aproximación, pensamos en crear un token por cada morfema pero, dadas las características de la salida de EusTagger que genera una segmentación muy fina (con muchos morfemas por palabra), decidimos unir todos los sufijos en un único token; los prefijos también fueron tratados de la misma manera. La razón principal para llevar a cabo esta segmentación es facilitar el alineamiento, ya que de este modo habrá menos alineamientos múltiples a la vez que se reduce la dispersión.

Gracias a este proceso de segmentación podemos aprender pseudosintagmas en los que toman parte sólo algunos de los morfemas de una palabra. De este modo se podría extraer el par de pseudosintagmas 'voy a' '-ra noa', donde la preposición 'a' se traduce con el sufijo '-ra' cuando acompaña al verbo 'ir' independientemente del lema al que esté unido. Sin la segmentación no sería posible esta clase de generalización teniendo que extraer pseudosintagmas distintos para cada ejemplo.

El hecho de usar el texto segmentado para entrenar el traductor estadístico, conlleva que necesitemos generar el texto final en euskara basándonos en la salida del traductor, ya que esta estará segmentada al igual que el corpus utilizado en el entrenamiento. Para generar el texto final hemos utilizado el módulo de generación del traductor basado en reglas matxin[6] (que utiliza en el mismo léxico que el analizador).

A la hora de generar el texto final hay que tener en cuenta que el traductor estadístico puede producir combinaciones de morfemas que no correctas pudiéndole asignar a un nombre la flexión correspondiente a un verbo o incluso llegando a asignarle algún tipo de flexión a tokenes que no se pueden flexionar como los signos de puntuación. En este caso y, como primera aproximación, eliminamos la flexión dejando únicamente el lema.

Finalmente, para poder incorporar un modelo de lenguaje basado en palabras (el decodificador usará uno basado en el texto segmentado), en vez de obtener sólo la mejor traducción que el decodificador es capaz de encontrar, obtenemos una lista de las n traducciones más probables y, tras la generación, reordenamos la lista de traducciones incorporando el modelo de lenguaje basado en palabras como si fuera un modelo más.

2.2. Hibridación SMT-EBMT: sistema MaTrEx

En colaboración con Nacional Centre for Language Technology de la Dublin City University hemos adaptado su sistema MaTrEx[1] para utilizarlo con el euskara. Este sistema consiste en enriquecer la tabla de traducción con pares de pseudosintagmas extraídos usando técnicas de la traducción automática basada en ejemplos.

Para extraer los nuevos pseudosintagmas, se analizan sintácticamente ambas partes del corpus paralelo y se marcan los sintagmas (hemos usado Freeling [7] para procesar el español y Eustagger para el euskara). En un segundo paso y basándose en los alineamientos palabra por palabra se alinean estos sintagmas y se incorporan a tabla de traducción.

3. SISTEMAS PRESENTADOS

Para crear nuestros sistemas hemos utilizado las siguientes herramientas:

- Alineador de palabras GIZA++ [8].
- Modelo de lenguaje SRILM [9]
- Moses SMT Toolkit [10]

Mediante estas herramientas de uso libre y los corpora habilitados por la organización (en la tabla 1 se muestra algunos datos de los corpora) hemos creado un sistema *baseline*, usando los *scripts* y los parámetros que Moses trae por defecto. Hay que tener en cuenta que el sistema *baseline* de Moses incorpora técnicas de reordenación lexicalizada además de la basada en distancia. En la creación del *baseline* se han llevado a cabo la optimización de los pesos de cada modelo usando BLEU y Minimum Error Rate Training.

Basándonos en este *baseline* hemos incorporado las técnicas explicadas en la sección 2 creando distintos sistemas de traducción. Posteriormente hemos evaluado el impacto que tiene cada técnica. Para incorporar los pseudosintagmas correspondientes al sistema MaTrEx, hay que analizar ambos textos, alinear los sintagmas basándose en los alineamientos palabra por palabra e incorporar los nuevos pares de pseudosintagmas a la tabla de traducción antes de calcular los pesos de los modelos de traducción con los *scripts* proporcionados con Moses. Tras este proceso se continua con el entrenamiento del sistema.

Por otro lado a la hora de usar el texto segmentado, además de preprocesar y post-procesar las oraciones en euskara, para segmentar el texto y volver a generar la forma final, hemos tenido que modificar el proceso de optimización para poder optimizar también el peso del modelo de lenguaje basado en palabras. Como hemos explicado anteriormente, el decodificador utiliza un modelo de lenguaje basado en el texto segmentado, y el modelo de lenguaje basado en palabras se incorpora a la traducción después del post-proceso de generación mediante el reordenamiento de una lista n-best. Por lo que en cada paso de la optimización hay que incorporar tanto la generación como el reordenamiento de las listas basándose en la lista n-best.

Además de los sistemas donde probamos las técnicas presentadas individualmente, también hemos entrenado un sistema donde probamos la combinación de ambas.

4. RESULTADOS

Hemos evaluado los sistemas presentados en la sección 3 sobre el corpus de test usando las métricas automáticas más usuales (BLEU, MBLEU, WER, PER). En la tabla 2 se presentan los resultados para dichos sistemas y métricas.

Lo más destacable de los datos presentados es que ambas técnicas individuales (MaTrEx y segmentación del euskara) mejoran los resultados del sistema *baseline* para todas las métricas utilizadas. A su vez, la combinación de técnicas supera a cada una de ellas considerada individualmente, logrando los mejores resultados.

5. CONCLUSIÓN

Las técnicas que hemos utilizado han dado un resultado satisfactorio mejorando ambas el *baseline*. Además la combinación de las misma supera a las técnicas aplicadas individualmente.

Respecto al trabajo futuro, nos proponemos modificar la segmentación del euskara, buscando una forma alternativa para agrupar los morfemas. Actualmente, todos los morfemas que acompañan al lema se agrupan en un único

corpora	lenguaje	oraciones	tokenes	vocabulario-tokenes
entrenamiento	Español	58202	1284212	50927
	Euskara		1010545	95724
	Euskara-segmentado		1546304	40436
development	Español	1456	32743	7073
	Euskara		25778	9030
	Euskara-segmentado		39420	6191
test	Español	1446	31004	6836
	Euskara		24372	8695
	Euskara-segmentado		37347	5976

Tabla 1. Estadísticas de los corpora utilizados.

	BLEU	MBLEU	NIST	WER	PER
baseline	10.82	10.21	4.51	80.44	61.67
MaTrEx	11.03	10.38	4.54	80.13	61.65
segmentación euskara	11.19	10.49	4.65	79.27	60.60
segmentación + MaTrEx	**11.37**	**10.65**	**4.71**	**78.65**	**60.01**

Tabla 2. Evaluación de los distintos sistemas probados.

token ya que se consiguen mejores resultados que manteniendo cada morfema un token pero pensamos que una agrupación intermedia mejoraría los resultados.

Por otro lado, nos planteamos mejorar el proceso de generación de la forma final, modificando la secuencia de morfemas que devuelve el traductor estadístico en aquellos casos que ésta no sea morfológicamente correcta. Esta modificación puede implicar la reordenación de la secuencia o la eliminación de algunos de los morfemas.

6. BIBLIOGRAFÍA

[1] N. Stroppa y A. Way, "MaTrEx: DCU Machine Translation System for IWSLT 2006," in *Proceedings of the International Workshop on Spoken Language Translation*, Kyoto, Japan, 2006, pp. 31–36.

[2] S. Goldwater y D. McClosky, "Improving statistical mt through morphological analysis," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Vancouver, 2005.

[3] Kemal Oflazer y Ilknur Durgar El-Kahlout, "Exploring different representational units in english-to-turkish statistical machine translation," in *Proceedings of Statistical Machine Translation Workshop at ACL 2007*, Prague, Czech Republic, June 2007.

[4] E. Agirre, A. Díaz de Ilarraza, G. Labaka, y K. Sarasola, "Uso de información morfológica en el alineamiento español-euskara," in *XXII Congreso de la SEPLN*, Zaragoza, septiembre 2006.

[5] I. Aduriz y A. Díaz de Ilarraza, "Morphosyntactic disambiguation and shallow parsing in computational porcessing of basque," in *Inquiries into the lexicon-syntax relations in Basque*, Bernarrd Oyharçabal, Ed., Bilbao, 2003.

[6] I. Alegria, A. Díaz de Ilarraza, G. Labaka, M. Lersundi, A. Mayor, K. Sarasola, M. Forcada, S. Ortiz, y L. Padró, "An open architecture for transfer-based machine translation between spanish and basque," in *Workshop on Open-Source Machine Translation*, Asia-Pacific Association for Machine Translation (AAMT), Ed., Phuket, Thailand, September 2005, pp. 7–14.

[7] X. Carreras, I. Chao, L. Padró, y M. Padró, "Freeling: an open-source suite of language analyzers," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, 2004.

[8] F. Och y H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[9] Andreas Stolcke, "SRILM - An Extensible Language Modeling Toolkit," in *Proc. Intl. Conf. Spoken Language Processing*, Denver, Colorado, September 2002.

[10] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, y Evan Herbst, "Moses: Open Source Toolkit for Statistical Machine Translation," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, June 2007.

DESCRIPCIÓN DEL CONVERSOR DE TEXTO A VOZ AHOTTS PRESENTADO A LA EVALUACIÓN ALBAYZIN TTS 2008

Iñaki Sainz, Inma Hernáez, Eva Navas, Jon Sanchez, Iker Luengo, Ibon Saratxaga, Igor Odriozola, Eneritz de Bilbao, Daniel Erro

Aholab Signal Processing Laboratory.
Departamento de Electrónica y Comunicaciones.
Universidad del País Vasco

RESUMEN

En el presente artículo se describen las características básicas del conversor de texto a voz (CTV) AhoTTS desarrollado por el grupo Aholab de la Universidad del País Vasco. AhoTTS es un sistema en el que tanto el módulo prosódico como el acústico están basados en técnicas por corpus. Asimismo se detalla el proceso de generación de una voz en castellano dentro de la campaña de evaluación Albayzin TTS 2008.

1. INTRODUCCIÓN

La campaña de evaluación Albayzin TTS 2008 tiene como propósito principal comparar las distintas técnicas e implementaciones de los sistemas participantes, partiendo de un base de datos común. Para ello, sigue la línea trazada por la evaluación internacional "Blizzard Challenge". La última edición de dicha evaluación se llevó a cabo tanto para el Inglés como para el Chino Mandarín, mientras que Albayzin TTS 2008 en esta su primera edición, se ha centrado únicamente en el Castellano.

Cada participante ha dispuesto de un periodo de 7 semanas para generar una voz a partir de la base de datos proporcionada por la UPC. Tras dicho periodo, se han sintetizado múltiples textos de test que serán evaluados de forma subjetiva bajo los siguientes criterios de calidad: Parecido con la voz original, Naturalidad e Inteligibilidad.

En este artículo se explican las características principales del CTV AhoTTS. En la sección 2 se describen los módulos que componen el sistema de síntesis. El proceso de generación de la voz se explica en la sección 3. Finalmente, se presentan unas conclusiones sobre todo el proceso en la sección 4.

2. DESCRIPCIÓN DEL SISTEMA

AhoTTS es el CTV que el grupo Aholab lleva desarrollando desde 1997. Implementado en C/C++ dispone de un arquitectura modular y multiplataforma. En la actualidad se han desarrollado voces en los siguientes idiomas: Euskera, Castellano e Inglés (para este último haciendo uso de los módulos de procesado lingüístico de Festival [1]). En las siguientes subsecciones se explicarán las características principales de cada uno de los módulos básicos que componen el sistema general: Procesado Lingüístico, Predicción Prosódica y Módulo Acústico.

2.1. Procesado Lingüístico

La función principal de este módulo es la de proporcionar una secuencia de fonemas a partir de un texto de entrada. Este proceso implica varias fases: normalización, delimitación de las frases, categorización, silabificado, acentuación y transcripción fonética. Aunque AhoTTS ya disponía de dichas funciones para el castellano, se han realizado mejoras o modificaciones en las tres últimas etapas.

2.2. Predicción de la Prosodia

Partiendo de la información del módulo precedente se pretende modelar la prosodia (entonación, duración y potencia) buscando imitar lo mejor posible la del locutor original. Por ello se han desarrollado modelos basados en el corpus proporcionado por la organización de Albayzin TTS 2008. Cabe destacar que no se ha desarrollado ningún modelo de inserción de pausas para la voz en castellano, por lo que únicamente se utiliza la puntuación ortográfica del texto de entrada.

2.2.1. Duración

La duración de cada fonema se predice mediante árboles CART para vocales, semivocales, consonantes sonoras y consonantes sordas. El entrenamiento se lleva a cabo en base a la siguientes características con una ventana que recoge los 2 fonemas anteriores y posteriores al actual: Fonema, vocal/consonante, vocales (altura, amplitud y redondez), consonantes

(sonoridad, clase, punto articulación…), acento, posición (sílaba, palabra, frase), etc.

2.2.2. Entonación

Se trata sin duda del rasgo prosódico más relevante en la calidad y naturalidad de la síntesis obtenida. AhoTTS dispone de tres modelos entonativos:

- *Modelo 1*: Una implementación muy simple de picos y valles.
- *Modelo 2*: Un modelo estadístico basado en árboles y curvas de Fujisaki que proporciona una entonación con una alta consistencia y naturalidad. Sólo está implementado para Euskera.
- *Modelo 3*: Modelo entonativo basado en corpus.

El sistema presentado en la evaluación Albayzin TTS 2008 hace uso del modelo 3, por lo que se procederá a detallar las características del mismo en las siguientes líneas.

Como todo modelo entonativo basado en la selección de unidades, para formar la curva final resultante extrae y concatena curvas de pitch naturales. A diferencia de la mayoría de sistemas, que utilizan como unidad básica el grupo acentual, se ha optado por una implementación similar a la de [2] en la que la unidad básica es el fonema.

- *Coste Objetivo*: A partir de la transcripción fonética de entrada y para cada fonema sonoro, se realiza una preselección de candidatos en base a las siguientes características: Fonema (coste nulo si el fonema es idéntico y en caso contrario ponderado por clases fonéticas), Sonoridad, Duración, Tipo de Grupo Acentual, Posición en el Grupo Acentual, Posición en la Sílaba, Tipo de Fonema Adyacente (vocal, semivocal, consonante sonora/sorda y pausa), Tipo de Grupo Fónico, Distancia al acento más cercano, Posición del Grupo Acentual dentro del Grupo Entonativo y Número de Grupos Acentuales dentro del Grupo Entonativo. Los pesos de los costes objetivo se entrenan siguiendo un esquema similar al propuesto en [3] usando regresión lineal múltiple. Para ello se define como distancia a predecir, la del contorno de pitch del fonema sonoro anterior, actual y siguiente, dando mayor peso a la forma del contorno que al valor absoluto del pitch.
- *Coste Concatenación*: Tras la fase de preselección, se computan los costes de concatenación para obtener la curva definitiva. Dichos costes incluyen: Distancia entre el 'siguiente contorno natural del fonema anterior' y el del actual y viceversa (distancia entre el 'contorno del fonema anterior' y el 'contorno natural anterior del fonema actual'), Diferencia de pitch entre los extremos cuando se trata de dos fonemas sonoros adyacentes, y Penalización por máximo salto de pitch entre sílabas adyacentes (calculado a partir de la media y desviación estandar de los saltos en la voz natural).

Finalmente, se interpola el pitch en los fonemas sordos, se suaviza entre fonemas sonoros adyacentes si fuera necesario, y se modifica ligeramente la duración de cada fonema sonoro interpolando la predicha por el modelo de duración correspondiente, con la duración del contorno entonativo seleccionado.

2.2.3. Energía

Dado que la potencia no se utiliza como coste objetivo durante la selección de unidades acústicas, se ha optado por no generar un modelo de energía para la voz en castellano.

2.3. Módulo Acústico

El módulo acústico combina las fases típicas de un sistema concatenativo basado en corpus: Preselección de Unidades, Programación Dinámica combinando los costes objetivo y de concatenación y la Síntesis de la secuencia de unidades seleccionadas para generar la onda de audio final.

2.3.1. Selección de Unidades

La unidad básica empleada por nuestro sistema es el semifonema, pero si existen suficientes candidatos (umbral situado en unos pocos centenares) hacemos uso de difonemas. De esta forma se establece un compromiso entre la flexibilidad prosódica que permite el hacer uso de semifonemas y la preservación de la naturalidad motivada por el uso de difonemas.

Haciendo uso del algoritmo de Viterbi se busca la secuencia de unidades del corpus que minimice la función coste compuesta por subcostes objetivo y de concatenación tal y como se muestra en las siguientes fórmulas:

$$C(t_1...t_n,u_1...u_n)=\alpha\sum_{i=1}^{n}C^T(t_i,u_i)+(1-\alpha)\sum_{i=1}^{n-1}C^C(u_i,u_{i+1})$$

$$C^T(t_i,u_i)=\sum_{j=0}^{P}w_j^T C_j^T(t_i,u_i)$$

$$C^C(u_i,u_{i+1})=\sum_{j=0}^{Q}w_j^C C_j^C(u_i,u_{i+1})$$

Donde t_i identifica las unidades objetivo y u_i las candidatas. C^T y C^C representan los costes objetivo y concatenación respectivamente; w_j es el j-ésimo peso que pondera una de las subfunciones existentes: P subfunciones de coste objetivo y Q de concatenación.

El coste objetivo está formado por la suma ponderada de los siguientes subcostes aplicados a nivel de semifonema y normalizados entre 0 y 1 (coste máximo):

- *Trifonema*: Valor discreto para potenciar el uso de unidades consecutivas en el corpus.
- *Contexto*: En una ventana de 5 fonemas, conjunto de valores discretos que caracterizan los tipos de fonemas adyacentes.
- *Pitch*: Distancia euclídea del contorno entonativo normalizando la duración.
- *Duración*: Valor absoluto de la diferencia de longitud. Se tiene en cuenta la posibilidad de modificar ligeramente la duración de unidades sonoras durante la generación de la forma de onda.
- *Acento*: Distancia a la sílaba acentuada más próxima.
- *Tipo de grupo fónico*: Interrogativo, inacabado, exclamativo, enunciativo, etc. Se ponderan especialmente las unidades finales y también la iniciales en las oraciones interrogativas.
- *Posición*: Posición relativa de la unidad dentro del grupo fónico.
- *Posición en la palabra*: Las unidades se agrupan en 4 categorías (inicio, medio, final y única) a nivel tanto de palabra como de sílaba.
- *Sonoridad*: Penaliza unidades sonoras marcadas como sordas.
- *Calidad fonética*: Penaliza unidades que si bien no son marcadas como "fuera de rango" su distancia acústica es superior a un umbral respecto al centro del cluster.

Los pesos de los costes objetivo se ajustan de forma automática utilizando un método similar al utilizado en el módulo prosódico. Se mide la distancia acústica entre unidades del corpus para tratar de predecirla como la suma ponderada de las funciones coste; resolviendo el valor de los pesos como un problema de regresión lineal múltiple.

Tras realizar una preselección con las unidades de menor coste objetivo, se calculan los costes de concatenación entre unidades no consecutivas en el corpus en base a los siguientes criterios:

- *Pitch*: Diferencia de pitch en el punto de concatenación.
- *Rango de pitch*: Para controlar saltos excesivos de pitch entre sílabas adyacentes y normalizando el coste respecto a los valores medios medidos para la voz original.
- *Duración*: Calculada únicamente a nivel de fonema (sumando duraciones de semifonema izquierdo y derecho) como diferencia respecto a la predicha por el modelo de duración.
- *Potencia*: Potencia en los extremos a concatenar y potencia media para unidades sonoras a nivel de fonema.
- *Sonoridad*: Penaliza unión entre unidades sonoras marcadas como sordas que no sean consecutivas. Para evitar ruidos de concatenación debidos a marcas de pitch erróneas.
- *Punto de unión*: Se penaliza ligeramente las uniones en partes no estacionarias, es decir, transiciones entre fonemas.
- *Distancia acústica*: Distancia euclídea entre la última y primera OLA de las unidades a concatenar. Se parametriza mediante 13 coeficientes MFCC añadiendo primeras y segundas diferencias. Para normalizar los valores se computan previamente las distancias medias de las transiciones entre semifonemas de la voz original.

Los pesos relativos a los costes de concatenación son ajustados de forma manual, aunque no se modificaron los valores existentes para las voces en Euskera, salvo el coste α que pondera la importancia entre los costes objetivo y de concatenación.

2.3.2. Generación de la forma de onda

Para mantener al máximo la naturalidad de las unidades seleccionadas, no se realiza ningún tipo de modificación de pitch, suavizando únicamente las uniones con información del cierre del pulso glotal. Sí que se realiza en cambio, una modificación de la duración de las unidades sonoras cuando la diferencia respecto al objetivo excede un umbral. Así como una ligera modificación de la energía para evitar cambios bruscos de volumen.

3. CONSTRUCCIÓN DE LA VOZ

Para la generación de la voz se partía del corpus upc_esma [4] grabado en castellano por una locutora. El corpus de 1 hora y 45 minutos de duración, está formado por 3 tipos de textos: frases fonéticamente balanceadas (30 minutos), párrafos fonéticamente balanceados (30 minutos) y párrafos literarios con una mayor variación prosódica (45 minutos).

Junto a los ficheros de audio se ha proporcionado la segmentación fonética, marcas a periodo de pitch, y señal del laringógrafo. Para la construcción de la voz sólo se ha utilizado la información relativa a la segmentación fonética, añadiendo de forma automática los alófonos aproximantes (B,D,G) y alguna otra diferencia respecto a nuestro transcriptor para castellano.

Debido a restricciones temporales no se llevó a cabo ningún tipo de revisión manual de las transcripciones. En cambio se realizó una detección automática de outliers en base a la siguiente información: *Score* devuelto por el segmentador, Duraciones, y Distancia Acústica de las unidades respecto al centro del cluster para cada fonema. Dicha información es utilizada también en el coste objetivo "calidad fonética", para evitar en la medida de lo posible, errores de etiquetado y/o pronunciaciones pobres.

El resto del procesado está igualmente automatizado y comienza con la normalización del

audio en base a la potencia media de las vocales. La curva entonativa es estimada mediante un sistema propio [5] basado en información cepstral y programación dinámica, que posteriormente se estiliza para cada semifonema mediante 3 puntos (inicio, punto más significativo y final). Para el marcado a periodo de pitch se utiliza la herramienta *epochs* de la suite *ESPS* corrigiendo en una etapa posterior, errores de pitch "halving/doubling" mediante la comparación del pitch local con nuestra estimación del contorno entonativo. Con la ayuda de la aplicación *sig2fv* del paquete *speech tools* se obtienen los coeficientes MFCC utilizados tanto en el coste de concatenación, como para entrenar los pesos de los costes objetivo y detectar outliers.

El resto de información necesaria se extrae a partir del módulo de procesamiento lingüístico.

4. CONCLUSIONES

En el presente artículo se han descrito las características esenciales del sistema AhoTTS, así como el proceso necesario para generar una nueva voz. Ésta ha sido la primera voz en castellano desarrollada en el laboratorio, y aunque a priori los resultados han sido bastante satisfactorios, existe un amplio margen de mejora.

Si bien todos los módulos del sistema pueden ser objeto de dicha mejora, quizá la generación de onda sea nuestro "talón de Aquiles". La solución pasa por la utilización de algún tipo de codificación que permita realizar modificaciones prosódicas y suavizado espectral en las concatenaciones, con poca degradación de la naturalidad.

5. AGRADECIMIENTOS

Agradecer el esfuerzo realizado por todos los sistemas participantes inscritos en la campaña de evaluación Albayzin TTS 2008.

6. BIBLIOGRAFÍA

[1] Taylor, P., Black, A. and Caley, R, "The architecture of the Festival Speech Synthesis System", *3rd ESCA Workshop on Speech Synthesis*, pp. 147-151, Jenolan Caves, Australia, 1998.

[2] Raux, A., Black, A., "A unit selection approach to f0 modeling and its application to emphasis", *Proc. of ASRU 2003*, St Thomas, US Virgin Is, 2003.

[3] Hunt, A., Black A., "Unit selection in a concatenative speech synthesis system using a large speech database", *Proc. of ICASSP*, vol. 1, pp. 373–376, 1996.

[4] Bonafonte, A., Moreno, A., "Documentation of the upc_esma spanish database", *TALP Research Center*, Universitat Politecnica de Catalunya, Carcelona; Spain , 2008.

[5] Luengo, I., Saratxaga, I., Navas, E., Hernáez, I., Sanchez, J., Sainz, I., "Evaluation Of Pitch Detection Algorithms Under Real Conditions", *Proc. of 32nd IEEE ICASSP*, pp. 1057-1060, Honolulu, 2007.

DESCRIPCIÓN DEL SINTETIZADOR DE VOZ COTOVÍA PARA LA EVALUACIÓN ALBAYZIN TTS 2008

Eduardo R. Banga, Francisco Méndez, Francisco Campillo, Gonzalo Iglesias, Laura Docío

Grupo de Teoría de la Señal
Dpto. Teoría de la Señal y Comunicaciones
Universidad de Vigo – 36310 Vigo

RESUMEN

Este artículo describe el estado actual del sintetizador de voz basado en corpus Cotovía, desarrollado en la Universidad de Vigo con la colaboración del Centro Ramón Piñeiro para la Investigación en Humanidades. Cotovía es un sistema en el que se efectúa una búsqueda combinada tanto de las unidades acústicas y entonativas como de la estructura prosódica, con el objetivo de generar la voz sintética de mayor calidad posible a partir del corpus disponible.

1. INTRODUCCIÓN

Cotovía es un sistema de conversión texto–voz en gallego y castellano englobado dentro de las técnicas de concatenación. A diferencia de la mayoría de los sintetizadores de voz actuales, en los que se van generando las características fonéticas en una serie de etapas secuenciales, lo cual en cierta manera implica asumir independencia entre ellas, en Cotovía se aplica el concepto de la selección de unidades ([1]) tanto en la generación de la forma de onda como en el modelado entonativo, y se lleva un paso más allá escogiendo la mejor combinación de unidades acústicas y entonativas. De la misma forma, en la selección entonativa también se consideran diferentes estructuras entonativas, sacando así partido de la variabilidad de la voz, que permite que un mismo mensaje se pueda realizar de diferentes maneras sin afectar ni a la naturalidad ni a la inteligibilidad.

En este artículo se explican las características principales del sintetizador en el momento de presentarse a la evaluación Albayzin TTS 2008. En primer lugar, en la sección 2 se exponen los pasos que se siguieron para procesar la voz y poder generar a partir de ella la información necesaria para la síntesis. En La sección 3 se describen los principales módulos del sistema, desde la etapa lingüística hasta la generación de la forma de onda, incluyendo los diferentes modelos de estimación de la prosodia. Finalmente, en la sección 4 se presentan las conclusiones.

Este trabajo ha sido subvencionado por el Gobierno Español mediante el proyecto coordinado AVIVAVOZ (TEC2006-13694-C03)

2. GENERANDO LA VOZ

Para la evaluación de sistemas de conversión de voz Albayzin 2008 se ha puesto a disposición de los participantes el corpus upc_esma [2] como material de desarrollo. Este corpus consta de aproximadamente 1h 45 min. de voz (mono, frecuencia de muestreo 16 KHz, resolución de 16 bits por muestra), dividido en 3 subcorpus: frases fonéticamente equilibradas (506 ficheros, aproximadamente 30 minutos), párrafos fonéticamente equilibrados (208 ficheros, aproximadamente 30 minutos) y 45 minutos (62 ficheros) de párrafos literarios. Puesto que los niveles de grabación de cada subcorpus eran distintos, se ha hecho una normalización, fichero a fichero, al 70 % del valor máximo.

Para cada subcorpus se han proporcionado los ficheros de audio, la señal del laringógrafo, los ficheros de texto, la transcripción fonética (SAMPA) y una segmentación fonética revisada manualmente para una parte (todas las frases y 144 de los 208 párrafos fonéticamente equilibrados) y otra automática de la totalidad de los corpus.

Debido a discrepancias entre la transcripción y segmentación fonéticas proporcionados y el procesado lingüístico realizado por Cotovía, que es el que se utiliza para construir las voces de nuestro sistema, se ha decidido no utilizar directamente ninguna de las segmentaciones proporcionadas. En su lugar se ha adaptado de forma semi–automática a nuestro sistema la parte segmentada manualmente, realizan-

do una nueva segmentación automática del resto del material de desarrollo.

El proceso de segmentación automática [3] se ha realizado en dos etapas. En primer lugar, utilizando los ficheros segmentados de forma manual se ha entrenado un conjunto de HMMs continuos para cada una de las unidades fonéticas. Debido a la cantidad limitada de este conjunto de datos de entrenamiento se han considerado modelos de monofonemas independientes del contexto, con una topología de tres estados de izquierda–a–derecha y 4 gaussianas por estado. Con los modelos entrenados se ha realizado a continuación una segmentación automática de aquellos ficheros de los que no se dispone segmentación fonética manual. Dicha segmentación se ha realizado a través de un alineamiento forzado de Viterbi en el que se permite la posibilidad de insertar silencios/pausas opcionales entre palabras. El front–end utiliza como características 12 coeficientes melcepstrum, la log–energía, y sus correspondientes derivadas primeras y segundas.

Se han marcado de forma manual en todo el corpus de desarrollo las fronteras entonativas. El proceso automático de generación de voz para Cotovía ha requerido unas 7 horas de ejecución en un servidor Intel©Xeon^{TM}a 3.06 GHz con 2 GB de memoria RAM. Como herramientas externas se ha utilizado el programa Praat para calcular las marcas de pitch para la estimación de la frecuencia fundamental, el Festival para obtener la envolvente espectral (12 coeficientes MFCC) y el HTK para la segmentación automática.

3. DESCRIPCIÓN DEL SISTEMA

3.1. Módulo lingüístico

Como en cualquier aplicación de este estilo, el módulo lingüístico consta de una serie de fases en las que el texto de entrada se acaba transformando en una secuencia de unidades acústicas objetivo caracterizadas por un conjunto de factores que se emplean posteriormente en las etapas de modelado prosódico y generación de la forma de onda. En este caso, en Cotovía tiene especial relevancia la etapa de análisis morfosintáctico, ya no sólo por su importancia en la decisión del carácter tónico o átono de las palabras, sino por su influencia en la estimación de los contornos entonativos, tal y como se comenta en la sección 3.2.1. El analizador morfosintáctico empleado ([4]) consta de un conjunto reducido de reglas lingüísticas fiables, que eliminan para cada palabra aquellas categorías que no son posibles en función de su contexto, seguido de un analizador estadístico de ventana deslizante, en el que se decide la categoría más probable combinando un modelo contextual que considera la probabilidad de una secuencia de categorías, y otro modelo léxico, que considera la probabilidad de que una palabra tenga una categoría dada.

3.2. Estimación de la prosodia

Al igual que la mayoría de los sintetizadores de voz actuales, Cotovía incluye módulos de estimación de la energía, la duración, la entonación y de inserción de rupturas prosódicas. Sin embargo, en lugar de tratarse de una serie de módulos que se van ejecutando secuencialmente, en algunos de ellos se emplea la variabilidad de la prosodia para conseguir una mejor estimación conjunta. Así, por ejemplo, es el propio módulo entonativo el que se encarga de parte del problema de la inserción de rupturas prosódicas. A continuación se explica más detalladamente cada uno de los modelos.

3.2.1. Entonación

Cotovía emplea un modelo entonativo basado en corpus ([5]), con el grupo acentual (secuencia de palabras átonas que acaba en una palabra tónica), como unidad básica para la concatenación. Las principales características del modelo son:

- Cada grupo acentual se representa según su posición en el grupo fónico y entonativo, la posición en la frase, los tipos de frontera prosódica que lo rodean, el número de sílabas, la posición del acento, el tipo de oración (enunciativa, exclamativa, interrogativa e inacabada), la duración, la etiqueta morfosintáctica de la palabra tónica del grupo, el sintagma al que pertenece, y el sintagma que lo sigue.
- El coste de objetivo tiene en cuenta básicamente las desviaciones de las características antes mencionadas con respecto a los valores estimados. Lo más destacable es el tratamiento de la información gramatical ([6]), que se emplea tanto para penalizar la introducción o no de una ruptura entonativa entre dos grupos acentuales (ver sección 3.2.2), como para modelar el énfasis de los acentos.

- El coste de concatenación considera únicamente la continuidad de frecuencia fundamental y la continuidad de frontera prosódica (para evitar que se unan dos grupos que linden con diferentes fronteras en sus contextos originales).

3.2.2. Estructura prosódica

A diferencia de [5], donde se consideraban únicamente dos niveles de ruptura prosódica (pausa y no pausa), en la actualidad se consideran tres niveles de ruptura: pausa, no ruptura, y ruptura entonativa, definida ésta última como un límite de grupo entonativo que no coincide con pausa.

Las inserciones de pausas y de rupturas entonativas se tratan como dos problemas independientes. En primer lugar se decide la mejor posición para las pausas por medio de un árbol de clasificación, empleando factores como una ventana de cinco etiquetas morfosintácticas y la distancia en sílabas a las pausas circundantes. Posteriormente, en cada iteración del algoritmo de selección de unidades entonativas se consideran grupos acentuales candidato que pueden ir seguidos o no de una ruptura entonativa (en función del contexto del que fueron extraídos), tal y como se muestra en la Figura 1, donde los círculos sombreados representan grupos candidatos con una ruptura entonativa. De esta forma, modificando la función de coste de objetivo entonativo para considerar la inserción de rupturas entonativas (en este caso por medio de información sintáctica y morfosintáctica, tal y como se explica en [6]), el propio algoritmo de selección escoge la mejor combinación de grupos acentuales y estructura prosódica, produciendo una entonación sintética más variable y relacionada con el significado del mensaje que se desea transmitir. Como ejemplo de esta variabilidad, la Figura 2 muestra varios contornos posibles para la oración "El non sabía se saír ou quedar na casa" (*Él no sabía si salir o quedarse en casa*). Como se puede observar, el contorno 1 tiene una ruptura prosódica alrededor del instante $t \approx 0{,}8\ s$.

3.2.3. Duración

Por lo que respecta a la duración segmental, los fonemas se agrupan en diez clases (vocales abiertas, vocales medias, vocales cerradas, oclusivas sonoras, oclusivas sordas, fricativas sordas, laterales, nasales, vibrantes y silencio), y para cada una de ellas se calcula un modelo basado en regresión lineal multivariante, empleando como factores la identidad de los fonemas que lo rodean en una ventana de tamaño cinco, la posición en la palabra y en el grupo fónico, el tipo de oración y el carácter tónico o átono.

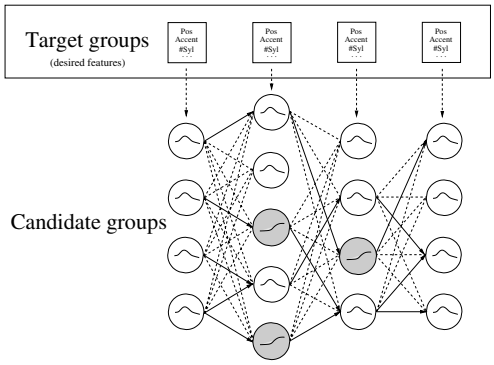


Figura 1. *Selección combinada de unidades entonativas y estructura prosódica ([6])*

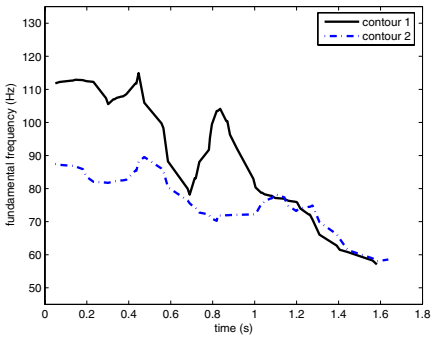


Figura 2. *Ejemplo de dos contornos con diferente estructura prosódica para una misma frase ([6])*

3.2.4. Energía

La energía es estimada por medio de un único modelo basado en regresión lineal multivariante, incluyendo como factores las clases del fonema y de los que lo rodean en una ventana de tamaño tres (según la misma clasificación del modelo de duración), la energía del fonema anterior, el número de sílabas desde el inicio y hasta el final del grupo fónico, y el carácter tónico o átono.

3.3. Selección de unidades acústicas

Tal y como sucedía con la entonación, en lo referente a las unidades acústicas Cotovía también esta basado en corpus, con el semifonema como unidad

básica para la concatenación. Además, dado que una misma frase se puede pronunciar con diferentes entonaciones sin afectar a su naturalidad, se repite la selección de unidades acústicas con cada uno de los N mejores contornos resultantes de la búsqueda entonativa ([5]), y se escoge la secuencia de semifonemas con mejor coste. Resumiendo, las principales características de la selección acústica son:

- Los semifonemas se parametrizan según su frecuencia fundamental, duración, energía, los fonemas que lo rodean, el carácter tónico, la posición en la palabra y en la frase, el tipo de frase a la que pertenece y los coeficientes cepstrales del semifonema y los que lo rodean.

- El coste de objetivo consta de dos partes ([7]). En primer lugar, la prosódica, donde se penalizan las desviaciones de la frecuencia fundamental, la duración y la energía con respecto a los valores predichos. Y en segundo lugar, la relacionada con la articulación del semifonema, donde se tienen en cuenta factores como los fonemas circundantes y la posición en la palabra y en la frase.

- El coste de concatenación considera la continuidad de frecuencia fundamental, energía y envolvente espectral.

3.4. Generación de la forma de onda

La señal sintética se genera mediante la concatenación de las formas de onda de las unidades acústicas escogidas. Cabe destacar que sólo se modifican prosódicamente aquellos semifonemas que se alejan de los valores estimados más de un umbral ($40\ ms$ para la duración y $5\ Hz$ para la frecuencia fundamental). Las unidades que no se tienen que modificar se copian directamente de la forma de onda original, recurriendo a las marcas de pitch únicamente en los puntos de concatenación.

4. CONCLUSIONES

En este artículo se ha descrito el estado actual del sintetizador de voz Cotovía, incluyendo tanto los pasos seguidos para la adición de una nueva voz, como el proceso que se sigue para la generación de la voz sintética. Durante la preparación de la voz para la evaluación quedó patente que pese a que la mayor parte del proceso es totalmente automático, es necesario desarrollar alguna herramienta que facilite el arduo proceso de revisión del etiquetado, sobre todo en lo referente a las fronteras prosódicas.

5. BIBLIOGRAFÍA

[1] A. Hunt y A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proceedings of ICASSP*, 1996, vol. 1, pp. 373–376.

[2] Antonio Bonafonte y Asuncion Moreno, "Documentation of the upc_esma spanish database," *TALP Research Center, Universitat Politecnica de Catalunya, Barcelona, Spain*, 2008.

[3] L. Docío y C. García, "Automatic segmentation of speech based on hidden markov models and acoustic features," in *Proceedings of 6th Internacional Conference on Spoken Language Processing*, 2000.

[4] F. Méndez, F. Campillo, E. R. Banga, y E. F. Rei, "Análisis morfológico estadístico en lengua gallega," *Procesamiento del lenguaje natural*, , no. 31, pp. 159–166, 2003.

[5] F. Campillo y E. R. Banga, "A method for combining intonation modelling and speech unit selection in corpus–based speech synthesis systems," *Speech Communication*, vol. 48, pp. 941–956, 2006.

[6] F. Campillo, J. van Santen, y E. R. Banga, "Combining phrasing and unit selection in intonation modelling," *IEE Electronic Letters*, vol. 44, no. 7, pp. 501–503, 2008.

[7] F. Campillo y E. R. Banga, "On the design of the cost functions for a unit selection speech synthesis," in *Proceedings of Eurospeech*, 2003, vol. 1, pp. 289–292.

DESCRIPCIÓN DEL SISTEMA I DE TELEFÓNICA I+D PRESENTADO A LA EVALUACIÓN ALBAYZIN'08 PARA CTV

M. Á. Rodríguez, J. G. Escalada y A. Armenta

División de Tecnología del Habla
Telefónica Investigación y Desarrollo

RESUMEN

Se hace una descripción general del CTV Sistema I de Telefónica I+D, presentado a la evaluación de conversores texto-voz Albayzín'08. Telefónica I+D ha presentado dos sistemas a la evaluación de conversores texto-voz (denominados como Sistema I y Sistema II). Ambos sistemas comparten la mayor parte de sus componentes, y se diferencian en las técnicas de procesado de señal empleadas para la codificación, modificación y síntesis de la señal de voz. Para facilitar la visión general de cada sistema de manera independiente, la descripción de cada uno de ellos es completa, de modo que las partes comunes aparecen en ambas descripciones. Se tratan las características del sistema relacionadas con la generación de la señal de voz sintética, y el proceso de creación de la voz a partir de la base de datos con las grabaciones proporcionadas por la organización.

1. CARACTERÍSTICAS GENERALES

El Sistema I de Telefónica I+D es un CTV multilingüe y multilocutor, basado en concatenación de unidades, que emplea una técnica de selección por corpus. Este sistema emplea técnicas de programación dinámica tanto para la selección de las unidades acústicas como para la selección de las unidades entonativas

Hasta el momento, los idiomas incorporados en nuestro CTV son español castellano, catalán, gallego, euskera, portugués de Portugal, español peruano, español mexicano, español iberoamericano neutro y portugués de Brasil.

En la figura 1 se representa la estructura general de este sistema, donde se aprecian los dos bloques principales (proceso lingüístico y síntesis de voz), más una serie de tablas lingüísticas (propias del idioma de funcionamiento) y de datos acústicos y prosódicos derivados de las grabaciones de un locutor humano de referencia.

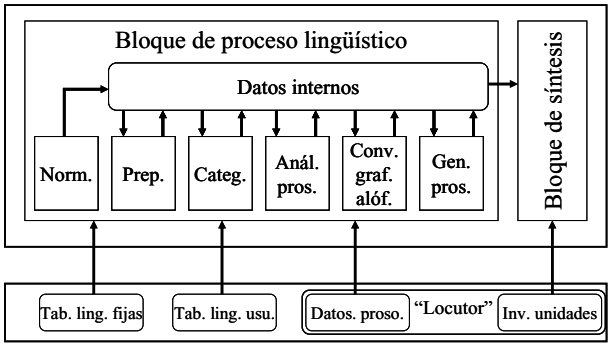


Figura 1. *Estructura del CTV Sistema I de Telefónica I+D*

2. CARACTERÍSTICAS DE LA SÍNTESIS DE VOZ

Como ya se ha dicho, el CTV Sistema I de Telefónica I+D es de los comúnmente denominados concatenativos. Genera la señal de voz sintética mediante selección y concatenación controlada de unidades acústicas.

Las unidades acústicas que maneja son difonemas que, generalmente, contienen el intervalo de señal de voz comprendido entre la parte estable de un sonido y la parte estable del siguiente sonido.

El inventario de unidades del CTV contiene multitud de opciones para cada una de las posibilidades de combinación entre dos sonidos, tantas como hayan sido incluidas en el proceso de creación de la voz. Si consideramos la combinación de sonidos a-b, el inventario contiene todos los difonemas correspondientes a las variantes de esa combinación de sonidos que aparecen en las grabaciones, y que se pueden distinguir entre sí por otras características como su contexto fonético, su F0, su duración, su localización dentro de la cadena hablada, su localización dentro de la palabra...

Para sintetizar un enunciado concreto, se hace una selección de difonemas mediante un procedimiento basado en corpus. El proceso lingüístico implementado dentro del CTV, que incorpora entre otros el módulo de análisis prosódico y el de generación de parámetros prosódicos (duraciones y contornos de F0), trata el texto para determinar cuál es la secuencia de sonidos que hay que generar, y les asigna a cada uno unos vectores de características (etiquetas) asociadas.

Con ello, se determina un "objetivo" para la síntesis: una secuencia de difonemas con características. El procedimiento de selección escoge la secuencia de difonemas recogida en el inventario de unidades que mejor se aproxima a la secuencia objetivo obtenida a partir del texto. Esta idea se concreta en un algoritmo de programación dinámica tipo Viterbi, que considera una serie de funciones de coste. La secuencia óptima es la que proporciona el coste mínimo.

Como ya se ha dicho, entre las características que se tienen en cuenta para la selección de unidades, aparte de la identidad de los sonidos implicados y su contexto, se incluyen otras de tipo prosódico. Seguidamente, describimos la forma en que se obtiene la información prosódica en nuestro sistema.

El módulo de análisis prosódico se ocupa de predecir y caracterizar los límites prosódicos en la lectura de un texto. Los límites tratados son tanto pausas (ortográficas o no) como frases entonativas, y se emplean para mejorar la generación de otros parámetros prosódicos (duración de los sonidos y contorno de F0). El funcionamiento del módulo de análisis prosódico no sólo tiene en cuenta características lingüísticas generales propias de un idioma determinado, sino que también se adapta al modo particular de hablar de un locutor humano de referencia. Este módulo ha sido personalizado usando las grabaciones suministradas para la construcción de la voz. Dentro del programa de las V Jornadas de Tecnología del Habla se presenta un artículo que describe este módulo ("Nuevo módulo de análisis prosódico del conversor texto-voz multilingüe de Telefónica I+D").

El modelo de duraciones de los sonidos es un modelo estadístico multiplicativo, cuyos parámetros se calculan para ajustarse a las duraciones recogidas en una base de datos de sonidos segmentados. Este modelo ha sido construido para la voz suministrada.

La generación del contorno de F0 se hace también mediante un procedimiento de selección por corpus de unidades entonativas elementales (patrones de F0). A partir del conjunto de patrones de F0 del corpus grabado, se compone la cadena de patrones más adecuada para construir el contorno de F0 correspondiente a las características obtenidas a partir del texto de entrada. Las unidades consideradas para la construcción de los contornos de F0 son los grupos acentuales (conjunto de sílabas comprendido entre el inicio de una sílaba tónica y el inicio de la siguiente tónica). El inventario de grupos acentuales también se obtuvo y etiquetó sobre los datos de la voz suministrada.

Las unidades acústicas almacenadas en el inventario se encuentran codificadas siguiendo un modelo sinusoidal propietario de Telefónica I+D. Se basa en el modelado del espectro de la señal usando componentes sinusoidales obtenidas con una técnica de análisis mediante síntesis, y que maneja la información de fase de las componentes sinusoidales para preservar la forma de onda de la señal original y asegurar la coherencia de fase durante la síntesis.

Para cada trama de voz, el procedimiento de análisis mediante síntesis (descrito en [1]) calcula una función de error $e(n)$ entre la señal original $s(n)$ y la señal estimada $\tilde{s}(n)$. Se considera que la señal estimada en cada trama es el resultado de una suma de sinusoides de amplitud y frecuencia constantes:

$$\tilde{s}(n) = \sum_{j=1}^{J} A_j \cos(\omega_j n + \phi_j)$$

De manera iterativa, en cada paso se obtiene una terna de valores de frecuencia ω_j, amplitud A_j y fase ϕ_j que minimiza la señal de error. Con los valores obtenidos se actualiza la señal estimada (se añade una componente sinusoidal más a las extraídas anteriormente), también se actualiza la señal de error, y se busca una nueva terna de valores hasta que se alcanza un umbral de aproximación.

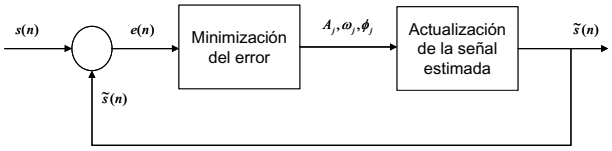


Figura 2. *Análisis sinusoidal mediante síntesis*

Una vez seleccionadas las unidades acústicas, se manipulan las tramas de voz codificada correspondientes con los propósitos siguientes:

- Modificar la duración y la entonación de los sonidos, en caso de que la diferencia entre los valores objetivo y los seleccionados supere determinados umbrales.
- Interpolar de manera adecuada los espectros de las tramas en los puntos de pegado, puntos en los que los difonemas en cuestión no se encontraban adyacentes en las grabaciones originales. La interpolación se realiza sobre la amplitud, el F0 y la envolvente espectral de las señales.

El modelo sinusoidal empleado permite hacer la interpolación y el ajuste de los parámetros prosódicos de manera robusta, manteniendo una continuidad y suavidad destacables en la voz sintética.

Cuando ya se han realizado las modificaciones necesarias en la secuencia de tramas de voz codificada, se efectúa la decodificación mediante la reconstrucción sinusoidal (suma de sinusoides de acuerdo al modelo empleado en el análisis), y solapamiento y suma de muestras entre tramas sucesivas, para obtener las muestras de voz sintética.

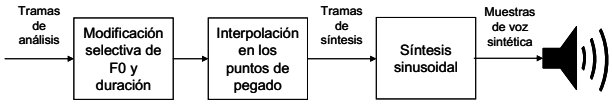


Figura 3. *Tratamiento de las tramas de las unidades acústicas seleccionadas*

3. PROCESO DE CREACIÓN DE LA VOZ

El primer paso para la creación de nuestro locutor sintético fue el tratamiento de los ficheros de texto por parte del proceso lingüístico de nuestro CTV. Ello nos permite obtener los datos necesarios para muchas otras de las tareas implicadas en la construcción, como la transcripción fonética, el etiquetado de características lingüísticas asociadas a los sonidos...

Se obtuvieron los contornos de F0 correspondientes a los ficheros de voz, que también son necesarios como información de entrada para la codificación de la voz, el etiquetado de características asociadas a los sonidos, la construcción del inventario de grupos acentuales...

Con los contornos de F0, se hizo el análisis de los ficheros de voz, para obtener los ficheros de tramas de voz codificada.

Después se realizó la segmentación en alófonos de los ficheros de voz, usando nuestras propias herramientas. Aunque podríamos haber adaptado el formato de la segmentación proporcionada por la organización (para que luego fuera válido como entrada al resto de herramientas implicadas en el proceso de construcción del locutor) resultaba para nosotros más directo emplear nuestras propias herramientas de segmentación. Nuestro segmentador (descrito en [2]) se basa en hacer reconocimiento forzado mediante HMM's, y en aplicar un conjunto de reglas de lógica difusa para el ajuste posterior de la segmentación proporcionada por el reconocedor.

Una vez segmentada y etiquetada la voz por procedimientos completamente automáticos, se procedió a la construcción de los datos del módulo de análisis prosódico que son dependientes del locutor, al cálculo de los parámetros del modelo de duraciones, y a la construcción del inventario de grupos acentuales manejado por el generador de contornos de F0.

A continuación, se hizo una primera construcción del inventario de unidades acústicas, y se obtuvo un locutor de partida.

Para la construcción de este primer locutor se emplearon todas las grabaciones disponibles.

Durante el proceso de construcción, nuestras herramientas nos permiten detectar puntos en los que puede haber algún problema o desajuste. Son puntos sospechosos, en los que puede haber algún tipo de error o no: sonidos de duración excesivamente corta o excesivamente larga, desajustes en la transcripción, valores de F0 llamativos...

La localización de esos puntos sospechosos nos permite hacer un repaso selectivo de porciones de los ficheros de voz y, en caso necesario, realizar las correcciones oportunas mediante herramientas semiautomáticas. Las correcciones pueden afectar a la segmentación, a los contornos de F0 o a cualquier otro aspecto del etiquetado. Evidentemente, es mucho mejor realizar un repaso exhaustivo de todos los ficheros de voz y su etiquetado, en toda su extensión. Pero cuando esto no es posible, el repaso selectivo de porciones de los ficheros ayuda a mejorar los resultados en un plazo más corto.

Dado el tiempo limitado del que se dispuso para la construcción de la voz, pudimos hacer este repaso selectivo a una parte de los ficheros: todos los ficheros de la parte "phonetically balanced sentences" (506 ficheros), más los 172 primeros ficheros de la parte "phonetically balanced paragraphs". Con este conjunto de 678 ficheros se hizo una nueva iteración de construcción de los datos relacionados con la prosodia y del inventario de unidades, y se obtuvo el locutor con el que se hizo la generación de los estímulos de voz sintética con los textos de prueba enviados por la organización.

Los elementos componentes del locutor sintético resultante fueron los siguientes:

- Datos necesarios para el procedimiento de determinación de límites prosódicos, empleados por el módulo de análisis prosódico.
- Parámetros del modelo de duraciones de los sonidos.
- Inventario de grupos acentuales para la construcción de los contornos de F0.
- Inventario de difonemas.

El inventario de grupos acentuales contiene 4.940 elementos. De ellos, la gran mayoría corresponden a grupos extraídos de frases de modalidad enunciativa (4.707). Del resto, 212 pertenecen a frases de modalidad interrogativa, y únicamente 21 a frases de modalidad exclamativa.

En cuanto al inventario de difonemas, contiene un total de 38.004 unidades, que contienen 420 identidades de difonemas distintas (considerando que la identidad viene dada por la etiqueta de los sonidos inicial y final del difonema). Las variantes de cada identidad varían en número, desde las 780 de la unidad más frecuente [D-e] hasta el caso de identidades con una sola variante (hay 22 identidades con una sola variante).

4. BIBLIOGRAFÍA

[1] E. B. George y M. J. T. Smith, "Speech Analysis/Synthesis and Modification Using an Analysis-by-Synthesis/Overlap-Add Sinusoidal Model", IEEE Transactions on Speech and Audio Processing, vol. 5, no. 5, pp. 389-406, septiembre 1997.

[2] D. Torre, M. Á. Rodríguez, J. G. Escalada, "Trying to Mimic Human Segmentation of Speech Using HMM and Fuzzy Logic Post-correction Rules", Proceedings of the 3rd ESCA/COCOSDA Workshop on Speech Synthesis, pp. 207-212, noviembre 1998.

DESCRIPCIÓN DEL SISTEMA II DE TELEFÓNICA I+D PRESENTADO A LA EVALUACIÓN ALBAYZIN'08 PARA CTV

J. G. Escalada, A. Armenta y M. Á. Rodríguez

División de Tecnología del Habla
Telefónica Investigación y Desarrollo

RESUMEN

Se hace una descripción general del CTV Sistema II de Telefónica I+D, presentado a la evaluación de conversores texto-voz Albayzín'08. Telefónica I+D ha presentado dos sistemas a la evaluación de conversores texto-voz (denominados como Sistema I y Sistema II). Ambos sistemas comparten la mayor parte de sus componentes, y se diferencian en las técnicas de procesado de señal empleadas para la codificación, modificación y síntesis de la señal de voz. Para facilitar la visión general de cada sistema de manera independiente, la descripción de cada uno de ellos es completa, de modo que las partes comunes aparecen en ambas descripciones. Se tratan las características del sistema relacionadas con la generación de la señal de voz sintética, y el proceso de creación de la voz a partir de la base de datos con las grabaciones proporcionadas por la organización.

1. CARACTERÍSTICAS GENERALES

El Sistema II de Telefónica I+D es un CTV multilingüe y multilocutor, basado en concatenación de unidades, que emplea una técnica de selección por corpus. Este sistema emplea técnicas de programación dinámica tanto para la selección de las unidades acústicas como para la selección de las unidades entonativas

Hasta el momento, los idiomas incorporados en nuestro CTV son español castellano, catalán, gallego, euskera, portugués de Portugal, español peruano, español mexicano, español iberoamericano neutro y portugués de Brasil.

En la figura 1 se representa la estructura general de este sistema, donde se aprecian los dos bloques principales (proceso lingüístico y síntesis de voz), más una serie de tablas lingüísticas (propias del idioma de funcionamiento) y de datos acústicos y prosódicos derivados de las grabaciones de un locutor humano de referencia.

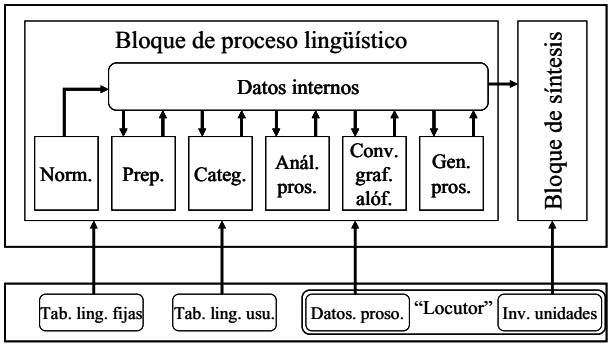


Figura 1. *Estructura del CTV Sistema II de Telefónica I+D*

2. CARACTERÍSTICAS DE LA SÍNTESIS DE VOZ

Como ya se ha dicho, el CTV Sistema II de Telefónica I+D es de los comúnmente denominados concatenativos. Genera la señal de voz sintética mediante selección y concatenación controlada de unidades acústicas.

Las unidades acústicas que maneja son difonemas que, generalmente, contienen el intervalo de señal de voz comprendido entre la parte estable de un sonido y la parte estable del siguiente sonido.

El inventario de unidades del CTV contiene multitud de opciones para cada una de las posibilidades de combinación entre dos sonidos, tantas como hayan sido incluidas en el proceso de creación de la voz. Si consideramos la combinación de sonidos a-b, el inventario contiene todos los difonemas correspondientes a las variantes de esa combinación de sonidos que aparecen en las grabaciones, y que se pueden distinguir entre sí por otras características como su contexto fonético, su F0, su duración, su localización dentro de la cadena hablada, su localización dentro de la palabra...

Para sintetizar un enunciado concreto, se hace una selección de difonemas mediante un procedimiento basado en corpus. El proceso lingüístico implementado dentro del CTV, que incorpora entre otros el módulo de análisis prosódico y el de generación de parámetros prosódicos (duraciones y contornos de F0), trata el texto para determinar cuál es la secuencia de sonidos que hay que generar, y les asigna a cada uno unos vectores de características (etiquetas) asociadas.

Con ello, se determina un "objetivo" para la síntesis: una secuencia de difonemas con características. El procedimiento de selección escoge la secuencia de difonemas recogida en el inventario de unidades que mejor se aproxima a la secuencia objetivo obtenida a partir del texto. Esta idea se concreta en un algoritmo de programación dinámica tipo Viterbi, que considera una serie de funciones de coste. La secuencia óptima es la que proporciona el coste mínimo.

Como ya se ha dicho, entre las características que se tienen en cuenta para la selección de unidades, aparte de la identidad de los sonidos implicados y su contexto, se incluyen otras de tipo prosódico. Seguidamente, describimos la forma en que se obtiene la información prosódica en nuestro sistema.

El módulo de análisis prosódico se ocupa de predecir y caracterizar los límites prosódicos en la lectura de un texto. Los límites tratados son tanto pausas (ortográficas o no) como frases entonativas, y se emplean para mejorar la generación de otros parámetros prosódicos (duración de los sonidos y contorno de F0). El funcionamiento del módulo de análisis prosódico no sólo tiene en cuenta características lingüísticas generales propias de un idioma determinado, sino que también se adapta al modo particular de hablar de un locutor humano de referencia. Este módulo ha sido personalizado usando las grabaciones suministradas para la construcción de la voz. Dentro del programa de las V Jornadas de Tecnología del Habla se presenta un artículo que describe este módulo ("Nuevo módulo de análisis prosódico del conversor texto-voz multilingüe de Telefónica I+D").

El modelo de duraciones de los sonidos es un modelo estadístico multiplicativo, cuyos parámetros se calculan para ajustarse a las duraciones recogidas en una base de datos de sonidos segmentados. Este modelo ha sido construido para la voz suministrada.

La generación del contorno de F0 se hace también mediante un procedimiento de selección por corpus de unidades entonativas elementales (patrones de F0). A partir del conjunto de patrones de F0 del corpus grabado, se compone la cadena de patrones más adecuada para construir el contorno de F0 correspondiente a las características obtenidas a partir del texto de entrada. Las unidades consideradas para la construcción de los contornos de F0 son los grupos acentuales (conjunto de sílabas comprendido entre el inicio de una sílaba tónica y el inicio de la siguiente tónica). El inventario de grupos acentuales también se obtuvo y etiquetó sobre los datos de la voz suministrada.

Las unidades acústicas almacenadas en el inventario se encuentran codificadas de acuerdo a un modelo de solapamiento y suma de ventanas (tramas) de la señal, basado en el dominio del tiempo. Es un modelo de los denominados OLA ("overlap and add") [1] que precisa conocer los instantes de tiempo de cada periodo en las zonas sonoras de la señal, en los que se centran las ventanas de análisis ("onsets" o "epochs"). En las zonas sordas, se toman ventanas a un intervalo fijo de 5 mseg. Para la determinación de estos instantes, se parte de la información obtenida por una herramienta de análisis sinusoidal de la voz (semejante a la descrita para el análisis de la voz en el Sistema I). Esta información es posteriormente filtrada y ajustada por otra herramienta que proporciona los "epochs" adecuados para el análisis OLA (buscando una adecuada localización de ventanas en las transiciones sonoro-sordo y sordo-sonoro).

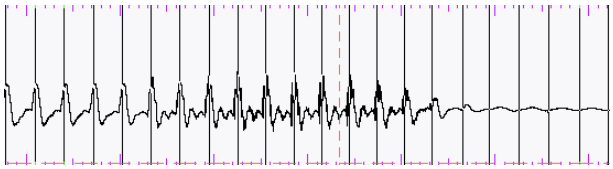


Figura 2. *Instantes de localización de las ventanas de análisis*

Una vez seleccionadas las unidades acústicas, se manipulan las tramas de voz codificada correspondientes con los propósitos siguientes:

- Modificar la duración y la entonación de los sonidos, en caso de que la diferencia entre los valores objetivo y los seleccionados supere determinados umbrales.
- Interpolar las tramas en los puntos de pegado (puntos en los que los difonemas en cuestión no se encontraban adyacentes en las grabaciones originales). La interpolación se limita a los valores de amplitud de las tramas.

El modelo de solapamiento y suma es relativamente simple y exige poca carga de cálculo, si bien no es tan flexible y robusto como el modelo sinusoidal de nuestro Sistema I en cuanto a hacer interpolaciones y modificaciones prosódicas. Cuando hay pocos pegados y el contorno de F0 se ajusta bien a las unidades seleccionadas, la calidad acústica es muy destacable.

Cuando ya se han realizado las modificaciones necesarias en la secuencia de tramas de voz codificada, se efectúa la decodificación combinando las muestras de ventanas consecutivas, para obtener las muestras de voz sintética.

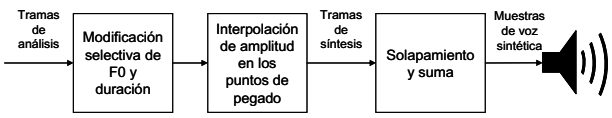


Figura 3. *Tratamiento de las tramas de las unidades acústicas seleccionadas*

3. PROCESO DE CREACIÓN DE LA VOZ

El primer paso para la creación de nuestro locutor sintético fue el tratamiento de los ficheros de texto por parte del proceso lingüístico de nuestro CTV. Ello nos

permite obtener los datos necesarios para muchas otras de las tareas implicadas en la construcción, como la transcripción fonética, el etiquetado de características lingüísticas asociadas a los sonidos...

Se obtuvieron los contornos de F0 correspondientes a los ficheros de voz, que también son necesarios como información de entrada para la codificación de la voz, el etiquetado de características asociadas a los sonidos, la construcción del inventario de grupos acentuales...

Con los contornos de F0, se hizo el análisis de tipo sinusoidal al que nos hemos referido anteriormente, para obtener la información de instantes de tiempo en los que se centran las ventanas de análisis OLA, seguido del adecuado filtrado y ajuste. No hemos empleado los instantes de tiempo proporcionados en la base de datos de voz suministrada, para mantener la compatibilidad de tipo de información y formatos con el resto de nuestras herramientas de construcción de locutores.

Acto seguido, se realizó el análisis de solapamiento y suma, determinando las ventanas de análisis sobre la señal de voz a partir de la información de los "onsets". Se emplean ventanas tipo Hanning.

Después se realizó la segmentación en alófonos de los ficheros de voz, usando nuestras propias herramientas. Aunque podríamos haber adaptado el formato de la segmentación proporcionada por la organización (para que luego fuera válido como entrada al resto de herramientas implicadas en el proceso de construcción del locutor) resultaba para nosotros más directo emplear nuestras propias herramientas de segmentación. Nuestro segmentador (descrito en [2]) se basa en hacer reconocimiento forzado mediante HMM's, y en aplicar un conjunto de reglas de lógica difusa para el ajuste posterior de la segmentación proporcionada por el reconocedor.

Una vez segmentada y etiquetada la voz por procedimientos completamente automáticos, se procedió a la construcción de los datos del módulo de análisis prosódico que son dependientes del locutor, al cálculo de los parámetros del modelo de duraciones, y a la construcción del inventario de grupos acentuales manejado por el generador de contornos de F0.

A continuación, se hizo una primera construcción del inventario de unidades acústicas, y se obtuvo un locutor de partida.

Para la construcción de este primer locutor se emplearon todas las grabaciones disponibles.

Durante el proceso de construcción, nuestras herramientas nos permiten detectar puntos en los que puede haber algún problema o desajuste. Son puntos sospechosos, en los que puede haber algún tipo de error o no: sonidos de duración excesivamente corta o excesivamente larga, desajustes en la transcripción, valores de F0 llamativos...

La localización de esos puntos sospechosos nos permite hacer un repaso selectivo de porciones de los ficheros de voz y, en caso necesario, realizar las correcciones oportunas mediante herramientas semiautomáticas. Las correcciones pueden afectar a la segmentación, a los contornos de F0 o a cualquier otro aspecto del etiquetado. Evidentemente, es mucho mejor realizar un repaso exhaustivo de todos los ficheros de voz y su etiquetado, en toda su extensión. Pero cuando esto no es posible, el repaso selectivo de porciones de los ficheros ayuda a mejorar los resultados en un plazo más corto.

Dado el tiempo limitado del que se dispuso para la construcción de la voz, pudimos hacer este repaso selectivo a una parte de los ficheros: todos los ficheros de la parte "phonetically balanced sentences" (506 ficheros), más los 172 primeros ficheros de la parte "phonetically balanced paragraphs". Con este conjunto de 678 ficheros se hizo una nueva iteración de construcción de los datos relacionados con la prosodia y del inventario de unidades, y se obtuvo el locutor con el que se hizo la generación de los estímulos de voz sintética con los textos de prueba enviados por la organización.

Los elementos componentes del locutor sintético resultante fueron los siguientes:

- Datos necesarios para el procedimiento de determinación de límites prosódicos, empleados por el módulo de análisis prosódico.
- Parámetros del modelo de duraciones de los sonidos.
- Inventario de grupos acentuales para la construcción de los contornos de F0.
- Inventario de difonemas.

El inventario de grupos acentuales contiene 4.940 elementos. De ellos, la gran mayoría corresponden a grupos extraídos de frases de modalidad enunciativa (4.707). Del resto, 212 pertenecen a frases de modalidad interrogativa, y únicamente 21 a frases de modalidad exclamativa.

En cuanto al inventario de difonemas, contiene un total de 38.004 unidades, que contienen 420 identidades de difonemas distintas (considerando que la identidad viene dada por la etiqueta de los sonidos inicial y final del difonema). Las variantes de cada identidad varían en número, desde las 780 de la unidad más frecuente [D-e] hasta el caso de identidades con una sola variante (hay 22 identidades con una sola variante).

4. BIBLIOGRAFÍA

[1] E. Moulines, F. Charpentier, "Pitch Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones", Speech Communication 9, pp 453-467, 1990.

[2] D. Torre, M. Á. Rodríguez, J. G. Escalada, "Trying to Mimic Human Segmentation of Speech Using HMM and Fuzzy Logic Post-correction Rules", Proceedings of the 3rd ESCA/COCOSDA Workshop on Speech Synthesis, pp. 207-212, noviembre 1998.

EL SISTEMA DE IDENTIFICACIÓN DE LA LENGUA DE PRHLT

Míriam Luján-Mares, Vicent Tamarit, Roberto Paredes,
Vicent Alabau, Carlos-D. Martínez-Hinarejos

Instituto Tecnológico de Informática, Universidad Politécnica de Valencia,
Camino de Vera, s/n, 46022, Valencia, Spain

RESUMEN

El artículo explica los detalles de los sistemas de identificación de la lengua del grupo de investigación *Pattern Recognition and Human Language Technology* (PRHLT).

1. INTRODUCCIÓN

La tarea consiste en reconocer la lengua de un fragmento de habla, adquirido de un programa de televisión de una de las 4 lenguas objetivo (castellano, catalán, euskera y gallego) o una lengua desconocida.

La verificación de la lengua es un problema que puede ser estudiado desde dos puntos de vista: utilizando la información espectral de la señal o la información sintáctica y semántica de dicha señal.

Los sistemas que se presentan en este trabajo se basan solamente en la información espectral de la señal, ya que han sido entrenado únicamente con los datos proporcionados. Dichos datos contaban sólo con habla efectiva y su idioma correspondiente. Por tanto, no se han empleado corpora externo para la creación de modelos de lenguaje utilizables para un reconocimiento de habla, ni tampoco para mejorar modelos basados en señal.

Para ambos sistemas los datos han sido preprocesados con técnicas del estado del arte en verificación de la lengua. Los sistemas se basan en dos clasificadores diferentes para llevar a cabo la identificación: uno de ellos por k-vecinos y el otro por *Gaussian mixture models* (GMMs).

2. DATOS DE ENTRENAMIENTO

Los datos de entrenamiento provienen de programas de televisión (informativos, documentales, debates, entrevistas, reportajes, magazines, etc.). Dichos datos sólo cuentan con las cuatro lenguas objetivo (castellano, catalán, euskera y gallego).

Las señales se han adquirido a través de un mismo dispositivo (una grabadora digital Roland Edirol R-09) y se han depositado en ficheros WAV (monocanal, 16 Khz, 16 bits/muestra). Las grabaciones incluyen diversos tipos de habla: leída, planificada, conversacional formal, espontánea, etc. Asimismo, aunque la relación señal-ruido (SNR) es bastante buena en casi todos los casos, las condiciones ambientales y de canal son también muy diversas: entrevistas en estudio sin ruido de fondo, reportajes desde la calle, desde una fiesta, desde una manifestación, llamadas telefónicas en directo, reportajes con una ligera música de fondo, programas concurso o de humor con risas y aplausos, etc.

El conjunto de entrenamiento consta de aproximadamente 8 horas por lengua (unas 32 horas en total), en ficheros de duración variable. Estos ficheros contienen mayoritariamente voz (en condiciones ambientales y de canal diversas) y sólo pequeños fragmentos de silencio o ruido de fondo [1].

Para encontrar los parámetros idóneos para el sistema hemos utilizado un conjunto de desarrollo disjunto e independiente del grupo de entrenamiento. Esto significa, por ejemplo, que un programa utilizado para entrenamiento no aparecerá en desarrollo. Estos datos de desarrollo provienen también de programas de televisión y tienen las mismas características de grabación.

3. SISTEMA PRIMARIO

Este sistema se basa en el empleo de GMMs entrenados con SDCs.

3.1. Preproceso

Para la identificación del idioma utilizando GMMs se pueden utilizar los Shifted Delta Cepstrum (SDC) [2], una extensión de las derivadas de los cepstrales. Estos nuevos vectores de características se calculan sobre una ventana de cepstrales, buscando extraer información fonética de más largo alcance. Los cepstrales de partida fueron extraídos utilizando el parametrizador de audio de iatros [3], que fue ampliado con un nuevo módulo para generar los SDCs. Los SDCs se definen a partir de cuatro parámetros, N-d-P-k, donde N indica el número de cepstrales originales, d se refiere al número de cepstrales utilizados para el cálculo de las derivadas y P hace referencia a la distancia entre los sucesivos cálculos de derivadas. El último

Este trabajo ha sido subvencionado por VIDI-UPV bajo las becas FPI del programa PAID06 y por el EC (FEDER), por el proyecto subvencionado por el Ministerio de Educación y Ciencia Español TIN2006-15694-C02-01 y por el programa Español Consolider Ingenio 2010: MIPRCV (CSD2007-00018).

valor, k, especifica el número de derivadas calculadas a partir de un vector de cepstrales.

3.2. Descripción del sistema

Como sistema primario presentamos un modelo de mixturas de gaussianas (GMM) para llevar a cabo el reconocimiento. Dicho sistema ha sido entrenado con el *toolkit* HTK [4] y los SDCs obtenidos en el preproceso. Para obtener los SDCs se llevó a cabo un estudio para determinar los parámetros N-d-P-k idóneos para dicha experimentación. Finalmente, los SDCs se obtuvieron con valores de 7-1-3-7, resultando vectores de características de 49 dimensiones calculadas cada 210ms. Se entrenaron desde 2 gaussianas hasta 8192, determinando que 4096 gaussianas es el número de gaussianas con las que se obtiene una mejor tasa de acierto para el conjunto de desarrollo.

Para llevar a cabo el reconocimiento se ha implementado un módulo dentro del reconocedor propio iatros [3]. Dicho módulo está concretamente diseñado para llevar a cabo reconocimiento con GMMs, basándose en sumar las probabilidades de emisión por el GMM de todos los vectores de características de una señal dada.

4. SISTEMA ALTERNATIVO

Este sistema se basa en clasificar las muestras de test con la técnica de k-vecinos sobre los vectores de características en nuestro caso, estos vectores son cepstrales.

4.1. Preproceso

Los vectores de cepstrales son la característica contínua utilizada para el reconocimiento del habla [5]. Se obtienen de la señal, aplicando una ventana que se desliza a lo largo de la misma a partir de la cual se calcula el vector de cepstrales. Los cepstrales eliminan de la señal los rasgos propios del locutor y resaltan la información fonética. Para su extracción se definen diversos parámetros, como el tamaño de la ventana (normalmente unos 0.025 segundos), la frecuencia de submuestreo (cada cuánto extraemos una ventana), tamaño de la transformada de Fourier, el factor de pre-énfasis y el número de elementos del vector (típicamente 11). Adicionalmente pueden calcularse también la primera y segunda derivada en el tiempo de estos vectores, obteniendo así los vectores de 33 elementos utilizados.

4.2. Descripción del sistema

Este sistema utiliza en primer lugar los datos de entrenamiento y el algoritmo c-medias [6] para aprender un *codebook* de C *codewords*. Con dicho *codebook* aprendido se realiza la cuantificación vectorial de los ficheros de audio.

Los ficheros de audio pasan a ser representados mediante un único vector. Dicho vector es el histograma de la frecuencia de aparición de cada uno de los *codewords* en dicho fichero de audio. Por lo tanto, el tamaño de dicho vector es C. Este proceso se aplica a todos los ficheros de audio de entrenamiento y test.

Con los vectores obtenidos se puede utilizar cualquier técnica clasica de clasificación, por ejemplo, la técnicas de k-vecinos [7]. El resultado se puede mejorar con un aprendizaje discriminativo.

Para aplicar este aprendizaje discriminativo se aprende una base de proyección de C a d dimensiones [8]. Dicha proyección se obtiene sólo con las muestras de entrenamiento, pero se aplica tanto a vectores de entrenamiento como de test.

Por las pruebas realizadas se puede afirmar que en el espacio reducido de d dimensiones la clasificación mejora. Por tanto, en este punto podemos llevar a cabo la clasificación de las muestras de test con la técnica de k-vecinos.

5. BIBLIOGRAFÍA

[1] "Kalaka," Speech database created for the 2008 Language Recognition Evaluation on Spanish Languages, organized by the Spanish Network on Speech Technology. Produced by the Software Technologies Working Group (GTTS, http://gtts.ehu.es), University of the Basque Country.

[2] B. Bielefeld, "Language identification using shifted delta cepstrum," in *Proc. Fourteenth Annual Speech Research Symposium*, 1994.

[3] Míriam Luján, Vicent Tamarit, Vicent Alabau, Carlos-D. Martínez-Hinarejos, Moisés Pastor, Alberto Sanchís, y Alejandro Toselli, "iatros: A speech and handwritting recognition system," *V Jornadas en Tecnología del Habla*, Noviembre Bilbao, 2008.

[4] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, y P. Woodland, *The HTK Book*, CUED, UK, v3.2 edition, July, 2004.

[5] Lawrence Rabiner y Biing-Hwang Juang, *Fundamentals of Speech Recognition*, Prentice Hall Ptr, 1993.

[6] R. Duda y P Hart, *Pattern Classification and Scene Analysis*, Wiley, 1973.

[7] B. S. Kim y S. B. Park, "A fast k nearest neighbor finding algorithm based on the ordered partition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 761–766, 1986.

[8] Mauricio Villegas, Roberto Paredes, Alfons Juan, y Enrique Vidal, "Face verification on color images using local features," in *Proceedings of the IEEE Computer Society Workshop on Biometrics, in association with CVPR 2008*, Anchorage, AK, USA, June 2008, IEEE Computer Society.

EVALUACIÓN ALBAYZÍN-08 DE SISTEMAS DE VERIFICACIÓN DE LA LENGUA: SISTEMA DEL GRUPO SOFTLAB DE LA UC3M

M.J. Poza, B. Ruiz, L. Puente y D. Carrero

Grupo SoftLab, Universidad Carlos III de Madrid
mjpoza@entornotec.com, bruiz@inf.uc3m.es, lpuente@it.uc3m.es, dcarrero@di.uc3m.es

RESUMEN

El grupo SOFTLAB de la UC3M ha desarrollado un sistema de identificación de lengua basado en GMMs, y lo ha presentado a la Evaluación ALBAYZIN-08 de Sistemas de Verificación de la Lengua, organizada por el Grupo de Trabajo en Tecnologías Software de la UPV/EHU, en el marco de las V Jornadas en Tecnología del Habla organizadas por la Red Temática en Tecnologías del Habla y el Grupo AHOLAB de Procesado de Señal de la UPV/EHU. Este artículo describe el sistema de identificación presentado a dicho plan de evaluación.

1. INTRODUCCIÓN

Un sistema de identificación de lengua es, básicamente, un sistema de reconocimiento de patrones que hace uso de la señal de voz de un discurso o conversación inteligible de cualquier individuo para decidir si el idioma utilizado en la conversación o discurso es alguno de los que el sistema reconoce.

Todo sistema de clasificación basado en reconocimiento de patrones (ver Figura 1) tiene una fase previa de entrenamiento en que se capturan y modelan las características distintivas de cada uno de los 'usuarios' del sistema (en este caso, idiomas): en esta fase se generan los patrones o modelos que luego se usarán durante el funcionamiento normal, en la toma de decisión sobre si el discurso a clasificar está pronunciado en alguno de los idiomas para los que el sistema ha sido entrenado:

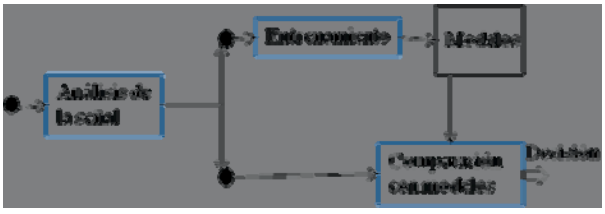


Figura 1. *Sistema de clasificación basado en reconocimiento de patrones*

Generalmente se distingue entre verificación e identificación de la lengua. Dado un segmento de habla Y, y un hipotético idioma I, la tarea de verificación consiste en determinar si el segmento Y fue pronunciado en el idioma I. La tarea de identificación considera un conjunto cerrado de idiomas (I_1,I_2... I_N), y trata de determinar en cuál de ellos fue pronunciado el segmento de habla Y.

Nuestro sistema de identificación de lengua actúa como un clasificador de patrones. Cada patrón está formado por un conjunto de características o parámetros, extraídos de una determinada locución, y es 'enfrentado' o comparado con distintos modelos generados para cada idioma. La salida del clasificador ofrece una verosimilitud o una medida de distancia, entre el patrón de entrada y el modelo; y en última instancia una decisión, basada en un umbral, que clasifica la locución como perteneciente o no a un determinado idioma.

Cada modelo de un idioma es generado mediante patrones extraídos de locuciones del mismo; siendo necesario que cada uno de los idiomas involucrados en el sistema, disponga de su propio conjunto de datos de entrenamiento. Este conjunto será distinto del conjunto de datos sobre los cuales se pruebe el sistema.

Es bien conocido, que una de las causas principales que degradan el rendimiento de los sistemas de reconocimiento basados en voz se debe a la variabilidad acústica entre los conjuntos de entrenamiento y test. Esta variabilidad no sólo es debida a la diferencia acústica en los distintos idiomas (en sistemas de reconocimiento de idioma), sino también a otro tipo de variaciones, como las distorsiones producidas por los distintos canales, las diferencias entre micrófonos, el ruido ambiental, etc. El uso de técnicas de compensación de canal, ya sea sobre el audio, los parámetros a modelar o el propio modelo, mejora las tasas de reconocimiento. Estas técnicas se basan en eliminar información no discriminativa que varía de forma no controlada entre las distintas locuciones.

Existen diversas y muy variadas técnicas aplicadas a la compensación o eliminación de la variabilidad de canal; Nuestro sistema se basa en la técnica CMS (cepstral mean substraction), también conocido como CMN (cepstral mean normalization). En una

parametrización basada en coeficientes cepstrales [1], una locución, es dividida en ventanas de tiempo, de la cual son extraídos un cierto número de coeficientes cepstrales. CMN se basa en sustraer para cada coeficiente cepstral extraído la media de dicho coeficiente a lo largo de toda la locución. De esta forma se reduce la distorsión introducida por elementos de variación lenta, como por ejemplo ruido estacionario.[2].

Los sistemas de reconocimiento de idioma sobre habla 'espontánea' tienen más limitaciones que los sistemas que usan voz 'limpia', como son el ruido de las conversaciones (ruidos de sillas, música, conversaciones de fondo, etc) y los silencios en las mismas. Además cuanto mayor sea el nivel de reconocimiento requerido mayor tendrá que ser la duración de la conversación.

2. DESCRIPCIÓN DEL SISTEMA

El diagrama de bloques del sistema verificador de idioma se muestra en la siguiente figura:

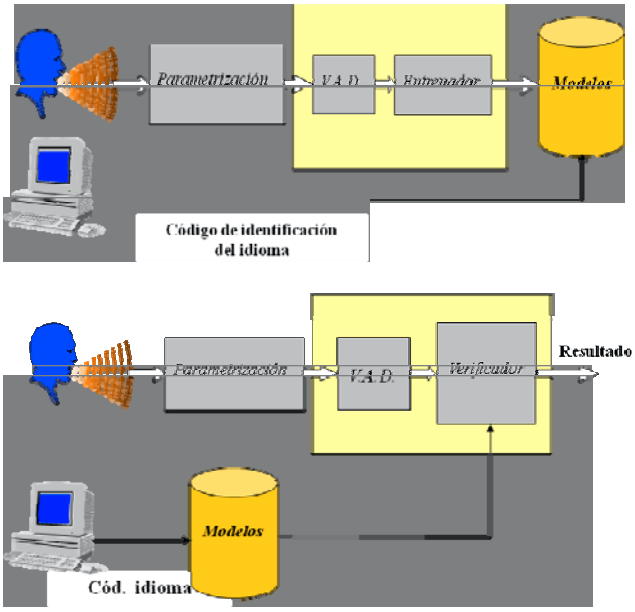


Figura 2. *Fases de entrenamiento y verificación del sistema de identificación de idioma.*

El reconocimiento automático del idioma comparte muchas técnicas con el reconocimiento de locutor, lo que hace que ambos problemas suelan ser abordados de un modo similar. Nuestro sistema se compone, básicamente, de tres módulos funcionales: parametrización (o extracción de características), detector de actividad vocal (para eliminar 'silencios' de la señal de entrada al entrenador y al verificador) y generación/comparación con mdelos.

2.1. Extracción de características

La extracción de características es el primer paso en cualquier sistema de reconocimiento automático. y comienza por la captación de la señal sobre la que se desea trabajar (voz) mediante un sensor, que en este caso será un sensor apto para la señal de voz (micrófono, teléfono, etc.).

La extracción de parámetros de nuestro sistema está basada en el análisis a corto plazo de la señal de voz, usando una de las técnicas más habituales en reconocimiento automático de voz: el análisis MFCC *(Mel-Frequency Cepstral Coefficients)*, junto con varias medidas de entropía [3] (la entropía de la señal, de su potencia y del logaritmo de su potencia, así como sus primeras derivadas respectivas). Así, los vectores de características estarán principalmente compuestos por algunos parámetros cepstrum y cepstrum diferenciales, pero también se añadirá otro tipo de información, como la energía, su derivada, y los valores de las entropías antes mencionados.

Sobre estos parámetros pueden llevarse a cabo mejoras con el fin de paliar las distorsiones sufridas por la señal de voz y mejorar el rendimiento de los sistemas, estas mejoras son conocidas como compensaciones de canal [4] y en nuestro sistema se utiliza la conocida como CMN *(Cepstral Mean Normalization)* [1], consistente en eliminar la media de los parámetros cepstrales con el fin de eliminar de ellos información de variación más lenta que la de la señal de voz, como puede ser la introducida por el ruido de fondo (ruido aditivo, no correlado con la señal de voz y casi estacionario en el tiempo). Esa técnica de eliminación de medias se ha empleado también para la energía y para la entropía de la señal.

La Figura 3 muestra un diagrama de bloques del módulo extractor de características.

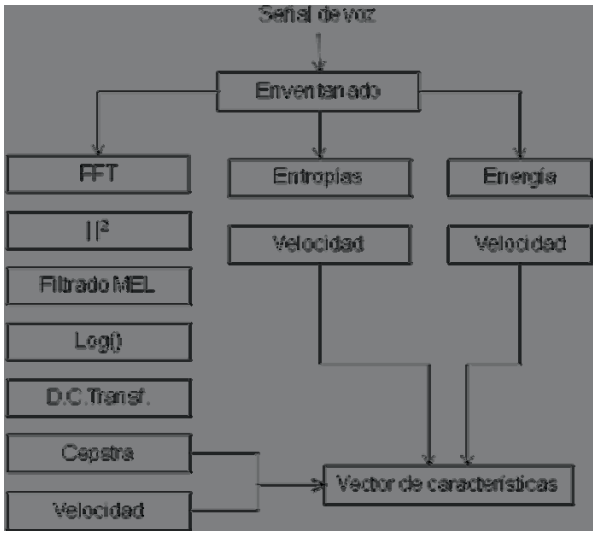


Figura 3. *Bloque extractor de características*

2.2. Detector de actividad vocal

Una señal vocal comprende hasta un 60% de silencio o de ruido de fondo, y ese 60% de la señal tiene características muy similares para todos los idiomas: por ello, eliminar de la señal de entrada esos tramos de

'solo-ruido' aumentará la eficiencia y la eficacia de los sistemas de reconocimiento, por aumentar el poder de discriminación entre los distintos idiomas (eliminando 'partes comunes') y disminuir la carga de trabajo del sistema. El módulo detector de actividad vocal se encarga de esa eliminación de silencios y/o ruidos.

Clásicamente hay dos formas de abordar la eliminación de silencios en una señal [5] [6]:

- Basándose en umbrales (umbrales para la energía, umbrales para los 'cruces por cero', etc: información 'local')
- Usando técnicas de clasificación (como pueden ser los modelos ocultos de Markov o las redes neuronales): esta familia de detectores de silencios se basa en características estadísticas de la señal en vez de en características locales.

El primer grupo de detectores de actividad vocal es el más utilizado, por su simplicidad de implementación, pero precisa ajustar varios umbrales para que funcione adecuadamente, y que esos umbrales se vayan adaptando a las características de la señal de entrada: a las variaciones ambientales. El segundo grupo de detectores elimina esa dependencia, con el coste de una fase previa de entrenamiento del clasificador.

El sistema propuesto por el grupo SOFTLAB incluye un módulo detector de actividad vocal basado en la energía de la señal, implementado como una máquina de 6 estados cuyas transiciones se realizan cuando la señal cruza por alguno de los 4 umbrales de energía que utiliza (dos para decidir transición silencio-voz y otros dos para decidir voz-silencio), dentro de ciertas limitaciones temporales.

2.3. SVMs: Módulo Decisor

En los últimos años se ha observado un incremento considerable de la utilización de las Support Vector Machines dentro del Aprendizaje Automático [7][8][9]. Su alto rendimiento hace de las SVM una de las metodologías más sólidas en este área.

Las SVM son básicamente clasificadores para 2 clases. Tienen como objetivo obtener un hiperplano óptimo capaz de separar lo mejor posible dos clases distintas de los puntos de entrada. Como un clasificador de una sola clase, la descripción dada por los datos de los vectores de soporte es capaz de formar una frontera de decisión alrededor del dominio de los datos de aprendizaje con muy poco o ningún conocimiento de los datos fuera de esta frontera. Los datos son mapeados por medio de un *kernel* Gaussiano u otro tipo de *kernel* a un espacio de características en un espacio de más dimensiones, donde se busca la máxima separación entre clases. Esta función de frontera, cuando es traída de regreso al espacio de entrada, puede separar los datos en distintas clases, cada una formando un agrupamiento.

El kernel utilizado en el sistema de identificación de idioma presentado por los autores del presente artículo es el RBF (Radio Basis Function) o gaussiana:

$$K(x, z) = \exp(- |x - z|^2 / \sigma^2)$$

3. AGRADECIMIENTOS

Este trabajo ha sido posible gracias a la financiación de los Ministerios de Industria (proyecto SEGUR@) y de Educación y Ciencia (proyecto TEC2006-12365-C02-01).

4. BIBLIOGRAFÍA

[1] Furui S., ”Cepstral analysis technique for automatic speaker verification”. IEEE Transactions on Speech and Audio Processing,Vol.ASSP-29,No.2 .April 1981

[2] Liu F., Stern R., Huang X. and Acero A. “Efficient Cepstral Normalization for Robust Speech Recognition”. Proceedings of ARPA Human Language Technology Workshop, March 1993.

[3] Cunha, S., Correira, S., Aguilar, B. "Pathological voice discrimination based on entropy measurements", BIODEVICES 2008: International Conference on Biomedical Electronics and Devices (Madeira, Portugal).

[4] Vaier C., Colibro D., Castaldo F., Dalmasso E., Laface P., "Channel factors compensation in model and feature domain for speaker recognition”. Odissey 2006

[5] Stadermann J., Stahl V., Rose G., “Voice activity detection in noisy environments”, Proc. of Eurospeech 2001.

[6] Carli G., Gretter R., “A start-end point detection algorithm for a real-time acoustic front-end based on DSP32C VME board”, ICSPAT’92.

[7] Campbell W. M., "Generalized linear discriminant sequence kernels for speaker recognition", Proceedings of the International Conference on Acoustics Speech and Signal Processing, 2002, pp. 161.164

[8] Vapnik V., "The nature of statistical learning theory". Springer, 1995.

[9] Burges, C. "A Tutorial on Support Vector Machines for Pattern Recognition", 1998 Journal on Data Mining and Knowledge Discovery, Vol 2, pp 121-167.

GENERACIÓN DE UNA VOZ SINTÉTICA EN CASTELLANO BASADA EN HSMM PARA LA EVALUACIÓN ALBAYZÍN 2008: CONVERSIÓN TEXTO A VOZ

R. Barra-Chicote[1]*, J. Yamagishi*[2]*, J. M. Montero*[1]*, S. King*[2]*, S. Lufti* [1]*, J. Macias-Guarasa*[3]

Grupo de Tecnología del Habla, Universidad Politécnica de Madrid[1],
Center for Speech Technology Research, University of Edinburgh[2],
Universidad de Alcalá[3]

RESUMEN

Este artículo describe el proceso de generación de una voz en castellano utilizando el corpus *UPC ESMA* de UPC proporcionado por la *Evaluación Albayzín 2008: Conversión Texto a Voz*. Se ha implementado una voz basada en selección de unidades mediante el paquete *Multisyn* de *Festival* y otra basada en *Hidden Semi-Markov Models* (HSMM) mediante *HTS*. Tras una breve evaluación de la calidad de ambas voces, se detallan las características principales de la voz basada en HSMM, sistema final presentado a la evaluación.

1. INTRODUCCIÓN

La *Evaluación Albayzín 2008: conversión texto a voz* tiene como objetivo la evaluación de las técnicas de síntesis actuales aplicadas al castellano, del mimo modo que la competición Blizzard Challenge para inglés y chino mandarín.

Cada equipo participante debe proporcionar una voz generada a partir del corpus proporcionado en un plazo de 7 semanas. Posteriormente deben sintetizar un conjunto de ejemplos de test, que serán evaluados perceptualmente, de forma conjunta con los del resto de equipos, en términos de similaridad con la voz original, naturalidad e inteligibilidad.

2. CORPUS

El corpus UPC ESMA [1] proporcionado para la evaluación del sistema consiste en las grabaciones de un conjunto de textos leídos con estilo neutro por parte de una locutora profesional.

El corpus proporciona 506 frases fonéticamente balanceadas (30 minutos), 208 párrafos de longitud media fonéticamente balanceados (30 minutos) y 62 párrafos literarios de mayor longitud (45 minutos).

Además del audio, señal de voz y señal del laringógrafo, se cuenta con el texto de referencia, la trascripción fonética y un diccionario con la información léxica. Con el corpus se proporciona la segmentación fonética y la marcación automática de *pitch*. Adicionalmente se dispone de la marcación manual de un subconjunto de la base de datos.

Este trabajo ha sido parcialmente financiado por el M.E.C. y los proyectos proyecto ROBONAUTA (DPI2007-66846-C02-02), EDECAN (TIN2005-08660-C04-04).

3. ANÁLISIS LINGÜÍSTICO

Para la realización del análisis lingüístico se han utilizado las herramientas proporcionadas por *Festival* [2]. Se ha prescindido de la información proporcionada con la base de datos y se ha empleado un alfabeto propio, un silabicador y un conversión grafema-alófono incorporados a *Festival*. El alfabeto utilizado consta de 30 alófonos típicos en castellano, entre los que se incluye el silencio.

Los módulos incorporados a *Festival* para llevar a cabo el análisis lingüístico son:

- Módulo de preproceso y normalización, que trata la pronunciación de nombres propios, acrónimos, números romanos y cifras.
- Módulo conversor grafema-alófono, que a partir de reglas fonéticas extrae la secuencia de alófonos del texto.
- Módulo silabicador, que a partir de la trascripción fonética y basándose en reglas, estima automáticamente la división en sílabas.
- Módulo acentuador, que determina, a partir de reglas, las sílabas tónicas y átonas de la secuencia alofónica.
- Módulo categorizador, que únicamente diferencia del resto el conjunto de palabras función.

A partir del análisis lingüístico se han extraído un conjunto de 65 características lingüísticas. Algunas de las más relevantes son:

- **A nivel de alófono:** Alófono anterior al predecesor, predecesor, actual, posterior, siguiente al posterior, y la posición del alófono actual en la sílaba.
- **A nivel de sílaba:** nº de fonemas y acentuación de la sílaba anterior, actual y posterior; posición de la

sílaba dentro la palabra y del grupo fónico; y la vocal de la sílaba.

- **A nivel de palabra:** la categoría gramatical (*POS*) de la palabra anterior, actual y posterior; nº de sílabas de la palabra anterior, actual y posterior; posición dentro del grupo fónico desde el comienzo y desde el final; y la posición del grupo fónico dentro de la frase.
- **A nivel de grupo fónico:** Nº de sílabas y de palabras del grupo fónico anterior, actual y posterior, y tipo de entonación final.
- **A nivel de frase:** Nº de sílabas, de palabras y de grupos fónicos.

4. SELECCIÓN DE UNIDADES *VERSUS* SÍNTESIS HSMM

En este trabajo se ha implementado una voz basada en selección de unidades y otra basada en Semi-Modelos Ocultos de Markov (HSMM: *Hidden Semi Markov Models*); con el fin de evaluar la bondad de cada técnica aplicada al corpus de la evaluación. Ambas voces han utilizado como módulo de preproceso el explicado en el apartado anterior.

En el caso de la voz basada en selección de unidades se ha utilizado el motor *multisyn*[3] de *Festival*. Durante la generación de esta voz se han encontrado un conjunto de problemas que han dado lugar a las siguientes limitaciones:

- Se ha tenido que prescindir de los párrafos literarios en el entrenamiento de HMM para la segmentación automática del corpus, usando únicamente las frases y los párrafos fonéticamente balanceados.
- A pesar de normalizar la intensidad de los ficheros de audio, se comprobaron variaciones de intensidad en los ejemplos sintetizados.
- Dado el tamaño del corpus, no se dispone de la suficiente cobertura de contextos lingüísticos como para modelar de forma implícita la parte prosódica [3], afectando a la naturalidad de la voz.

La voz basada en HSMM ha sido generada mediante *HTS 2.1* [4]. Algunos de los aspectos que diferencian esta voz de la anterior y que a priori mejoran la calidad de la voz (a falta de una evaluación exhaustiva) son:

- La segmentación fonética es un proceso implícito en el entrenamiento de los HSMM. A diferencia de la segmentación con *multisyn*, en este caso se utiliza información referente a la fuente de excitación (log F0 y componente aperiódica), un mayor número de coeficientes cepstrales y mayor número de estados.

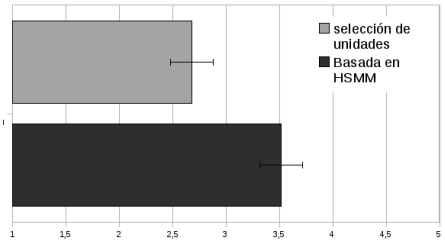


Figura 1. *Evaluación de la calidad de la voz basada en selección de unidades y la basada en HSMM.*

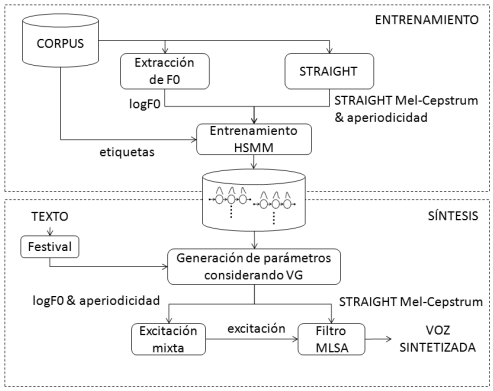


Figura 2. *Descripción del sistema (adaptada de [5]).*

- El uso de un modelo paramétrico proporciona mayor robustez, evitando discontinuidades. A priori, esta técnica proporciona una voz más estable y una síntesis más robusta para este volumen de datos de entrenamiento.

Se ha realizado una breve evaluación de calidad de las voces con objeto de seleccionar la mejor de ambas para la evaluación. 5 oyentes han evaluado 10 textos seleccionados del conjunto de ejemplos de test enviados por la organización de la evaluación, puntuando cada ejemplo siguiendo la escala MOS. Los resultados mostrados en la gráfica 1, indican que la calidad de la voz basada en HSMM (3,52) es mejor que la basada en selección de unidades (2,68).

5. CONVERSIÓN DE TEXTO A VOZ BASADA EN HSMM

En esta sección se describen las características principales del sistema empleado finalmente. Cada uno de los algoritmos empleados se detalla exhaustivamente en [5] y [4]. La Figura 2 presenta un diagrama general del sistema.

5.1. Modelo de producción de voz

Uno de los modelos de producción más extendidos es el conocido como *vocoder*. Este modelo consiste en modelar la voz humana como la convolución de un señal de

excitación con un filtro, el cual representa la información asociada al tracto vocal.

El uso de este modelo limita la calidad de la voz sintetizada, debido a que asume independencia entre la excitación y el filtro dado que simplifica la señal de excitación a un tren de impulsos en los sonidos sonoros, y a ruido en caso de los sonidos sordos. El resultado suele ser la percepción de una voz robótica.

Como solución a este problema, el sistema presentado incorpora STRAIGHT [6], vocoder que mejora la calidad de la síntesis al aplicar un procedimiento adaptativo sobre F0 en la estimación de la envolvente espectral. De esta forma se consigue separar la envolvente espectral de la componente periódica de la señal. Adicionalmente, se estiman medidas de aperiodicidad del espectro, basadas en la relación entre la zona de alta y de baja frecuencia de la envolvente espectral, las cuales representan la distribución relativa de energía de cada componente aperiódica [7].

En el proceso de síntesis, se utiliza un modelo de excitación mixta, basado en la suma de un tren de impulsos con manipulación de la fase y un ruido gausiano. La ponderación de ambas señales se realiza en el dominio de la frecuencia mediante las medidas de aperiodicidad comentadas anteriormente.

5.2. Entrenamiento de los modelos acústicos

Se ha utilizado una frecuencia de muestreo de $16kHz$ y un análisis trama a trama con un enventanado de tipo Blackman de $25ms$ y un desplazamiento de ventana de $5ms$.

Como ya se ha mencionado, en el sistema se han utilizado HSMMs para modelar la envolvente espectral, la información de aperiodicidad y el contorno de F0 (logaritmo de F0 realmente). Con el fin de que los modelos sean entrenables, es necesario codificar la información para disminuir la dimensionalidad de las observaciones. Para ello, a partir de la envolvente espectral se estiman los 40 primeros coeficientes cepstrales (*global mel cepstrum*) y las medidas de aperiodicidad se promedian en 5 subbandas de frecuencia.

Se ha prescindido de la información de las marcas de *pitch* proporcionadas con la base de datos. En nuestro sistema se ha buscado robustecer la estimación del contorno de logaritmo de F0 mediante el empleo de tres tipos de algoritmos de extracción de F0 a partir de la señal de voz. Finalmente, el contorno resultante es el promedio del resultado ofrecido por cada uno de los algoritmos por separado.

Adicionalmente, se calculan la primera y segunda derivada de cada una de las componentes estáticas, formando así un vector de 138 componentes.

En el caso de $logF0$ y sus derivadas se han modelado utilizado distribuciones MSD (*Multi Space Ditribution*) [8], en las que las tramas sonoras se modelan mediante un distribución gausiana con una matriz de covarianzas diagonal, y las tramas sordas mediante una distribución discreta.

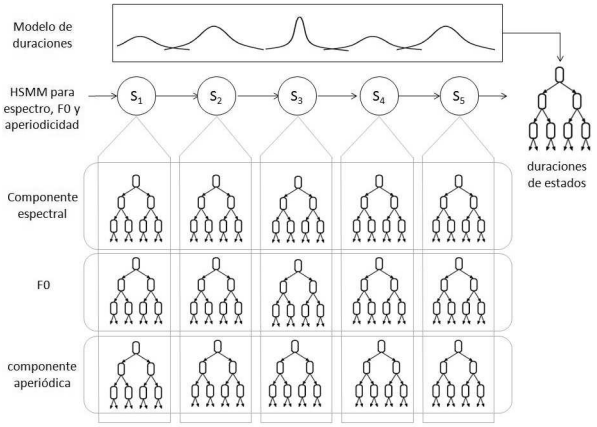


Figura 3. *HSMM dependientes del contexto (adaptada de [11]).*

5.2.1. Empleo de HSMM y modelado de duraciones

Los HSMM modelan la duración de cada estado de forma explícita mediante una función de distribución en lugar de utilizar las probabilidades de transición de los HMM convencionales, lo cual permite modelar el ritmo de una forma más apropiada [9].

En este caso se ha utilizado una función de distribución gausiana multivariable de dimensión equivalente al número de estados (5 en nuestro caso).

5.2.2. Modelos dependientes del contexto

Cada fonema se modela como un HSMM de 5 estados de izquierda a derecha. Para cada estado y cada una de las componentes del modelo (espectro, F0, aperiodicidad y duraciones) se entrenan, de forma independiente pero síncrona [10], un conjunto de modelos dependientes del contexto para cada estado. Éstos se estiman mediante el entrenamiento un árbol de decisión para cada componente aplicando un criterio basado en la *Minimum Description Length* (MDL).

En la generación del árbol de decisión, se ha partido de un conjunto inicial de 2042 preguntas relacionadas con el contexto a nivel fonético (se han utilizado pentafonemas), de sílaba, de palabra o grupo fónico.

El resultado es un conjunto de 63773 modelos para la componente espectral, logF0 y aperiódica y 17556 para el modelado de duraciones. La Figura 3 muestra el conjunto de modelos entrenados.

5.3. Generación de parámetros considerando su varianza global

La generación de secuencias de parámetros se lleva a cabo mediante el algoritmo introducido en [12]. Mediante

la relación entre las características estáticas y dinámicas se generan trayectorias suavizadas de parámetros.

Habitualmente, este suavizado suele ser excesivo, y para evitar esto se incorpora la varianza global de las características como parámetro de optimización junto al de la probabilidad de la observación dada la secuencia de parámetros. En [13] se describe en detalle la consideración de la varianza global en la generación de trayectorias.

5.4. Síntesis de voz

A la hora de sintetizar la señal de voz es necesario estimar la envolvente espectral. Dicha envolvente se aproxima mediante un filtro MLSA (*Mel Log Spectrum Approximation*), con el fin de reducir el coste computacional, estimado a partir de los coeficientes mel-cepstrum. La síntesis se realiza periodo a periodo como la convolución de una fuente de excitación mixta y dicho filtro MLSA [5].

6. CONCLUSIONES

Este trabajo describe la implementación de una voz sintética en castellano basada en HSMM para la *Evaluación Albayzín 2008: Conversión Texto a Voz*. Se han implementado voces basadas en las dos técnicas actuales que compiten en síntesis de voz, selección de unidades y síntesis basada en HSMM. Dichas voces se han implementado usando *Multisyn* de *Festival* y *HTS 2.1* respectivamente. Se ha realizado una evaluación limitada para decidir el mejor sistema para la competición, y finalmente se han descrito las características principales de cada uno de sus módulos. Una demostración de ambos sistemas se puede encontrar on-line en [14].

7. AGRADECIMIENTOS

Los autores agradecen a los miembros de CSTR y GTH su colaboración en la preparación de este trabajo.

8. BIBLIOGRAFÍA

[1] Antonio Bonafonte y Asuncion Moreno, "Documentation of the upc_esma spanish database," *TALP Research Center, Universitat Politecnica de Catalunya, Barcelona*, pp. 2781–2784, 2008.

[2] Paul Taylor, Alan W Black, y Richard Caley, "The architecture of the festival speech synthesis system," in *In The Third ESCA Workshop in Speech Synthesis*, 1998, pp. 147–151.

[3] Robert A. J. Clark, Korin Richmond, y Simon King, "Multisyn: Open-domain unit selection for the festival speech synthesis system," *Speech Communication*, vol. 49, no. 4, pp. 317–330, 2007.

[4] The HTS working group, "Hmm-based speech synthesis system (hts). http://hts.sp.nitech.ac.jp," Último acceso: septiembre de 2008.

[5] H. Zen, T. Toda, M.Ñakamura, y K. Tokuda, "Details of nitech hmm-based speech synthesis system for the blizzard challenge 2005," *IEICE Trans. Inf. & Syst*, vol. E90-D, no. 1, pp. 325–333, January 2007.

[6] Hideki Kawahara, Haruhiro Katayose, Alain de Cheveign´e, y Roy D. Patterson, "Fixed point analysis of frequency to instantaneous frequency mapping for accurate estimation of f0 and periodicity," *In Proc. of Eurospeech*, pp. 2781–2784, 1999.

[7] Hideki Kawahara, Jo Still, y Osama Fujimura, "Aperiodicity extraction and control using mixed mode excitation and group delay manipulation for a high quality speech analysis, modification and syntesis system straight," *Proc MAVEBA*, pp. 13–15, September 2001.

[8] K. Tokuda, T. Masuko, N. Miyazaki, y T. Kobayashi, "Multi-space probability distribution hmm," *IEICE Trans. Inf. & Syst.*, vol. E85-D, no. 3, pp. 455–464, March 2002.

[9] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, y T. Kitamura, "Hidden semi-markov model based speech synthesis," *In Proc. of ICSLP*, vol. II, pp. 1397–1400, October 2004.

[10] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, y T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," *In Proc. of Eurospeech*, pp. 2347–2350, September 1999.

[11] Keiichi Tokuda, Heiga Zen, y Alan W. Black, "An hmm-based speech synthesis system applied to english," *Proc. of IEEE SSW*, vol. E90-D, no. 5, pp. 806–824, September 2002.

[12] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, y T. Kitamura, "Speech parameter generation algorithms for hmm-based speech synthesis," *In Proc. of ICASSP*, pp. 1315–1318, June 2000.

[13] T. Toda y K. Tokuda, "A speech parameter generation algorithm considering global variance for hmm-based speech synthesis," *IEICE Transactions*, vol. E90-D, no. 5, pp. 806–824, May 2007.

[14] R. Barra-Chicote et al., "Madrid-bsdm. http://lorien.die.upm.es/ barra/sintesis-albayzin08," Último acceso: septiembre de 2008.

PHRASE SEGMENTS OBTAINED WITH STOCHASTIC INVERSION TRANSDUCTION GRAMMARS FOR SPANISH-BASQUE TRANSLATION

Germán Sanchis-Trilles, Joan Andreu Sánchez,

Instituto Tecnológico de Informática
Universidad Politécnica de Informática
Camino de Vera, s/n. 46022 Valencia, Spain
{gsanchis,jandreu}@dsic.upv.es

ABSTRACT

One of the weaknesses of the so-called phrase-based translation models is that they carry out a blind extraction of the phrase translation table, i.e., they do not take into account the possible linguistic restrictions that each language introduces because of its own syntax. In this work, we use Stochastic Inversion Transduction Grammars as a phrase extraction technique which is able to yield similar results to more popular, but heuristic, techniques. We present encouraging results obtained on the Albayzin 2008 corpus.

1. INTRODUCTION

The grounds of modern Statistical Machine Translation (SMT), a pattern recognition approach to Machine Translation, were established in [1], where the problem of machine translation was defined as following: given a sentence $\mathbf{x}$ from a certain source language, an adequate sentence $\hat{\mathbf{y}}$ that maximises the posterior probability is to be found. Such a statement can be specified with the following formula

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, Pr(\mathbf{y}|\mathbf{x}). \tag{1}$$

Applying the Bayes theorem on this definition and operating appropriately, one can easily obtain the following formula

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}}\, Pr(\mathbf{y}) \cdot Pr(\mathbf{x}|\mathbf{y}), \tag{2}$$

where $Pr(\mathbf{y}|\mathbf{x})$ has been decomposed into two different probabilities: the *statistical language model* of the target language $Pr(\mathbf{y})$ and *the (inverse) translation model* $Pr(\mathbf{x}|\mathbf{y})$, and the denominator has been neglected because it does not affect the maximisation.

In practise, the direct modelling of the posterior probability $Pr(\mathbf{y}|\mathbf{x})$ has been widely adopted. To this purpose, different authors [2, 3] propose the use of the so-called log-linear models, where the decision rule is given by the expression

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmax}} \sum_{m=1}^{M} \lambda_m h_m(\mathbf{x}, \mathbf{y}) \tag{3}$$

where $h_m(\mathbf{x}, \mathbf{y})$ is a score function representing an important feature for the translation of $\mathbf{x}$ into $\mathbf{y}$, M is the number of models (or features) and λ_m are the weights of the log-linear combination.

2. PHRASE-BASED MODELS

The derivation of the Phrase-Based (PB) models stems from the concept of bilingual segmentation, i.e. sequences of source words and sequences of target words. It is assumed that only segments of contiguous words are considered, the number of source segments being equal to the number of target segments and each source segment being aligned with only one target segment and vice versa.

An important issue when training PB models is the algorithm by means of which the bilingual phrases are extracted. Hence, a wide variety of methods have been proposed for this purpose, spanning through statistically motivated procedures [4], heuristic algorithms [5], and linguistically motivated methods [6]. In this work, we will be following this last approach, which relies on Stochastic Inverse Transduction Grammars (SITGs) [7] for phrase extraction.

In this work we will be following the approach by [8], in which SITGs are used for phrase extraction, reporting preliminary results on the EuroParl corpus. In [9], such work was extended with a more thorough experimentation, improving considerably the translation quality previously obtained.

This work has been partially supported by the Spanish MEC under scholarship AP2005-4023 and under grants CONSOLIDER Ingenio-2010 CSD2007-00018, by the EC (FEDER) and the Spanish MEC under grant TIN2006-15694-CO2-01, and by the Generalitat Valenciana under grant GVPRE/2008/331, research project "Traducción Automótica del Corpus UPenn Treebank mediante Tócnicas Interactivas (UPennSpanish)."

3. STOCHASTIC INVERSION TRANSDUCTION GRAMMARS

Being closely related to stochastic context free grammars, Stochastic Inverse Transduction Grammars [7] specify a subset of stochastic syntax-directed stochastic grammars. Analysing two strings simultaneously, SITGs may be used to extract bilingual segments from a parallel corpus while taking into account syntax-motivated restrictions. The internal nodes of the parse tree define a span over each pair of strings. These spans can be considered as paired segments of words.

In [7], an algorithm similar to the CYK algorithm for context free grammars is proposed in order to parse a sentence pair with a SITG. This algorithm has a time complexity of $O(|x|^3|y|^3|R|)$, being $|x|$ the length of the source sentence, $|y|$ the length of the target target sentence, and $|R|$ the number of rules in the SITG. However, if the input part of the corpus (the source language), the output part (the target language) or both of them has been previously parsed (each part with a monolingual parser) and is given in a bracketed form, [6] suggests the use of a version of the algorithm given in [7] which is more efficient while performing the analysis, achieving a time complexity of $O(|x||y||R|)$ when x and y are fully bracketed. In this work, we will be taking profit of bracketing information provided by freely available monolingual parsing toolkits in order to achieve an important increase of speed within the estimation algorithm, without a significant loss in terms of final translation quality [9].

4. SITGS FOR PHRASE EXTRACTION

First, we built an initial SITG by following the method described in [8]. The basic idea is to construct the maximum number of syntactic rules with a given number of non-terminal symbols. These non-terminal symbols were not syntacticly motivated. The lexical rules of the initial SITG were obtained from a lexical dictionary. Then, the source language in the training corpus (Spanish) was bracketed by using FreeLing [10], which is an open-source suite of language analysers. This being done, we then used the bracketed corpus to perform two stochastic estimation iterations on the initial SITG and obtain improved SITGs. Finally, the SITG obtained after the estimation iterations was used to parse the bracketed training corpus and extract segment pairs to setup a phrase-based translation model.

Once extracted, the phrase pairs were scored according to the following translation models:

1. Following common knowledge in SMT, we computed both the inverse and direct translation probabilities of each segment pair according to the formulae

$$p(\mathbf{s}|\mathbf{t}) = \frac{C(\mathbf{s},\mathbf{t})}{C(\mathbf{t})} \qquad p(\mathbf{t}|\mathbf{s}) = \frac{C(\mathbf{s},\mathbf{t})}{C(\mathbf{s})}$$

where $C(\mathbf{s},\mathbf{t})$ is the number of times segments **s** and **t** were extracted throughout the whole corpus.

2. We also scored the phrase pairs with syntax-based translation models. These are obtained following the technique described in [9], where each segment pair is assigned a probability according to the corresponding SITG. When a given segment pair $(\mathbf{s},\mathbf{t})$ is parsed by the SITG, a joint probability $\widehat{p}(\mathbf{s},\mathbf{t})$ is obtained. Since this probability may differ depending on the parse tree it comes from, we need to normalise accordingly. Let Ω the multiset of spans (word segments) obtained from the training sample, and $\Omega_{\mathbf{s},\mathbf{t}} \subseteq \Omega$ the multiset of $(\mathbf{s},\mathbf{t})$ spans. The expected value of $\widehat{p}(\mathbf{s},\mathbf{t})$ is defined according to the empirical distribution as:

$$E_\Omega(\widehat{p}(\mathbf{s},\mathbf{t})) = \frac{\sum_{(\mathbf{a},\mathbf{b})\in\Omega_{\mathbf{s},\mathbf{t}}} \widehat{p}(\mathbf{a},\mathbf{b})}{|\Omega|}.$$

Similarly,

$$p(\mathbf{s}|\mathbf{t}) = \frac{E_\Omega(\widehat{p}(\mathbf{s},\mathbf{t}))}{E_\Omega(\widehat{p}(\mathbf{t}))}, \quad p(\mathbf{t}|\mathbf{s}) = \frac{E_\Omega(\widehat{p}(\mathbf{s},\mathbf{t}))}{E_\Omega(\widehat{p}(\mathbf{s}))}.$$

3. In addition, we also considered the use of lexical weights, as described in [5]. These lexical weights attempt to account for the lexical soundness of each phrase pair, estimating how well each of the words in one language translates to each of the words in the other language.

With these scores, we build three sets of phrase-tables. The first one was built by only including the direct and inverse translation probabilities (1) and the syntactic probabilities (2), since this was the combination reported in [9]. This combination will be referred as *Vsyn*. However, in [9], lexical weights were not included. For this reason, in these experiments we analysed the effect of only including direct and inverse translation probabilities and lexical weights (this combination will be referred to as Vlex), and including all six sets of probabilities (from now on, VII). These phrase-tables were fed to Moses [11] for producing the final translation.

5. EXPERIMENTS

We performed our experiments on the Spanish-Basque Albayzin corpus, with the partition established in the *V Jornadas en Tecnología del Habla* (2008). The statistics of the corpus can be seen on Table 1. As it can be seen on the Table, translating both from or into Basque is a difficult task, since the amount of Out of Vocabulary words quickly becomes very high.

As Table 2 shows, the translation quality tends to get better when increasing number of non-terminal symbols are used, as measured by BLEU. Moreover, the VII combination, in which all translation models are used, seems

Table 1. Characteristics of Albayzin corpus. OoV stands for "Out of Vocabulary" words, Dev. for Development, K for thousands of elements and M for millions of elements.

		Spanish	Basque
Training	Sentences	58K	
	Run. words	1151K	885M
	Avg. length	19.8	15.2
	Voc.	49.4K	87.8K
Dev.	Sentences	1456	
	Run. words	29K	23K
	Avg. length	20.1	15.5
	OoV	489	8376
Test	Sentences	1446	
	Run. words	28K	22K
	Avg. length	19.3	14.9
	OoV	483	8096

Table 2. Translation results for Spanish-Basque translation when using a SITG with only one, three and five non-terminal symbols

non terms	combination	BLEU	WER	TER
1	Vsyn	8.8	82.0	78.5
	Vlex	8.8	81.8	78.2
	VII	9.0	81.7	78.1
3	Vsyn	8.9	81.9	78.6
	Vlex	8.9	81.8	78.3
	VII	9.1	**81.4**	**77.9**
5	Vsyn	9.1	82.2	78.7
	Vlex	9.2	81.5	78.9
	VII	**9.3**	81.6	78.1

to yield improvements over the other alternatives, as measured by BLEU, WER and TER. However, it must be noted that these differences are not statistically significant. The results shown in this table were obtained restricting the decoder to perform a monotonic translation procedure, since at this stage we have not yet implemented a SITG-based reordering model. In this case, the language model used was a 5-gram, applying interpolation with Knesser-Ney discount.

For comparison purposes, the best scores obtained by the Moses toolkit in its monotonic setup are 9.4 BLEU, 81.7 WER and 78.3 TER, which are not significantly better than the scores obtained by our system trained with 5 non-terminal symbols in the VII combination.

6. CONCLUSIONS AND FUTURE WORK

We have presented an alternative method for phrase extraction, which is competitive in terms of quality. This method obtains phrase segments from paired sentences by parsing both of them in a completely unlexicalixed manner.

In the future, we plan to compute more complex SITGs and introduce further models to improve our translation table, such as the lexical alignment models or other models obtained by combining the various probabilities that SITG estimation entails. In this line, we also plan to investigate which effect has the combination of our phrase table with the phrase table produced by Moses.

7. BIBLIOGRAFA

[1] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, y Robert L. Mercer, "The mathematics of machine translation," in *Computational Linguistics*, June 1993, vol. 19, pp. 263–311.

[2] K. Papineni, S. Roukos, y T. Ward, "Maximum likelihood and discriminative training of direct translation models," in *Proc. of ICASSP'98*, 1998, pp. 189–192.

[3] F. Och y H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proc. of the ACL'02*, 2002, pp. 295–302.

[4] J. Tomas y F. Casacuberta, "Monotone statistical translation using word groups," in *Proceedings of the Machine Translation Summit VIII*, Santiago de Compostela, Spain, 2001, pp. 357–361.

[5] R. Zens, F.J. Och, y H. Ney, "Phrase-based statistical machine translation," in *Advances in artificial intelligence. 25. Annual German Conference on AI. Lecture Notes in Computer Science*, 2002, vol. 2479, pp. 18–32.

[6] J.A. Sánchez y J.M. Benedí, "Obtaining word phrases with stochastic inversion transduction grammars for phrase-based statistical machine translation," in *Proc. 11th Annual conference of the European Association for Machine Translation*, Oslo, Norway, June 2006, pp. 179–186.

[7] D. Wu, "Stochastic iversion transduction grammars and bilingual parsing of parallel corpora," *Computational Linguistics*, vol. 23, no. 3, pp. 377–403, 1997.

[8] J.A. Sánchez y J.M. Benedí, "Stochastic inversion transduction grammars for obtaining word phrases for phrase-based statistical machine translation," in *Proceedings of the Workshop on Statistical Machine Translation*, New York City, 2006, pp. 130–133.

[9] G. Sanchis-Trilles y J.A. Sánchez, "Using parsed corpora for estimating stochastic inversion transduction grammars," in *6th edition of the International Conference on Language Resources and Evaluation*, Marrakech, Morocco, May 26 – June 1 2008.

[10] J. Asterias, B. Casas, E. Comelles, M. González, L. Padró, y M. Padró, "Freeling 1.3: Syntactic and semantic services in an open-source nlp library," in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy, 2006.

[11] Alexandra Birch Chris Callison-Burch Marcello Federico Nicola Bertoldi Brooke Cowan Wade Shen Christine Moran Richard Zens Chris Dyer Ondrej Bojar Alexandra Constantin Evan Herbst Philipp Koehn, Hieu Hoang, "Moses: Open source toolkit for statistical machine translation," in *ACL 2007, demonstration session*, 2007.

THE AVIVAVOZ PHRASE-BASED STATISTICAL MACHINE TRANSLATION SYSTEM FOR ALBAYZIN 2008

Carlos A. Henríquez Q.[1], *Maxim Khalilov*[1],
José B. Mariño[1], *Nerea Ezeiza*[2]

[1]Department of Signal Theory and Communications
TALP Research Center (UPC)
Barcelona 08034, Spain
{carloshq|khalilov|canton}@gps.tsc.upc.edu

[2]Department of Language and Computer Systems
University of the Basque Country
n.ezeiza@ehu.es

ABSTRACT

This paper describe the SMT system developed by the TALP Research Center of the UPC (Universitat Politècnica de Catalunya) for the Albayzin 2008 evaluation campaign (Spanish to Basque translation task). Apart from a standard set of feature models, the system introduces two target language models: one based on lemmas and the second based on linguistic classes (Part-of-Speech). The word-to-word alignment was obtained using a segmented version of the Basque corpus. The results obtained over the development and test sets are analyzed and discussed.

1. INTRODUCTION

Nowadays, the available bilingual material for automatic translation between Spanish and Basque is limited. Aiming to overcome this issue, we present our first approach to phrase-based Statistical Machine Translation (SMT) for this pair of languages trying to reach acceptable translation quality under conditions of smaller training material. The developed SMT system uses a POS language model and a lemmatize language model for Basque in order to improve the translations obtained by a baseline system. System configuration is discussed and the results obtained with the provided parallel corpus are presented.

This paper is organized as follow. Section 2 gives a brief description of the phrase-based model that the system is based on. Section 3 describes the corpus statistics, alignment procedure, features models and the case restoration method. Section 4 reports the main results of our system performance during development and testing. Finally section 5 sums up the main conclusions from our institution's participation in the evaluation.

This work has been funded by the Spanish Government under grant TEC2006-13694-C03 (AVIVAVOZ project).

2. PHRASE-BASED SMT SYSTEM

The phrase-based translation system[4] implements a log-linear model in which a foreign language sentence $f^J = f_1, f_2, \ldots, f_J$ is translated into another language sentence $e^I = e_1, e_2, \ldots, e_I$ by searching for the translation hypothesis $\hat{e}^I$ maximizing a log-linear combination of several feature models[2]:

$$\hat{e}_I = \arg\max_{e^I}\{\sum_{m=1}^{M} \lambda_m h_m(e^I, f^J)\}$$

Where the feature function h_m refers to the system models and λ_m refers to the corresponding optimized model weights.

The main system models are the translation model and the language model. The first one deals with the issue of which target language phrase f_j translates a source language phrase e_i and the latter model estimates the probability of translation hypothesis. Apart from these two models, a set of additional models was used in the AVIVAVOZ system. They are presented in section 3.3.

The development of the AVIVAVOZ SMT system is based on the MOSES toolkit[3].

3. ALBAYZIN 2008 EVALUATION FRAMEWORK

3.1. Corpus

For the system design, the corpus used was the one provided for the evaluation campaign. It is a set of 61104 sentences, divided in three subsets: a training corpus containing 58202 sentences, a development corpus of 1456 sentences and a test corpus with the remaining 1446 sentences. Only one reference for each set was supplied. The basic corpus statistics can be found in table 1.

The tokenization, Part-of-Speech (POS) tags and lemmatization for both languages was also provided altough only the Basque information was used during development. According to the campaign documentation, the POS tags for Basque were obtained with the Eustagger[1] tool and only the category and subcategory of each tag were provided.

An additional preprocessing consisted in changing the text encoding from ISO88591 to UTF8, removing all the sentences bigger than 100 words and lowercasing the entire corpus. The removing step was performed only over the train corpus due to a restriction implied by the alignment tool. The removed set was smaller than 1% of the training corpus.

3.2. Alignment

Altough the baseline system used the lowercased corpus to obtain the word-to-word alignment, the final system used a different approach.

A segmentation tool was developed which splitted the Basque words using the POS information and a suffix dictionary; wherever a verb, an adjective or a name was found, the word was checked with the dictionary, and if the word ended with any of the listed suffix, it was splitted in two e.g. "publikoen", which is a Basque adjective, ended with the suffix "en" (listed in the dictionary), therefore it was splitted into "publiko+ +en".

With the segmentation tool, the Basque lowercase corpus was segmented and the alignment was computed on this corpus. Once the alignment was completed, and using a developed desegmentation tool which joins the words with their splitted suffixes, the corpus was desegmented to its original version and the links were properly relocated to the original words.

The Spanish lowercased corpus remained the same during the process. The alignment was automatically computed by the GIZA++[5] toolkit.

3.3. Features

The SMT system developed uses the following feature functions during translation:

- Phrase translation probability on both directions, based on a joint probability model[4].
- Lexical weighting on both directions, based on word-to-word IBM Model 1 probabilities[6].
- Phrase penalty features which compensate the system's preference for short output sentences.
- Target language model of order 5.
- POS target language model of order 7.
- Lemma target language model of order 7.

	Baseline	POS LM	Seg. Align	Lemma LM
BLEU	12.66	12.76	13.01	13.26
NIST	4.77	4.72	4.80	4.90
WER	77.90	78.38	78.61	77.61
PER	60.15	60.61	60.36	59.37

Table 2. Results obtained on the development set

	Baseline	POS LM	Seg. Align	Lemma LM
BLEU	11.43	11.68	11.89	11.95
NIST	4.68	4.66	4.65	4.74
WER	78.74	78.71	79.11	78.50
PER	60.38	60.64	60.65	59.93

Table 3. Results obtained on the test set

Beeing the first four the features used in our baseline system. The reordering model is based on lexicalized reordering[8] in all the performed experiments. This distortion model takes into account the relative movement between a given phrase and its adjacent phrases.

3.4. Case Restoration

Because all the design used a lowercased corpus, a final case-restoration tool is needed to stablish a truecase translation. For this matter, the *disambig* and *ngram-count* tools from the SRI Language Model toolkit[7] were used.

4. EXPERIMENTS AND RESULTS

For the Spanish-Basque task, four different systems were developed in a progressive fashion. The first one, called the *baseline*, has the default feature functions and parameters of MOSES. From that starting point, a *POS target language model* was add to the SMT system. In the third system we performed a modified *alignment*, which consisted in performing the alignment with a Basque segmented corpus as commented in section 3.2.

The final system also included a *target language model based on lemmas*. As mentioned in section 3.1, the lemmas for Basque were computed automatically and were provided by the organizers of the evaluation campaign. Both language models (of the POSs and the lemmas) were 7-gram models and the order of the surface words language model was 5. The maximum phrase size was set to 5 for all the systems.

Table 2 and 3 show the different results obtained with the systems developed. Each column corresponds to a different system, starting with the baseline and ending with the system that was submitted to the evaluation. It can be seen that the addition of the different features resulted in final improvement of 0.5% BLEU points over the test set and 0.6% over de development set.

	Train Corpus		Devel. Corpus		Test Corpus	
	Spanish	Basque	Spanish	Basque	Spanish	Basque
Number of sent	58202		1456		1446	
Max. sent size	242	236	93	76	317	232
Avg. sent size	19.77	15.20	23.33	18.70	22.32	17.85
Vocab size	97558	140931	7170	9031	6926	8691

Table 1. Basic corpus statistics

5. CONCLUSIONS

In this paper we introduced the AVIVAVOZ phrase-based SMT system which partipated in the Albayzin 2008 evaluation campaign. Starting with a brief introduction to the phrase-based statistical translation modelling, the corpus and preprocessing description were presented. Further, we described the set of feature models that were taken into account for the design of the system and the results obtained with different systems configurations.

The main conclusion which can be drawn from the results is that despite the additional features were useful, a different approach better dealing with global word reordering and the agglomerative characteristic of Basque is needed to obtain an improved system performance.

6. REFERENCES

[1] Itziar Aduriz, Nerea Ezeiza, and Ruben Urizar. Euslem: A lemmatiser/tagger for basque. pages 17–26, Göteborg. Göteborg University, Department of English, 1996.

[2] Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics.*, 16(2):79–85, 1990.

[3] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 177–180, 2007.

[4] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *HLT-NAACL*, pages 48–54, 2003.

[5] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[6] Franz Josef Och and Hermann Ney. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, 2004.

[7] A. Stolcke. Srilm - an extensible language modeling toolkit. In *International Conference on Spoken Language Processing*.

[8] Christoph Tillman. A block orientation model for statistical machine translation. In *HLT-NAACL*, 2004.

THE CEREVOICE SPEECH SYNTHESISER

Juan María Garrido[1], Eva Bofias[1], Yesika Laplaza[1], Montserrat Marquina[1]
Matthew Aylett[2], Chris Pidcock[2]

[1]Barcelona Media Centre d'Innovació, Barcelona, Spain

[2]Cereproc Ltd, Edinburgh, Great Britain

ABSTRACT

This paper describes the CereVoice® text-to-speech system developed by Cereproc Ltd, and its use for the generation of the test sentences for the Albayzin 2008 TTS evaluation. Also, the building procedure of a Cerevoice-compatible voice for the Albayzin 2008 evaluation using the provided database and the Cerevoice VCK, a Cereproc tool for fast and fully automated creation of voices, is described.

1. INTRODUCTION

CereVoice® is a unit selection speech synthesis software development kit (SDK) produced by CereProc Ltd., a company based in Edinburgh and founded in late 2005 with a focus on creating characterful synthesis and massively increasing the efficiency of unit selection voice creation [1, 2, 3, 4, 5].

Cereproc Ltd and *Barcelona Media Centre d'Innovació* (BM) started in 2006 a collaboration which led to the development of two text normalization modules, for Spanish and Catalan, the lexicon and the letter-to-sound rules for transcription in both languages, and a Spanish-Catalan bilingual voice ('mar') for the CereVoice® system. As a result of this collaboration, BM became an official 'voice developer' of Catalan and Spanish voices for CereVoice®, and got an academic license to use the CereVoice® system for research purposes.

In this paper a brief description of the CereVoice® TTS-system engine is given, and the procedures of voice building and speech files generation for the Albayzin 2008 evaluation, carried out by BM with the support of Cereproc, are described.

2. GENERATING SPEECH USING CEREVOICE

To generate speech using the CereVoice® system, three components are necessary: the Cerevoice engine (Cerevoice SDK), a text normalization module (the one provided by Cereproc or any other compatible with Cerevoice), and a TTS Voice. Also, some optional modules, such as user lexicons or user abbreviations tables, can be used to improve the text processing in particular applications. Figure 1 shows a workflow scheme of the system.

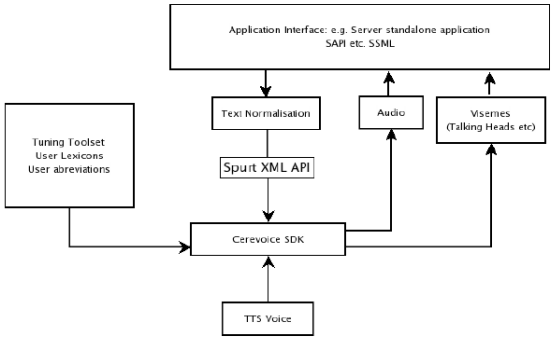


Figure 1. *Overview of the architecture of the Cerevoice synthesis system. A key element in the architecture is the separation of text normalization from the selection part of the system and the use of an XML API.*

In the following subsections, a brief description of the main features of the Cerevoice engine, the text processing module and the voices is given.

2.1. Cerevoice engine

Cerevoice is a new faster-than-realtime speech synthesis engine, available for academic and commercial use. The system is designed with an open architecture, has a footprint of approximately 70Mb for a 16Khz voice and runs at approximately 10 channels realtime. The core Cerevoice engine is an enhanced synthesis 'back end', written in C for portability to a variety of platforms. To simplify the creation of applications based on Cerevoice, the core engine is wrapped in higher level languages such as Python using Swig.

The Cerevoice core engine is a diphone based unit selection system with pre-pruning and a Viterbi search for selecting candidates from the database. The system uses both symbolic (e.g. stress, break index information) and parametric target cost functions (e.g. F0 and duration). Transition costs are based on Line Spectral

Frequencies, F0 smoothed over voiced plus unvoiced speech, and energy. Target and transition weights can be set manually for fine tuning of each particular voice.

The engine does not fit the classical definition of a synthesis back end, as it includes lexicon lookup and letter-to-sound rule modules. The Spanish and Catalan versions of these lexica and rules were jointly developed by BM and Cereproc.

An XML API defines the input to the engine. The API is based on the principle of a 'spurt' of speech. A spurt is defined as a portion of speech between two pauses. The XML API defines also the set of tags that the engine is able to interpret to control several speech output parameters, such as tempo, global tone, genre, or even the language.

One of the aspects that can be controlled through the insertion of tags in the input text, and that can be used to improve the quality of the output speech, is the use of *variants*. In Cerevoice it is possible to ask the engine to prune out a section of the best path found during the Viterbi search and to rerun the Viterbi over that section to find a less optimal alternative or variant. Inside an XML spurt, a word can enclosed by a 'usel' tag containing a variant attribute to force this behaviour. For example <usel variant='0'> is equivalent to no tag, and <usel variant='6'> would be the sixth alternative according to the Viterbi search.

2.2. Text normalization module

The Cerevoice engine is agnostic about the 'front end' used to generate spurt XML. However, the Cerevoice system includes its own modular Python system for text normalization in several languages. Those modules include a set of normalization rules for processing hours, dates, telephone numbers, figures, abbreviations, letters and any other element that needs to be expanded.

The Spanish and Catalan modules provided within the Cerevoice system where jointly developed by Cereproc Ltd and BM. The Spanish text processing module has been used for the generation of the test sentences for the evaluation.

2.3. Cereproc Voices

TTS voices for the Cerevoice SDK are currently available in English (Scottish and British), Catalan, Spanish and Japanese. The Spanish and Catalan voices are the result of a collaboration between BM and Cereproc Ltd.

Voice building is carried out using the Cerevoice Voice Creation Kit (VCK), a tool designed and tuned for fast development of new voices. Using this tool, voice building is a heavily automated, modular, dependency-driven process, consisting of two main types of component: speech parameterization and segmentation. Speech data, text transcriptions, and a lexicon are the only required inputs to voice building.

Voice building includes often a second stage of voice tuning. Voice tuning consists of a manual adjustment of the weights for the target and transition cost functions used for unit selection during the synthesis process. This process involves iterative trial-and error modification of the weights in order to set the optimal combination.

3. THE CEREVOICE VOICE CREATION KIT

The Cerevoice VCK is a tool for fast voice building with minimal manual intervention of the voice developer. Only the speech data, in RIFF wav files, the corresponding orthographic transcription of each utterance, as a single UTF-8 text file, and a lexicon of the language are needed as input. It runs on Linux machines and requires Python for running.

CereProc recommends 15 hours of data for a general purpose voice, although around 5-6 hours of audio data should be enough for an acceptable voice. Voices can also be built incrementally, adding data gradually, until the required quality is achieved.

Cerevoice VCK expects the orthographic text to be provided in the form of a recording script, a text file containing a text line for each wav file, and an identifier preceding the text. If CereProc's Voice Recorder software is used for the recordings, a valid recording script is also needed to display the texts to be read by the speaker, and to name and store automatically the corresponding wav files. If speech data coming from an external source are used for voice building, orthographic transcriptions have to be merged into a single script, in the format required by the VCK.

Examples of valid script lines are:

```
v0001_001 Acme Limited is the best company
in the wold.
v0001_002 Acme Limited (company code A C
M) made one point five billion dollars
profit last year.
```

Finally, a valid lexicon is also needed for the building process. CereProc provides a lexicon for every supported languages and accents. If additional sentences have been added to the VCK recording script, it may be necessary to add words to the pronunciation lexicon. During voice building, sentences are excluded from the voice if a word does not exist in the lexicon.

The building process needs also some additional information about the location of the script and speech files, and other related information, which has to be set in an XML configuration file. The creation of such a configuration file is also one of the tasks needed to build a new voice.

Voice building consists of two steps: segmentation and parameterization. In the first one, segmentation is carried out using the HTK Hidden Markov Model toolkit in forced alignment mode. In the second one, F0

and pitch mark parameters are generated using the ESPS tools 'epochs' and 'get f0'. Edinburgh Speech Tools' 'sig2fv' is used to generate cepstral parameters, which are used to generate Line Spectral Frequencies.

During the segmentation of the speech, bad data may be thrown out of the build. This may happen if there are noisy or truncated audio files, if the speech does not match the text in the script, files where the lexicon pronunciation for a word is incorrect, or for files where the speaker has mispronounced a word. VCK provides several ways of recovering and inspecting the discarded files, in order to fix input problems.

A speech GUI is also provided to allow the developer to find data errors such as lexicon problems and mismatches between audio files and text while running the voice.

4. BUILDING THE ALBAYZIN EVALUATION VOICE

The Albayzin 2008 TTS evaluation has been considered at BM, first of all, as an excellent chance to compare Cerevoice with other systems. But it has been seen also as a way to test the capabilities of the Cerevoice VCK to build voices from external data in a fast way, and with a minimum of manipulation of the input data. And finally, considering that the database used for this evaluation contains only two hours of speech, the evaluation has been used to test the behavior of the Cerevoice engine with voices build from a small amount of speech data (the usual amount of speech used to build commercial voices is about 5-6 hours).

According to these considerations, the building procedure of the voice involved the usual steps when creating a voice with the Cerevoice VCK from external data:

a. preparation of the script file;
b. preparation of the wav files (renaming the files to have a valid name);
c. preparation of the configuration file;
d. running of the VCK;
e. checking and fixing of errors.

Creating the script file was a straightforward task, which involved deletion of the pause marks in the files, merging all the text files into a single file, adding a VCK-compatible identifier to each line, and translating the resulting file to UTF-8 format.

Preparing the wav files was also fast, and involved renaming the files with the corresponding VCK-compatible identifier (the same as the corresponding orthographic text line in the script file), and preprocessing of the file to perform a peak normalization of the files. This task was carried out using an audio processing tool provided by Cereproc as part of the VCK. We decided to include the whole set of sentences, although they came from three different 'styles' (paragraphs, sentences and literary) due to the small global size of the database. However, we decided to identify them as different genres, Questions were not considered a different genre, as it was the case in the 'mar' voice building procedure, because it was not possible to take advantage of this differentiation for the generation of the sentences (it was not allowed to use genre tags).

Finally, a configuration file was also prepared. To do this, a signature file was provided by Cereproc. This is a necessary step for the creation of any new voice.

Once all the necessary material was ready for processing, VCK was launched for voice building. This process was really fast: about half an hour in an Intel-based server. No special problems arose during the building procedure, so it was not necessary to rebuild the voice several times.

No special tuning of the weights was carried out for the building of the voice. The ones used are those established for the building of the 'mar' Spanish voice.

5. GENERATING THE TEST SENTENCES

Test sentences were generated using the latest available version of the Cerevoice SDK (2.1.0). The text normalization module developed by Cereproc and BM for Spanish was used for text processing. No modification of the input text files was carried out during the generation process, as established in the evaluation requirements, apart from their conversion to valid XML files in UTF-8, to allow Cerevoice to process them. This condition excluded the insertion of variant tags in the text files.

6. CONCLUSIONS

In this paper, the CereVoice® TTS system and the Cerevoice Voice Creation Kit have been presented. Also, the building procedure of the evaluation voice and the generation of the test sentences have been described. The building procedure has shown the capabilities of the VCK to create voices for CereVoice® with minimal effort and manual intervention. The results of the evaluation will show to what extent the CereVoice® system is able to generate high-quality synthetic speech when the amount of data available for voice building is smaller than usual.

7. REFERENCES

[1] M.P. Aylett and Yamagishi, J., "Combining Statistical Parametric Speech Synthesis and Unit-Selection for Automatic Voice Cloning". LangTech 2008, Rome.

[2] M.P. Aylett and C.P. Pidcock, "The CereVoice Characterful Speech Synthesiser SDK (Industrial Demo)". IVA 2007, Paris, France, Proceedings. Lecture Notes in Computer Science 4722 Springer.

[3] M.P. Aylett, J.S. Andersson, L. Badino and C.J. Pidcock, “The Cerevoice Blizzard Entry 2007: Are Small Database Errors Worse than Compression Artifacts?”, Blizzard Challenge Workshop, Bonn, 2007.

[4] M.P. Aylett and C.P. Pidcock, “The CereVoice Characterful Speech Synthesiser SDK”, AISB 2007, Newcastle. pp.174-8

[5] M.P. Aylett, C.P. Pidcock and M.E. Fraser, “The CereVoice Blizzard Entry 2006: A prototype Database Unit Selection Engine”, Blizzard Challenge Workshop, Pittsburgh, 2006.

THE UPC TTS SYSTEM DESCRIPTION

Antonio Bonafonte[1]*, Pablo Daniel Agüero*[2]

TALP Research Center, Universitat Politècnica de Catalunya, Spain [1]
Communications Lab, University of Mar del Plata, Argentina [2]

ABSTRACT

This paper presents the UPC TTS system named Ogmios. Ogmios is a system based on unit-selection using acoustic and phonetic features both in target and concatenation costs.

Most of the modules of Ogmios rely on data driven techniques. It has been an useful approach to generate voices in many languages, such as Spanish, Catalan, UK English, and Mandarin Chinese.

1. INTRODUCTION

This paper describes Ogmios, the UPC Text-to-Speech system used for the evaluation. The system was originally designed for Spanish and Catalan but has been extended to English and Mandarin [1, 2]. This paper is organised as follows: Section 2 describes the system and Section 3 explains the process of building the voices.

2. SYSTEM DESCRIPTION

2.1. Text and Phonetic Analysis

The first task of the system is to detect the structure of the document and to transform the input text into words. For this task we have used rules for tokenizing and classifying *non-standard words* in Spanish. The rules for expanding each token into *words* are language dependent, but are based in a few simple functions (spellings, natural numbers, dates, etc.) by means of regular expressions.

The second process is the POS tagger. Ogmios includes a statistical tagger based on FreeLing. The FreeLing package consists of a library providing language analysis services. Main services used of FreeLing library are PoS tagging and probabilistic prediction of unknown word categories. Freeling provides services for all currently supported languages: Spanish, Catalan, Galician, Italian, and English [3].

2.1.1. Phonetic Transcription

The goal of the *phonetic* module is to provide the pronunciation of the words. This is used not only for producing the test sentences but also for transcribing the training database which is used for building the voices.

For Spanish the pronunciation of each word is based on a set of rules that take into account the transcription rules of Spanish and phonotactics.

Some particular words are transcribed using a lexicon, specially foreign words, abbreviations and signs.

2.2. Prosody

Prosody generation is done by a set of modules that sequentially perform all the tasks involved in prosody modelling: phrasing, duration, intensity and intonation.

2.2.1. Phrasing

Phrasing is one of the key topics in the linguistic part of text-to-speech technologies and consists of breaking long sentences into smaller prosodic phrases. Boundaries are acoustically characterised by a pause, a tonal change, and/or a lengthening of the last syllable. Phrase breaks have strong influence on naturalness, intelligibility and even meaning of sentences.

In Ogmios phrasing is obtained using two algorithms. The first algorithm consists in a Finite State Transducer that translates the sequence of part-of-speech tags of the sentence into a sequence of tags with two possible values: break or non-break [4]. This uses the same tool which was used for the grapheme-to-phoneme task: x-grams [5]. The method uses very few features, but the results are comparable to CART using more explicit features.

The second algorithm predicts phrase break boundaries combining a language model of phrase breaks [6] and probabilities of phrase breaks given contextual features [7]. Phrase break boundaries are found by maximizing the following equation:

$$J(C_{1,n}) = argmax_{j_{1,n}} \prod_{i=1}^{n} \frac{P(j_i|C_i)}{P(j_i)} P(j_i|j_{i-k,i-1}) \quad (1)$$

The latest algorithm was chosen in this evaluation for Spanish due to its better subjective performance in training data.

2.2.2. Duration

Phone duration strongly depends on the rhythmic structure of the language. For example, English is stressed-

timed while Spanish is syllable-timed. Ogmios predicts phone duration with a two steps algorithm: prediction of the suprasegmental duration (syllable or stress unit), and then phone duration is predicted by factoring the suprasegmental duration.

The suprasegmental duration is predicted using CART. Features include the structure of the unit, represented by articulatory information of each phoneme contained in it (phone identity, voicing, point, manner, vowel or consonant), stress, its position in the sentence and inside the intonation phrase, etc.

Once the duration of the suprasegmental unit is calculated, the duration of each phoneme is obtained using a set of factors to distribute suprasegmental duration over its constituent phonemes. These factors are predicted using CART with a set of features extracted from the text, such as articulatory information of the phoneme itself and the preceding and succeeding ones, position in the unit, in the word and in the sentence, stress, and whether the unit is pre-pausal.

2.2.3. Intensity

The intensity of the phonemes is predicted by means of a CART. Features are again articulatory information of the actual, preceding and succeeding phone, stress, and the position in the sentence relative to punctuation and phrase breaks.

2.2.4. Intonation

Ogmios has two available intonation models: a superpositional polynomial model trained using JEMA (*Join feature Extraction and Modelling Approach* [8]), and a *f0 contour selection* model. In some cases, using the superpositional approach results in over-smoothed intonation contours with a loss of expressiveness.

Thus, in this evaluation we generate the f0 contour using the selection approach [9]. For each accent group we select a real contour from the database taking into account the *target cost* (position in the sentence, syllabic structure, etc.) and the *concatenation cost* (continuity). The selected contour is represented using a 4th order Bezier polynomial. The contour is generated using this polynomial, once the time scale is adapted to the required durations. The final result is a more expressive intonation contour than the JEMA model. However, in some cases, the contour is not adequate for the target sentence due to natural language understanding limitations of TTS systems.

2.3. Speech Synthesis

Our unit selection system runs a Viterbi algorithm in order to find the sequence of units $u_1 \dots u_n$ from the inventory that minimises a cost function with respect to the target values $t_1 \dots t_n$. The function is composed by a target and a concatenation cost: both of them are computed as a weighted sum of individual sub-costs as shown below:

$$C(t_1 \dots t_n, u_1 \dots u_n) = w^t \sum_{i=1}^{n} \left(\sum_{m=1}^{M^t} w_m^t C_m^t(t_i, u_i) \right) + w^c \sum_{i=1}^{n-1} \left(\sum_{m=1}^{M^c} w_m^c C_m^c(u_i, u_{i+1}) \right)$$

where w^t and w^c are the weights of the global target and concatenation costs ($w^t + w^c = 1$); M^t is the number of the target sub-costs and M^c the number of concatenation sub-costs; $C_m^t(.)$ is the m th target sub-cost which is weighted by parameter w_m^t; and $C_m^c(.)$ is the m th concatenation sub-cost weighted by w_m^c.

Tables 1 and 2 show the features used for defining the sub-cost functions. There are two types of sub-costs functions. Binary, which can only have 0 or 1 values, and continuous. For continuous sub-costs functions, a distance function is defined and a sigmoid function is applied in order to restrict their range to $[0 - 1]$.

To adjust the target weights, we applied a similar approach to the one proposed in [10]. For each pair of units, we compute their distance using feature vector (MFCC, f0, energy) taken every 5 msec. Let $\overline{d}$ be the vector of all distances for each pair of units, C a matrix where $C(i, j)$ is sub-cost j for unit pair i and $\overline{w}$ the vector of all weights to be computed. If we assume $C\overline{w} = \overline{d}$ then it is possible to compute $\overline{w}$ as a linear regression. In other words, the target function cost becomes a linear estimation of the acoustic distance. The weights of the concatenation sub-costs functions were adjusted manually.

phonetic accent	B
duration difference	C
energy difference	C
pitch difference	C
pitch diff. at sentence end	C
pitch derivative difference	C
pitch deviate sign is different	B
accent group position	B
triphone	B
word	B

Tabla 1. Target costs: B stands for binary cost and C for continuous cost.

energy	C
pitch	C
pitch at sentence end	C
spectral distance at boundary	C
voice-unvoiced concatenation	B

Tabla 2. Concatenation costs: B stands for binary cost and C for continuous cost.

Concerning the waveform generation process, in our experience, listeners assign higher quality scores to the synthetic utterances where the prosodic modifications are minimal. Thus, most of the units selected for generating synthetic speech are simply concatenated using glottal closure instant information, without any prosodic manipulation. Therefore, the use of the information provided by the prosody generation block is restricted to the unit selection process.

3. BUILDING THE ALBAYZIN VOICE

Once the normalization and phonetic transcription rules are ready (section 2.1), our system is able to build a new voice automatically from the audio files and their corresponding prompts. This automatic procedure consists of four main steps: automatic segmentation of the database, training of the prosodic models, selection weights adjust plus database indexing. The prosody training and the selection weights adjust procedures have been described in previous sections. Therefore, in the present section, we will describe the segmentation process and the database indexing.

Once the database was supplied we built the unit inventory. In our system, the units are context dependent demiphones. However, the selection algorithm forces the use of diphones imposing a high cost in phone transitions. The database is automatically segmented into phones by means of the HMM-based aligner named Ramses [11]. We used the front-end described in section 2.1 to automatically transcribe the whole database into phones.

Afterwards, we trained a different set of context dependent demiphone HMM models from each data set, corresponding to each of the three voices. The phone boundaries are determined using a forced alignment between the speech signal and the models defined by the phonetic transcription. A silence model, trained at punctuation marks, was optionally inserted at each word boundary during the alignment. In addition, the detected silences are also used for the pause prediction model (see Section 2.2).

Previous experiments have shown that when a correct phonetic transcription is given, HMM models can achieve similar speech synthesis quality than manual segmentation [12, 13]. Therefore, additional effort was devoted to phonetic transcription and database pruning to obtain correctly segmented voices, as show in the following paragraphs.

Automatic phonetic transcription of a speech synthesis database has to cope with pronunciation variants, pronunciation errors and recording noise. In order to overcome the former problem, the alignment took into account all possible transcriptions of a single word. At this point, the alignment may have errors either because there is a mismatch between front-end and speaker production or because there is an alignment error.

We assume that wrong units will never represent a big portion of the database and that it is affordable to reject such part of it. Therefore we tried to detect undesired units in order to remove them from the inventory by means of a pruning procedure. After computing the alignment likelihood for every unit, 10% of them, those with worst scores, were removed. Previous experiments have shown that it is possible to remove 90% of wrong units by means of this pruning procedure [14].

In this evaluation we do not include any pruning due to the small amount of data provided to generate the synthetic voice. Therefore, we rely on spectral measures at unit selection to avoid problematic units.

Once the speech signals were segmented and the list of sentences are ready, we can start building the voices for our TTS system. The process consists of three main steps: feature extraction, unit indexing and voice generation. The first step extracts F0, duration, energy and MFCC for each speech unit. The index file contains the relevant information needed for computing the target and concatenation costs. In the last step, the parameters of the prosody models and the weights of the unit selection algorithm are computed.

4. ACKNOWLEDGEMENTS

This work has been funded by the Spanish Government (project AVIVAVOZ, TEC2006-13694-C03).

The authors thank to the Albayzin 2008 Team for organising, and the Aho-Lab of the University of the Basque Country (EHU: Euskal Herriko Unibertsitatea).

5. BIBLIOGRAFÍA

[1] Bonafonte, A., Agüero, P. D., Adell, J., Perez, J., and Moreno, A., "Ogmios: The UPC text-to-speech synthesis system for spoken translation", Proceedings of TC-STAR Workshop, Barcelona, Spain, June, 2006.

[2] Bonafonte, A., Moreno, A., Adell, J., Agüero, P.D., Banos, E., Erro, D., Esquerra, I., Perez, J., and Polyakova, T., "The UPC TTS System Description for the 2008 Blizzard Challenge", Blizzard 2008, Brisbane, Australia, September, 2008.

[3] Atserias, J., Casas, B., Comelles, E., Gonzalez, M., Padro, L., and Padro, M., "FreeLing 1.3: Syntactic and semantic services in an open-source NLP library", Proceedings of the fifth international conference on Language Resources and Evaluation (LREC 2006), ELRA. Genoa, Italy. May, 2006.

[4] Bonafonte, A., and Agüero, P. D., "Phrase break prediction using a finite state transducer", Proceedings of the 11th International Workshop on Advances in Speech Technology, Maribor, Slovenia, July, 2004.

[5] Bonafonte, A., “Language modeling using x-grams”, Proceedings of International Conference on Spoken Language Processing, 1996.

[6] Black, A., and Taylor, P., “Assigning Phrase Breaks from Part-of-Speech Sequences”, Proceedings of Eurospeech, 1997.

[7] Agüero, P. D., and Bonafonte, A., “ Phrase break prediction: a comparative study”, XIX Congreso de la Sociedad Española para el procesamiento del Lenguaje Natural. Alcala de Henares, Spain, September, 2003.

[8] Agüero, P. D. and Bonafonte, A., “Intonation Modeling for TTS Using a Joint Extraction and Prediction Approach,”, Proceedings of the International Workshop on Speech Synthesis, Pittsburgh, USA, 67-72, 2004.

[9] Malfrère, F., Dutoit, T., and and Mertens, P., “Automatic prosody generation using suprasegmental unit selection”, Proceeding of the 3rd ISCA Speech Synthesis Workshop, Jenolan Caves, Australia, December, 1998.

[10] Hunt, A., and Black, A., “Unit selection in a concatenative speech synthesis system using a large speech database”, Proceedings of ICASSP, Atlanta, Georgia, 1996.

[11] Bonafonte, A., Mariño, J. B., Nogeuiras, A., and Rodriguez Fonollosa, J. A., “RAMSES: el sistema de reconocimiento del habla continua y gran vocabulario desarrollado por la UPC”, VIII Jornadas de Telecom I+D (TELECOM I+D ’98), Madrid, Spain, October, 1998.

[12] Makashar, M. J., Wightman, C. W., Syrdal, A. K., and Conkie, A., “Preceptual evaluation of automatic segmentation in Text-to-Speech synthesis”, Proceedings of ICSLP, Beijin, China, October, 2000.

[13] Adell, J., Bonafonte, A., Gómez, J. A., and Castro, M. J., “Comparative study of automatic phone segmentation methods for TTS”, Proceedings of ICASSP, Philadelphia, PA, USA, March, 2005.

[14] Adell, J., Agüero, P. D., and Bonafonte, A., “Database pruning for unsupervised building of text-to-speech voices”, Proceedings of ICASSP, vol. 1, Toulouse, France, May, 2006.

THE L^{2}F LANGUAGE VERIFICATION SYSTEMS FOR ALBAYZIN-08 EVALUATION

Alberto Abad and Isabel Trancoso

L^{2}F - Spoken Language Systems Lab
INESC-ID / IST, Lisboa, Portugal
{Alberto.Abad,Isabel.Trancoso}@l2f.inesc-id.pt

RESUMEN

This paper presents a description of the INESC-ID's Spoken Language Systems Laboratory (L^{2}F) Language Verification systems submitted to the ALBAYZIN-08 evaluation. Two completely different systems are presented for the restricted and the unrestricted evaluation. The restricted system relies on Gaussian mixtures models to classify language using the acoustic characteristics of the speech signals extracted by a front-end of shifted deltas. The unrestricted system is a Parallel Phone Recognition and Language Modeling system based on four different phone tokenizers. Results on the development data set for the different systems and evaluation conditions are presented. Additionally, measurements of the computational cost of processing the evaluation data set are provided.

1. INTRODUCTION

The "Red Temática en Tecnologías del Habla" (RTTH) has organized in the recent years a series of evaluations - so called ALBAYZIN evaluations - in some relevant speech processing topics devoted to encourage language research activities on the four official languages of Spain: Castilian, Catalan, Basque and Galician.

Similar to the well-known NIST Language Recognition Evaluation series, a Language Verification (LV) task has been proposed in ALBAYZIN-08. The objective is to determinate if each one of the four official languages of Spain is spoken (or not) in a given test file.

Language verification and recognition approaches can be classified according to the kind of source of information that they rely on. Most successful systems are based on the exploitation of the acoustic phonetics, that is the acoustic characteristics of each language, or the phonotactics which are the rules that govern the phone combinations in a language.

Acoustic systems model each language short-term acoustics by means of stochastic models/classifiers such as Gaussian mixtures models (GMM), Neural Networks (NN) or Support Vector Machines (SVM). Phonotactic systems usually use language dependent stochastic grammars to model phonemes or broad categories of phonemes extracted by a tokenizer.

This paper presents the LV systems developed by the INESC-ID's Spoken Language Systems Laboratory (L^{2}F) for the ALBAYZIN-08 campaign. In accordance with the evaluation conditions, two different systems have been presented: an acoustic system based on GMM modeling for the restricted evaluation (GMM-LV restricted), and a phonotactic Parallel Phone Recognition and Language Modeling system for the unrestricted evaluation (PPRLM-LV unrestricted). The next Section 2 presents a brief description of the task, the data provided for the evaluation and the evaluation metrics. Sections 3 and 4 describe the GMM-LV restricted and the PPRLM-LV unrestricted systems, respectively. Measurements of the computational deployment in the processing of the evaluation data set are also provided. In Section 5 results obtained by the two systems in the different evaluation conditions with the development data set are presented. Finally, Section 6 presents our main conclusions.

2. ALBAYZIN-08 LV: TASK, DATA AND METRIC DESCRIPTION

Detailed information on the ALBAYZIN-08 LV campaign can be found in the evaluation plan document[1].

2.1. Task and evaluation conditions

The task consists of deciding whether a speech segment belongs to each one of the four target languages (Castilian, Catalan, Basque and Galician) or not. That is, for each test signal, four decision results (true or false) for each one of the target languages are produced, together with a score of the decision.

Two system evaluation categories are proposed: one for restricted systems that rely only on the data provided for the evaluation, and another for unrestricted systems which can use any data or incorporate subsystems that have been trained with external data, for instance phone classifiers or voice activity detectors.

Additionally, the systems can be evaluated in closed mode or open mode. In contrast to the closed mode, in the open mode speech segments from unknown languages different from the target ones can appear in the test data

This work was partially funded by the FCT project PoSTPort (PTDC/PLP/72404/2006) and by the European project Vidi-Video.

and are taken into account for the systems performance evaluation.

2.2. Train, development and test data

All the data provided for the ALBAYZIN-08 evaluation are TV programs captured at 16 kHz. The training data set consists of approximately 8 hours per target language, in several files of varying length. The test data set consists of 1800 files with speech of the four target languages and in other unknown languages of 3 different durations: 3, 10 and 30 seconds. Additionally, a development data set consisting of also 1800 files of similar characteristics to the evaluation test set was provided with language identification labels.

2.3. Performance metric

An average performance score based on the false positive and false alarm rates obtained by the evaluating systems is used. The performance score, hereinafter referred to as C_{avg}, is computed independently for each test length duration (3, 10 and 30 seconds). Further details about the metrics can be found in [1].

3. THE GMM-LV RESTRICTED SYSTEM

For the restricted system a GMM acoustic modeling approach was considered to be the most adequate, since it does not need phonetic or word-level transcriptions of any kind. In this section, a detailed description of the developed system is provided.

3.1. Training data splitting

The training data provided for each language (~ 8 hours) was split into two distinct sets: the *models data* of approximately 400 minutes per language was used for training the acoustic models, and the *back-end data* of approximately 100 minutes per language was used for normalizing scores and back-end development.

3.2. System description

The system is a conventional GMM classifier. For each target language, a GMM is trained with the *model data* of that language extracted by the selected front-end. During test, the acoustic language models are used to compute the log-likelihood scores of a given speech signal for each language. These likelihood scores of the four language models are then processed to produce a decision for each target language. Only one GMM classifier for one set of features has been trained. Figure 1 shows a diagram block of the GMM-LV restricted system.

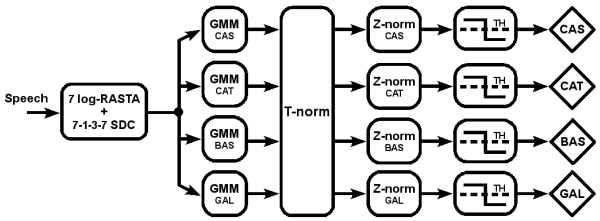


Figure 1. *Block diagram of the GMM-LV restricted system presented for ALBAYZIN-08 evaluation.*

3.2.1. Feature extraction

The extracted features are Perpetual Linear Prediction static features with log-RelAtive SpecTrAl speech processing (log-RASTA), and a stacked vector of shifted delta cepstra (SDC) of the same log-RASTA features. Concretely, 7 log-RASTA static features and a 7-1-3-7 SDC parameter configuration are computed, resulting in a final feature vector of 56 components.

On the one hand, log-RASTA features are known to be a robust representation for speech processing applications [2]. On the other hand, it has been shown that the use of SDC features (created by stacking delta cepstra computed across several frames) allows improved performances in LV tasks [3]. The selected front-end showed remarkable improvements compared to other evaluated feature representations during the development of the systems, such as mel-frequency cepstral coefficients, perceptual linear prediction features or the advanced ETSI front-end features.

3.2.2. Acoustic modelling

Gaussian mixture models of 1024 mixtures were trained for each target language using the *models data*. Each acoustic model was first initialized by means of vector quantization estimation. Then, 10 maximum likelihood Estimation Maximization iterations were applied to obtain the final language models.

More sophisticated model training procedures were also tested, without achieving significant differences. In particular, the use of a universal background model (UBM) for Bayesian adaptation to the target languages. This method was finally discarded, and UBM has not been used neither for model adaptation nor for score normalization.

3.2.3. Back-end: normalization and scoring

For each test utterance, log-likelihoods of the four acoustic models (Castilian, Catalan, Basque and Galician) were obtained. The log-likelihood of the claimed or tested language was T-normalized with the mean of the log-likelihoods of the other three competing languages.

The *back-end data* was split in shorter segments, according to an energy-based speech detector segmentation

system, in order to be more similar to the test data. T-norm scores of this *back-end data* were computed. Then, mean and variance of the T-norm score of a concrete language was estimated with the *back-end data* sets of the competing languages. During test, the mean and variance estimated for a target language were used to apply Z-norm to the T-normalized log-likelihood of this language, in order to obtain the final score used for decision.

The final decision threshold was selected in order to deploy a balanced performance (close to minimum C_{avg}) for the 3, 10 and 30 seconds evaluation conditions. It is worth noticing that a unique threshold is used independently on the claimed target language.

The difference between the closed and open systems for the restricted evaluation is on the decision threshold selected, that is slightly more selective in the open system to reduce the increased false alarm rate due to presence of unknown language speech sources.

3.3. Processing time

All the experiments in this paper were run in an Intel Quadcore 2.4 GHz (Q6600) machine with 8 GBytes of DDR2 RAM at 667 Mhz.

The total time deployed by the restricted system (both closed and open) in performing the test evaluation was approximately 19 minutes. Since the evaluation test set has a total duration of around 458 minutes, this result corresponds approximately to 0.04xRT.

4. THE PPRLM-LV UNRESTRICTED SYSTEM

The unrestricted system presented for the ALBAYZIN-08 LV evaluation is a PPRLM system that exploits the phonotactic information extracted by four parallel tokenizers: European Portuguese, Brazilian Portuguese, European Spanish (Castilian) and American English. The key aspect of this type of systems is the need for robust phonetic classifiers that generally need to be trained with word-level or phonetic level transcriptions. At the INESC-ID's L^2F group we have been working for several years in Large Vocabulary Continuous Speech Recognition (LVCSR) using hybrid Artificial Neural Network Hidden Markov models (ANN/HMM) recognizers, the so-called connectionist paradigm. During the last years, we have been developing phonetic classifiers (Multi-layer Perceptrons, MLP) for our current recognition systems in several languages. In this section, we present a detailed description of the PPRLM-LV system and its several components.

4.1. Training data splitting

Like in the case of the GMM-LV system, the training data has been split into two sets, one for training stochastic language models (*models data* of approximately 400 minutes per language), and the other for back-end development (*back-end data* of approximately 100 minutes per language).

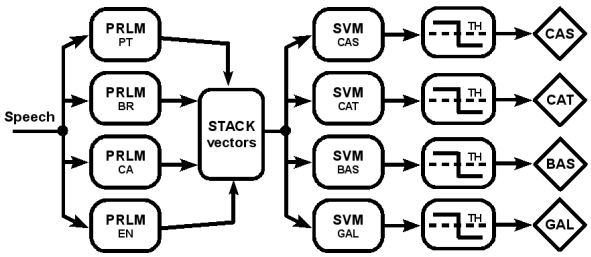


Figure 2. *Block diagram of the PPRLM-LV unrestricted system presented for ALBAYZIN-08 evaluation.*

The four phonetic classifiers used by the PPRLM-LV system were trained with additional external data. For the European Portuguese classifier, 57 hours of manually annotated data and more than 300 hours of automatically transcribed broadcast news (BN) data were used. The Brazilian Portuguese classifier was trained with around 13 hours of BN data. The Spanish system used 14 hours of manually annotated data and 78 hours of automatically transcribed data. Finally, the English system was developed with the HUB-4 96 and HUB-4 97 data sets, that contain around 142 hours of TV and Radio Broadcast data.

4.2. System description

The PPRLM-LV unrestricted system first uses the four phonetic tokenizers to extract the phonetic sequence of the *model data* of each target language. Then, for each target language and for each tokenizer a different phonotactic n-gram language model is trained. During test, the phonetic sequence of a given speech signal is extracted with the phonetic classifiers and the likelihood of each target language model is evaluated, resulting in a total of 16 likelihood scores (4 target languages x 4 phonetic tokenizers). These likelihood scores are normalized and combined with a Support Vector Machine approach for obtaining a final identification and probability score per target language. Figure 2 shows a diagram block of the PPRLM-LV unrestricted system.

4.2.1. Phonetic tokenizers/classifiers

The tokenization of the input speech data in both training and testing is done with the neural networks that are part of our hybrid Automatic Speech Recognition (ASR) system named AUDIMUS. This kind of recognizers are generally composed by one or more phoneme classification networks, particularly MultiLayer Perceptrons (MLP), that estimate the posterior probabilities of the different phonemes for a given input speech frame (and its context). Concretely, the system combines three MLP outputs trained with Perceptual Linear Prediction features (13 static + first derivative), log-RelAtive SpecTrAl features (13 static

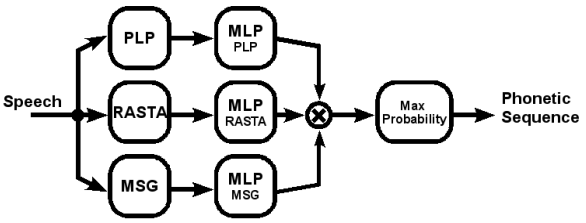


Figure 3. *Block diagram of a multi-stream MLP phonetic classifier used in the European Portuguese, Brazilian Portuguese, Castilian and English PRLM systems.*

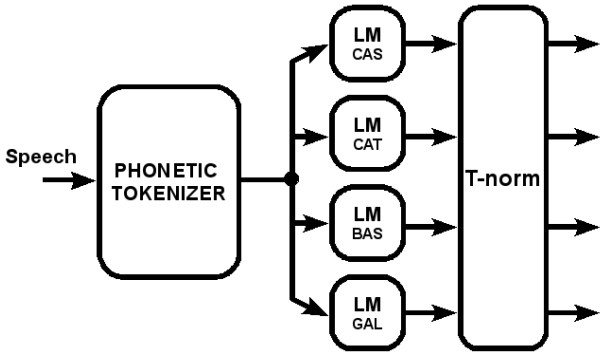


Figure 4. *Block diagram of one PRLM language verification system.*

+ first derivative) and Modulation SpectroGram features (28 static). Figure 3 shows the structure of one of the multi-stream phonetic classifiers used in this work. A detailed description of the European Portuguese BN transcription system can be found in [4].

The size of the neural networks of each ASR system (European Portuguese, Brazilian Portuguese, European Spanish and American English) differs due to the different amounts of training data. However, it is worth noticing the differences in the output layer, that is, the number of different phonetic tokens to classify. In the case of European Portuguese, 39 phonetic tokens (complete Portuguese phone set + silence) are considered. In the Brazilian, Spanish and English recognizers, besides the complete phone set of each language plus silence, additional sub-phonetic units are also classified. These units are mainly phoneme regions (transitional left and right regions and steady phone nucleus) and diphone units [5].

4.2.2. Phonotactics modeling and normalization

For every phonetic or sub-phonetic tokenizer, the phonotactics of each target language are modelled with a 3-gram model. For that purpose the SRILM toolkit has been used [6].

During test, a vector with the four likelihoods obtained with four competing target language models is formed for every tokenizer. Similarly to what was done for the GMM-LV system, we decided not to use background models for score normalization. Instead of background normalization, T-norm of the mean likelihood of the three competing languages was applied to the score of a claimed language. A diagram of one single phonotactic system is shown in Figure 4.

4.2.3. Linear SVM combination

In order to combine the 4-element (one per target language) T-normalized vectors obtained from each independent phone tokenizer, linear Support Vector Machines are trained with the libSVM toolkit [7].

The *back-end data* portion was first segmented into shorter segments by a speech-non-speech detector, and the four 4-element T-normalized vectors scores were extracted and stacked to form a single 16-element vector. Then, four binary "1 versus all" classifiers were trained for every target language in order to obtain probability estimations. In fact, due to the high confusability between Castilian and Galician observed during the development of the system, it was decided to use a two step classification procedure when claimed languages were Castilian or Galician. A "2 versus all" classifier is trained to detect Castilian and Galician and then a "1 versus 1" classifier is used to disambiguate between these two. It is worth noticing that the SVM classifiers were used to estimate the probability of each target languages and that decisions were finally taken based on this probability score and the decision threshold selected.

Like in the GMM-LV system, the decision threshold was adjusted to obtain the best possible performance for the several evaluation conditions. Again, a different threshold is selected for the closed and open evaluation modes.

4.3. Processing time

Using the above mentioned machine, the total time deployed by the unrestricted system (both closed and open) was approximately 148 minutes, corresponding to approximately 0.32xRT. It is worth to notice that the time consumed on loading the phonetic networks of each one of the PRLM systems is included in this time computation and that the networks are loaded for each testing file.

5. RESULTS ON THE DEVELOPMENT SET

Table 1 presents the results obtained in the development set, for the two systems in all the described conditions. The results confirmed our expectations. The best ones were obtained with the unrestricted system. The open mode is significantly more challenging than the closed

one. The use of longer segments contributes to a smaller error rate.

System	Condition	30 sec	10 sec	3 sec
Restricted	Closed	0.1556	0.1986	0.2462
Restricted	Open	0.1952	0.2221	0.2648
Unrestricted	Closed	0.0281	0.0663	0.1635
Unrestricted	Open	0.0838	0.1148	0.1969

Table 1. *C_{avg} performance on the ALBAYZIN-08 LV development set of the GMM-LV restricted and the PPRLM-LV unrestricted systems in both open an closed mode.*

6. SUMMARY AND CONCLUSIONS

The experiments described in this paper using both a restricted system based on GMM models and an unrestricted system based on phonotactic models confirmed the advantages of using extra knowledge sources in the language verification task. It would be interesting to add to the dataset the other language spoken in the Iberian Peninsula (European Portuguese), as well as Brazilian Portuguese and the different Latin American Spanish varieties, and detect the confusability between all the languages.

7. BIBLIOGRAPHY

[1] "Plan de Evaluación de Sistemas ALBAYZIN-08 Verificación de la Lengua (ALBAYZIN-08 VL)", URL: http://jth2008.ehu.es/Plan_Albayzin-08_VL_final.pdf.

[2] Hermansky, H. and Morgan, N., "RASTA processing of speech", IEEE Transactions on Speech and Audio Processing, Vol. 2(4), pp 578-589, Oct 1994.

[3] Torres-Carrasquillo, P. A. et alt., "Approaches to Language Identification using Gaussian Mixture Models and Shifted Delta Cepstral Features", in Proceedings of ICSLP 2002, pp. 89-92, Denver, Colorado, Sep 2002.

[4] Meinedo, H., Caseiro, D., Neto, J. and Trancoso, I., "Audimus.media: a broadcast news speech recognition system for the european portuguese language", in Proc. PROPOR 2003, Faro, Portugal, 2003.

[5] Abad, A. and Neto, J., "Incorporating Acoustical Modelling of Phone Transitions in an Hybrid ANN/HMM Speech Recognizer", in Proc. INTERSPEECH-08, Brisbane, Australia, Sep 2008.

[6] Stolcke, A., "SRILM - An Extensible Language Modeling Toolkit", in Proc. ICSLP 2002, Denver, Colorado, Sep 2002.

[7] Chang, C.-C. and Lin, C-J, "LIBSVM - A Library for Support Vector Machines", URL: http://www.csie.ntu.edu.tw/ cjlin/libsvm/index.html.

SESIÓN ORAL 2
SÍNTESIS DEL HABLA

Bayes Optimal Classification for Corpus-Based Unit Selection in TTS Synthesis

Hamurabi Gamboa Rosales, Oliver Jokisch and Ruediger Hoffmann

Laboratory of Acoustics and Speech Communication, Dresden University of Technology, Germany
[Hamurabi.Gamboa,Oliver.Jokisch,Ruediger.Hoffmann]@ias.et.tu-dresden.de

Abstract

Generally, corpus-based speech synthesis systems provide a considerable synthesis quality since the underlying unit selection approaches were optimized in the last decade. The unit selection of the synthesizer is attempting to find the best combination of unit sequences to assure that the perceptual differences between expected (natural) and synthesized speech signal are as low as possible. Depending on database and algorithmic design, numerous mismatches and distortions are possible and they are audible in the synthesized speech signal. Therefore, unit selection strategy and parameter tuning are still important issues. We present a novel concept to increase the efficiency of the exhaustive speech unit search within the database by an unit selection model, which is based on mapping analysis of only the concatenation costs and Bayes optimal classification (BOC). BOC has one principle advantage because it does not require an exhaustive training to set up weighted coefficients for target and concatenation sub-costs. It can provide an alternative for unit selection but requires further optimization e. g. by integrating target cost mapping.

Index Terms: speech synthesis, unit selection, speech intelligibility, speech analysis.

1. Introduction

Corpus-based concatenative speech synthesis has been studied and utilized in text-to-speech synthesis (TTS) systems over many years [1], [2]. In this approach, the speech database design covers a big variety of the phonetic and prosodic language features. Consequently, unit selection should be able to find the best unit sequence to synthesize an input text by minimizing the total cost function. The total cost function is modeled as the weighted sum of target and concatenations costs, which contains various features such as duration, F0 and energy as target but also linear spectral frequencies (LSFs), multiple centroid analysis (MCAs) [3] and Mel frequency cepstral coefficients (MFCCs) as concatenation features.

The target cost is defined as the estimation of the mismatch between a recorded acoustic speech unit and a predicted specification, which is estimated by using the prosody module of the TTS system. It is calculated as the weighted sum of characteristic distances between the components of the target and candidate feature vector like duration, pitch value (F0) and energy. Likewise, the concatenation cost reflects the mismatch or distortion between two speech units due to the frequency formants and others spectral features of the speech units that do not align properly [4]. Mismatches are known as concatenation cost, which could be considered as an estimator of the quality of speech synthesis. If the discordance between a speech unit and the predicted specification is also taken into account, the quality of the synthesized speech signal even suffers an extra degradation. Therefore, it is necessary to set up all these factors in one integrated function, which represents the influence of target and concatenation costs on the resulting speech synthesis quality and enables the finding of optimal speech units sequences to obtain the desired synthesized waveform. But the processing of all information requires an exhaustive training to set up the weighted coefficients for both sub-costs [1][2]. Therefore, we present an unit selection framework based on Bayes optimal classification (BOC) and its experimental evaluation. BOC has a principle advantage because it does not require an exhaustive training to set up weighted coefficients for target and concatenation sub-costs. Section 2 gives an overview about the proposed unit selection framework and its components. The BOC is described in section 3 and the experimental results are explained in section 4.

2. Unit Selection Framework

The target and concatenation costs have been integrated in a total cost function by [1], which represents the degradation on a synthesized speech signal. Additionally, they described a unit selection model as a search for a low cost candidate unit sequence. Although, different target and concatenation sub-costs have been proposed to unit selection, the sub-costs already mentioned have reached a significantly representation of the deterioration of a synthesized signal. Hence, they compose a special unit selection process in such a way that the sum of the target and concatenation costs determines the total cost C for a sequence of n speech units.

$$C(t^n, u^n) = \sum_{i=1}^{n} \sum_{j=1}^{p} w_j^t c_j^t(t_i, u_i) + \sum_{i=2}^{n} \sum_{j=1}^{q} w_j^c c_j^c(u_{i-1}, u_i) \quad (1)$$

Where t_j represents the searched predicted specification, u_i the speech unit, c_j^t target cost, p the number of weighted target sub-costs, c_j^c concatenation cost, q the number of concatenation sub-costs and w the weighted coefficients (WCF). The following step should be to find the weighted coefficients that determine the effect-weight of every target and concatenation sub-cost in the total cost function. This is considered as the best way to find the right speech unit sequence for the desired synthesized speech signal. However, the search for the optimal weighted coefficients is not a trivial task, because it normally requires training, which is a subjective work and time consuming for every speech database [1][5][7][8]. Therefore, we present a unit selection framework that is based on mapping of the concatenation sub-costs and a Bayes Classifier. Therewith, we avoid principally the exhaustive and subjective search of weighted coefficients. Also, we estimate in great part the quality or degradation of the synthesized signal by mapping the concatenation sub-costs.

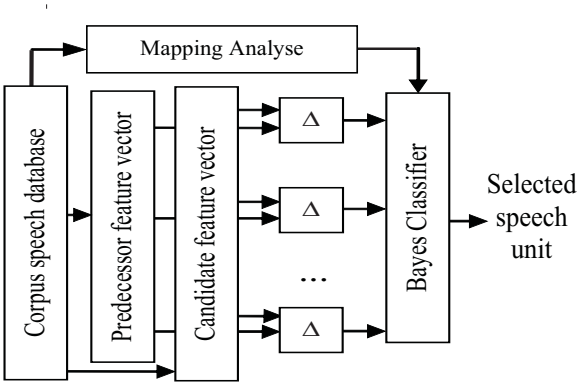


Figure 1: *Proposed Unit Selection Framework.*

2.1. Bayes classification framework

The Bayes classification framework is composed by different modules that are shown in Fig. 1. It illustrates the speech database, where all possible speech units that compose the desired synthesized speech signal are searched. The speech unit candidates are chosen in the speech database by Backward Oracle Matching algorithm (BOM) [6]. It picks up all possible speech units that compose the phonetic sequence of the text to be synthesized. Once the speech units are found, their MCAs, LSF and MFCCs coefficients are calculated at the right and left boundaries and represented in a vector form. Afterwards the distance Δ between predecessor and candidate speech unit sequence of the desired synthesized speech signal is calculated. The mapping is obtained by calculating the concatenation sub-costs distance of the speech units. Finally, the Bayes classification determines if the concatenation between the speech units is corrupt or proficient. In the following sections the components of the proposed unit selection framework will be described in more detail.

2.2. Speech corpus and database

We utilize the "TC-STAR" English speech database [9], which was designed with high quality criterion. The quality speech in the recordings was reached with 96 kHz sampling rate, 16 Bit precision, SNR $>$ 40 dB and bandwidth of 40 Hz to 20 kHz. The speech database has a duration of over 10 hours. It is composed for a corpus of about 90 000 words, which are contained in 5558 sentences. This amount is distributed on the subcorpora of transcribed speech, written text, constructed phrases and expressive speech. The 70% of the sentences were labeled automatically and 30% were hand labeled, where 30% cover all Diphonemes of the English language. Therefore, there is at least one error free labeled Diphoneme in the speech database. Also, the Diphoneme is established as the basic speech unit.

2.3. Parametric distance function

The Delta symbols in Fig. 1 show the distance function. They compare two values of the same feature and produce a distance value output. This function measures the degree of match between the features of two adjacent speech unit candidates. The distance is calculated with 20 ms frames, 9 MCAs, 26 LSFs and 24 MFCCs coefficients features vector in the corresponding boundary at the point of concatenation. We utilize the Mahalanobis distance measure, because it has shown a high correlation with human perception of discontinuity at concatenation boundaries [10].

$$d(\vec{x},\vec{y}) = \sqrt{(\vec{x}-\vec{y})^T K^{-1} (\vec{x}-\vec{y})} \quad (2)$$

Where $\vec{x}$ and $\vec{y}$ are the features vectors of the predecessor and candidate speech units and $K^{-1}$is the inverse covariance matrix. That is how the distance between the speech units for the MCA, LSF and MFCC features is calculated.

2.4. Mapping Analysis

The mapping consists of an off-line calculation of the concatenation sub-costs between speech units in the database, which do and do not present displeasing distortions when they are concatenated. It is estimated by the distance calculation between the speech unit features like MCAs, LSFs and MFCCs at the right and left boundaries [11]. The mapping of concatenation sub-costs that do not present any distortion is done by the concatenation sub-cost distance calculation between speech units, which are continuous in the words or sentences in the speech database. Although, the concatenation sub-costs of continuous speech units are set up to zero by definition [1][2], we utilize the calculated concatenation sub-cost distances to map the real values of continuous speech units off-line. In this way, we obtain a real reference of concatenation sub-costs without distortion. The mapping of those concatenation sub-costs that present unpleasant distortions is done by using a determined set of speech units. These speech units come from different words or sentences contained in the speech database and were previously selected to not concatenate properly by a listening test on synthetic utterances. Therewith, the second reference is also obtained with the same number of concatenation samples like the properly concatenation samples. We were able to differentiate between two mapped references, which represent the proficient and corrupt areas of concatenation as it is shown in the Fig. 2. It illustrates the mapping of the concatenation sub-cost distances between continuous and not continuous speech units at the point of concatenation. For this instance the concatenation type at the middle of a short vowel /U/ is shown, because the concatenation between short vowels has proved to be the most inclined case to concatenate not properly [5][11]. Finally, a mapping for every phoneme concatenation should be done. Consequently, the next task is to determine the concatenation sub-cost area, which can determine if the join between two not continuous speech units is a proficient or corrupt concatenation based on the corresponding previously mapping pro phoneme by a classification method like BOC.

3. Bayes Optimal Classification

Bayes optimal classification establishes that the class probability k given the feature vector $\vec{x}$ is equal to multiplication between the a priori likelihood the class $P(k)$ and the density probability function $P(\vec{x}/k)$ divided by the probability of the sample, according to equation (3).

$$P(k/\vec{x}) = \frac{P(\vec{x}/k) \cdot P(k)}{P(\vec{x})} \quad (3)$$

Where k is the proficient or corrupt concatenation class and $\vec{x}$ is the concatenation sub-cost distance vector between two speech units. The denominator is not considered, because it is common to both concatenation classes. A priori probabilities of continuous and not continuous concatenation sub-costs have been assumed equal 0.5. Also, we assumed the independence

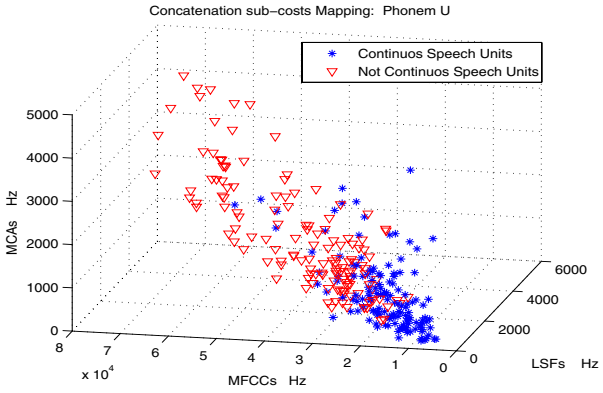


Figure 2: *Concatenation sub-costs mapping.*

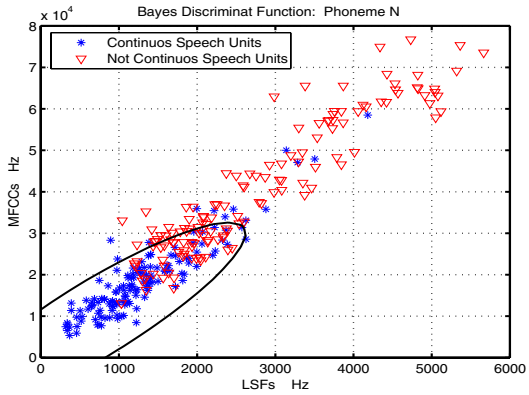


Figure 3: *Bayes discriminant function.*

between feature vectors, so that the BOC combines the impact and probability of feature vector on the class label. BOC was modeled with a multivariate density Gaussian distribution [12] considering that the feature vectors have a normal distribution as is shown in the following equation (4).

$$P(\vec{x}/k) = \frac{1}{(2\pi)^{N/2}\,|K_k|^{1/2}} \cdot \exp\left[-\frac{1}{2}(\vec{x}-\vec{\mu_k})^T K_k^{-1}(\vec{x}-\vec{\mu_k})\right] \quad (4)$$

Where $\vec{x}$ is Mahalanobis distance by concatenating speech units, the covariance matrix K and mean μ are calculated according to the class feature vectors of Mahalanobis distance. Afterwards we would like to find those speech units that have the maximum probability. It is achieved by a discriminant function as it is described in the following equation (5) and (6) .

$$e = \arg\max_{i=1,\ldots K} \quad d_i(\vec{x}) \quad (5)$$

$$d_i(\vec{x}) = P(k) \cdot P(\vec{x}/k) \quad (6)$$

Where e is the maximum argument of the discriminant function $d_i(\vec{x})$ in the equation (6), which contains the maximum probability. K is the number of classes (corrupt or proficient concatenation type).

3.1. Bayes discriminant function

By the substitution of the multivariate density Gaussian distribution (4)in the discriminant function (5) we obtain the corresponding distance Bayes discriminant function (7) as it is shown in the Fig. 3. The discriminant function allows to classify a concatenation between two not continuous speech units into corrupt and proficient concatenation type, which is based on its probability estimation.

$$d_k^*(\vec{x}) = \ln\left[d_k(\vec{x})(2\pi)^{N/2}\right] \quad (7)$$

$$d_k(\vec{x}) = \ln P(k) - \frac{1}{2}\ln|K_k| - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} K_k^{(mn)}\mu_{mk}\mu_{nk} + \sum_{n=1}^{N}\Big(\sum_{m=1}^{N} K_k^{(mn)}\mu_{mk}\Big)\cdot x_n - \frac{1}{2}\sum_{m=1}^{N} K_k^{(mn)} x_m^2 - \sum_{m=1}^{N-1}\sum_{m>m}^{N} K_k^{(mn)} x_m x_n$$

Equation (7) describes a Bayes discriminant function [12], which can be used to calculate the corresponding discriminant function for every concatenation of phonemes of not continuous speech units in the speech database. The Bayes discriminant function at the point of concatenation of the nasal phoneme /N/ is shown in Fig. 3. It illustrates 2-Dimensional mapping analysis, where the both concatenation areas are delimited by the Bayes discriminant function. It is easy to recognize that some concatenation sub-costs of not continuous speech units fall inside the proficient concatenation area, which is known as classification error [12].

3.2. Unit selection process

Firstly, the speech units that had been found for the desired synthesized speech signal by the BOM are processed by the Bayes Classifier. The BOC classifies the speech units in corrupt and proficient concatenation types by using the corresponding discriminant function, as it is shown in Fig. 3. Then, the speech units, which were found to concatenate corrupt and whose concatenation sub-costs do not fall into the proficient concatenation delimited area by the discriminant function, are removed from the unit selection process. Afterwards, the left over speech units are computed by the corresponding previously obtained distribution of the proficient concatenation type by using maximum likelihood method. Finally, the concatenation of speech units that shows the highest likelihood is selected. In this way, these speech units are selected that match best with the searched phoneme sequence to obtain the desired speech signal without distortions by the concatenative-based speech synthesis.

4. Listening Test

The DreSS TTS system [13] was used to synthesize three blocks of 10 utterances with three different unit selection methods. Furthermore, the previously mentioned speech database “TC-STAR” was utilized for the three unit selection methods in the DreSS TTS system. The first block (Conventional US) was synthesized by the unit selection method proposed in [1], which represents the basic principles of the sum of target and concatenation costs for unit selection and requires an exhaustive training to set up the weighted coefficients for target and concatenation sub-costs. By the second block (Masking US) the unit selection proposed in [14] was used, which bases its unit selection method on previously defined transparency and qual-

ity functions and determines if a determined concatenation will or will not present distortions. The last block (BOC US) is our proposed unit selection method.

4.1. Experiment

The synthesized utterances were evaluated by 10 listeners and finally a mean opinion score (MOS) of their absolute category decisions has been calculated. All listeners were students or researchers at Dresden University of Technology with good English proficiency, experience on speech recognition and synthesis. Their age varied from 20 to 30 years. The listening test consisted of the evaluation of intelligibility, naturalness and concatenation quality of the synthesized utterances. The probands listened to the test stimuli in random order. We asked them to rate the quality of the synthesized utterances on a scale of 1 (Bad) to 5 (Excellent). The MOS values obtained for the three unit selection methods are summarized in Table 1.

Table 1: *Mean opinion score listening test.*

Conventional US	Masking US	BOC US
2.76	2.25	2.67

4.2. Results

The mean opinion scores in Table 1 turned out to be significant at the one percent-level by paired t-test. Masking US based on the masking quality function has obtained the worst results in the listening test. This is due to the quality concatenation masking function that can not be determined by a linear function for every type of concatenation as it was proposed by [14]. Conventional US based on the sum of target and concatenations costs, performed slightly better than BOC. This reflects the potential improvements that can be obtained by taken into account the target sub-costs in the speech synthesis. Nevertheless, the task of setting up the weight coefficients on the total cost C function in the equation (1) was a very difficult subjective work, which required many hours of listening training for the specific corpus database. Summarized, the proposed BOC unit selection obtained better results than the proposed masking method of unit selection and it was slightly worse than the conventional unit selection method manifesting only a small perceptive difference between them. BOC unit selection performance is functional since it has shown an acceptable quality and avoided many hours of training to determine an appropriate search for the best speech unit sequence by mapping the concatenation sub-costs, which is mainly considered as a subjective task.

5. Conclusion

This paper presented another perspective on unit selection methods for corpus-based speech synthesis by proposing a Bayes optimal classifier. BOC unit selection is based on concatenation and sub-costs mapping of speech units representing distortions in the concatenated unit sequence. In this method, the mapping provides two references of proficient and corrupt concatenation areas. Furthermore, a discriminant function as shown in the equation (7) was developed, which calculates the probability estimation of proficient and corrupt concatenation type between two speech units by this discriminant function. BOC has one principle advantage because it does not require an exhaustive training to set up the weighted coefficients for target and concatenation sub-costs. Its operation is based on an objective mapping of the concatenation sub-costs. Therefore, BOC unit selection supports the integration of new speech databases in a TTS system avoiding exhaustive training for each newly integrated speech database. In future, it will be important to improve the BOC unit selection performance by the integration of a target cost mapping because the target cost has a great influence on the naturalness of the synthesized speech signal.

6. References

[1] Hunt, A.J. and Black, A.W., "Unit selection in a concatenative speech synthesis using a large speech database", in Proc. ICASSP, pp. 373-376, 1996.

[2] Beutnagel, M., Conkie, A. and Syrdal, A.K., "Diphone synthesis using unit selection", Proc. 3rd ESCA/COCOSDA International Workshop on Speech Synthesis, Jenolan Caves, pp. 185-190, 1998.

[3] Corwe, A. and Jack, MA., "Globally optimizing formant tracker using generalized centroids", Electronic Letters, Vol 23, No. 19, pp 1019-1020 Beijing,China, 1987.

[4] Gamboa Rosales, H. "Evaluation of smoothing methods for segment concatenation based speech synthesis", In Proc 16th Czech-German Workshop "Speech Processing", September 77-83, Prague, Czech Republic, pp. 270-273, 2006.

[5] Toda, T., Kawai, H., Tsuzaki, M., Shikano, K., "An Evaluation of Cost Functions Sensitively Capturing Local Degradation of Naturalness for Segment Selection in Concatenative Speech Synthesis", Speech Communication, Vol. 48, No. 1, pp. 45-56, Jan. 2006.

[6] Navarro, G. and Raffinot, M., "Flexible Pattern Matching in String", Cambridge University Press, 2002.

[7] Alas, F., Llor, X., Formiga, L., Sastry, K., Goldberg, D. E., "Efficient Interactive Weight Tuning For TTS Synthesis: Reducing User Fatigue By Improving User Consistency", In Proc. ICASSP, Toulouse, France, pp. 865-868, 2006.

[8] Vepa, J. and King, S., "Subjective evaluation of join cost functions used in unit selection speech synthesis", In Proc INTERSPEECH, Jeju Island, Korea, pp 1181-1184. 2004.

[9] Hain, H., Racky, J., Volk, T., "The Papageno TTS System", In Proceedings of the TC-Star Workshop 2006, Barcelona, Spain, June 2006.

[10] Vepa, J., King, S. and Taylor, P., "Objective distance measures for spectral discontinuities in concatenative speech synthesis", In ICSLP, Denver, USA, 2002.

[11] Gamboa Rosales, H., Jokisch, O. and Hoffmann, R., "Spectral distance costs for multilingual unit selection in speech synthesis", In Proc. of 11-th International Conference "Speech and Compute" SPECOM2006, St. Petersburg, Russia, pp. 270-273,2006.

[12] Hoffmann, R., "Signalanalyse und -erkennung", Ed. Springer,1998.

[13] Gamboa Rosales, H. and Jokisch, O., "KorpusDress1 - Korpusbasierte Konkatenative Sprachsynthesesysteme", In Proc 18. Konferenz Elektronische Sprachsignalverarbeitung, Cottbus, Germany, pp. 115-122, 2007.

[14] Coorman G., Fackrell, J., Rutten, P. and Van Coile, B.,"Segment selection in the LH Realspeak laboratory TTS system", In Proc of ICSLP, pp. 2:395-398, Beijing, China, 2000.

FLEXIBLE HARMONIC/STOCHASTIC MODELING FOR HMM-BASED SPEECH SYNTHESIS

Eleftherios Banos , Daniel Erro, Antonio Bonafonte, Asuncion Moreno

TALP Research Center, Universitat Politècnica de Catalunya, Spain
e-mail:{lefteris,derro,antonio,asuncion}@gps.tsc.upc.edu

Abstract

In this paper the preliminary results, of a new approach on speech modeling for statistical parametric HMM-based speech synthesis are presented. The proposed system is based on a flexible pitch-asynchronous harmonic/stochastic model (HSM) [1]. The speech is modeled as the superposition of two components: a harmonic component and a stochastic or aperiodic component. The fact that the synthesis model is pitch-asynchronous allows the direct integration to a HMM-based synthesis system. HTS [2], an open source software toolkit that provides HMM-based speech synthesis was used. The proposed HSM method was compared to the HTS baseline system with the same configurations and database. A number of different experiments were conducted. Results show that high quality of synthesized utterances is reached. A small perceptual test was carried out comparing the two systems on quality of the synthetic voice and similarity to the original voice. HSM outperforms the HTS baseline system in the quality test: HSM $53\,\%$, HTS $35{,}3\,\%$, and undecided $11{,}7\,\%$. Concerning similarity to the original voice, HSM-performed slightly better than HTS: HSM $35{,}3\,\%$, HTS $29{,}4\,\%$, and undecided $35{,}3\,\%$.

1. INTRODUCTION

Unit-selection is the dominant method in speech synthesis [3] due to performance advantages such as high quality, and naturalness of synthetic speech. However unit-selection systems are highly dependent on the database and the quality of the recorded database. Due to this quality dependency, voice modification at the selected units cannot be carried out, and voice conversion/adaptation is a difficult task by the time being, for unit selection systems. Furthermore, databases where perfect recording conditions are not possible to achieve cannot be used. Additionally big storage memory is necessary, which is prohibitory in specific applications. Because of these limitations much research has moved to statistical parametric speech synthesis and mainly to Hidden Markov Models (HMM)-based systems.

Statistical parametric speech synthesis (from now on we refer to HMM method only), cannot offer yet a high speech quality comparing to unit-selection, but definitely overcomes most of the problems listed above offering a very wide area for further research (i.e.polyglot systems). In addition, HMM theory and mathematics are well established in many areas of speech technology. The benefits of applying HMM to speech synthesis are numerous: (i) it is possible to take advantage from techniques tested in different fields and adapt them to a different application (i.e. speech recognition to speech synthesis); (ii) the limitations of HMMs are known; (iii) as the basic concept of HMMs is the same for all applications, high-level implemented systems can be used for different research fields and applications[4].

Moreover, continuous improvement has been observed at HMM-based-text-to-speech systems. To be more specific, according to the Blizzard challenge 2005 [5], 2006 [6], and 2007 [7], HTS system show a significant improvement every year. Although on [8] the organizers of the Blizzard evaluation, provide the results without pointing to each system by name, someone can have information about the evaluation methods. On [9] HTS researchers presented an evaluation of their own system for the three year Blizzard challenge. On 2005, HTS participated with a number of changes on the basic system [10]: a STRAIGHT-based high quality vocoding algorithm used for the F0 extraction, and spectral and aperiodic analysis, resulted to reduce the "buzzy" sound that was produced with the basic vocoding technique. Hidden-semi-Markov models(HSMMs) were used for improvements on duration modeling. Parameter generation from HMMs considering global variance (GV) was applied to reduce the oversmoothing of the generated parameters.

For Blizzard challenge 2006, a semi-tied covariance matrix was used for full-covariance modeling in the HSMMs, and the structure of the covariance matrices for the GV pdfs changed from diagonal to full covariance. The system that was used for the first two Blizzard challenge was a speaker-dependent system.

On 2007, a new speaker-independent system was introduced [7]. The system was guided from speaker adaptation approaches. The general results were satisfactory every year and on some occasions over expectations.

Two main areas of research in HMM-based synthesis are (i) improving the quality of the synthesized speech in terms of naturality and similarity to the original training voice, and (ii) training with a small amount of data. This paper focuses on the first one, which is closely related to the speech parametrization used by the system and its associated reconstruction method. In this paper, a high quality asynchronous (harmonic/stochastic model (HSM) [1]), is applied. The main problem of HSM modeling to be solved can be centralized on the voiced/unvoiced transitions where the separation of the harmonic part and stochastic part is not very precise. Vector generation takes advantage of multi-space distribution HMMs to separate as more precise as possible the Harmonic generation part from the stochastic generation part. Preliminary test show that the synthesized voice has a natural tinge, maintaining the main characteristics of the speaker voice, and outperforms HTS system using the same database and same configurations.

The remaining part of this paper is organized as follows: At Section 2 a technical description of the asynchronous HSM model is given. At Section 3 the integration of the HSM model to HMM system is discussed. Further, at Section 4 a general description of an HMM-based synthesis system is given. At Section 5 the main experiments are presented, and the results of a small perceptual test are discussed. Finally, concluding remarks followed by our future intentions and work plans to improve our model, are presented.

	Harmonic component	Stochastic component
Voiced frames	f0, {Aj}, {φj}	LPC filters
Unvoiced frames	-	

Table 1. *General HSM analysis scheme.*

	Harmonic component	Stochastic component
Voiced frames	f0, LSF+Gain vector	LSF+Gain vector
Unvoiced frames	-	

Table 2. *HMM adopted HSM analysis scheme.*

2. DESCRIPTION OF THE HSM IMPLEMENTATION

The harmonic plus stochastic model (HSM) assumes that the speech signal can be represented as a sum of a number of harmonically related sinusoids with time-varying parameters and a noise-like component. The harmonic component is present only in the voiced speech segments, and it can be characterized at each analysis frame by the fundamental frequency and the amplitudes and phases of the harmonics. The stochastic component models all the non-sinusoidal signal components, caused by the frication, breathing noise, etc. It can be represented at each frame by the coefficients of an all-pole filter. A particular implementation of the HSM was developed at UPC in order to provide a flexible framework for all kind of signal transformations [1], especially speech synthesis and voice conversion. During the next sub-sections, we describe the speech analysis and reconstruction procedures and we discuss some questions related to the integration of the model into a HMM-based system.

2.1. Analysis

The speech signals are analyzed at a constant frame rate of 100 or 125 frames per second. Given a speech frame to be analyzed, frame number k, the fundamental frequency $F0(k)$ is estimated and a binary voicing decision is taken. If the frame is considered to be voiced, the amplitudes $A_j(k)$ and phases $\phi_j(k)$ of all the harmonics below $5KHz$ are calculated by least squares optimization. The cut-off frequency is given a fixed value because spectral envelopes are to be extracted from the harmonic component, as it will be explained later. Once the harmonic component is characterized at every analysis instant, it is interpolated and regenerated from the measured values, using 1st order polynomials for the amplitudes and 3rd order polynomials for the frequencies and phases. Then, the regenerated harmonic component is subtracted from the original signal, and the remaining part of the signal, which is considered to be the stochastic component, is LPC-analyzed at each frame. Table 1 shows the analysis structure of the harmonic plus stochastic model.

2.2. Reconstruction

The signal is reconstructed by overlapping and adding $2N$-length frames, where N is the distance between the analysis frame centres, measured in samples. Each synthetic frame contains a harmonic part, built by summing sinusoids with harmonic frequencies and constant amplitudes and phases, and a stochastic part, generated by filtering white Gaussian noise through the measured LPC-filters. A triangular window is used to overlap-add the frames in order to obtain the time-varying synthetic signal. Being k and l the frame number and the harmonic number, respectively, the following expressions are used to reconstruct the signal $s[n]$:

$$s^{(k)}[n] = \sum_l A_l^2 \cos(2\pi l f_o^{(k)} \frac{n}{f_s} + \phi^{(k)}) + \sigma[n] * h_{LPC}^{(k)}[n] \quad (1)$$

, and

$$s[kN+m] = (\frac{N-m}{N}) s^{(k)}[m] + (\frac{m}{N}) s^{(k+1)}[m-N] \quad (2)$$

where m is in the range $[0, N-1]$. The speech signals reconstructed from the parameters measured during analysis are almost indistinguishable from the original ones.

3. TRAINING HMMS ON THE HSM PARAMETERS

The problem of integrating HSM into a HMM-based speech synthesis system can be faced in two different ways:

1. Training the HMMs directly from the HSM parameters, and generating speech directly from the synthetic parameters. This strategy is problematic for several reasons concerning mainly the harmonic parameters:
   - There is a variable number of harmonics, whereas HMMs require constant length training vectors.
   - The number of harmonics is in general high, which makes the learning process more complicated.
   - The variability of the amplitudes and phases with respect to $F0$ is extremely high.
2. Training the HMMs from spectral envelopes calculated by any method, and using the HSM for reconstructing the speech signals from the synthetic envelopes. The main problem of this approach is the loss of spectral resolution caused by the spectral envelope extraction process. Nevertheless, according to our experience in voice conversion, when both the harmonic component and the stochastic component are represented by all-pole filters, the quality of the resulting synthetic speech is reasonably high.

The strategy followed in the system presented in this paper is the second one. The harmonic all-pole filters are calculated by applying the Levinson-Durbin recursion to the autocorrelation sequence given by

$$R_x[n] = \sum_l A_l^2 \cos(2\pi l f_o \frac{n}{f_s}) \quad (3)$$

Note that in this case the phase information is discarded. Before training the HMMs, the all-pole filters are transformed into their associated line spectral frequencies (LSF), which are reported to have very good properties for this kind of mathematical modeling. Table 2 shows the parameters from which the training vectors of the HMMs are built. During the synthesis process, when new parameter vectors are generated by the system, the LSF vectors are converted back into all-pole filters and multiplied by the predicted gain. The amplitudes to be used in expression (1) are calculated by sampling the harmonic envelope $H(f)$ at multiples of the generated fundamental frequency.

$$A_l^{(k)} = |H^{(k)}(l f_0^{(k)})| \quad (4)$$

The minimum phase response of the harmonic all-pole filter $H(f)$ can be also used for estimating the phases of the harmonics, but a linear phase term α has to be added in order to keep them coherent with those of the previous frame. The recursive expression proposed for the linear phase term α is based on the assumption that the pitch varies linearly from frame $k-1$ to frame k.

$$\phi_l^{(k)} = l\alpha^{(k)} + \arg\{H^{(k)}(l f_0^{(k)})\} \quad (5)$$

,

$$\alpha^{(k)} = \alpha^{(k-1)} + \pi \frac{N}{f_s} (f_0^{(k-1)} + f_0^{(k)}) \quad (6)$$

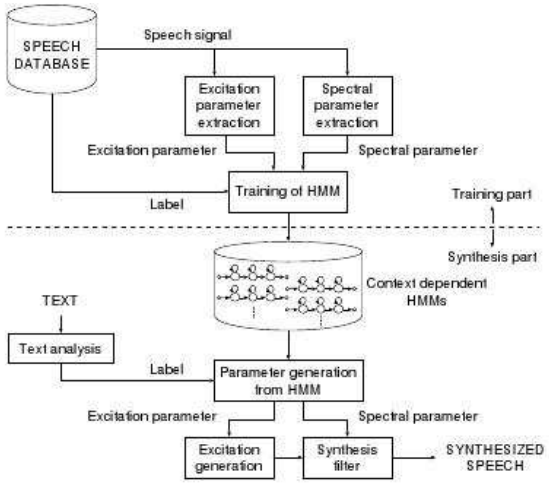


Figure 1. *Overview of a typical HMM-based speech synthesis system.*

4. STATISTICAL PARAMETRIC SYNTHESIS

4.1. Overview of a typical system

Figure 1 illustrates the block diagram of a basic HMM-based TTS system. It is composed of training and synthesis stages. In this system context dependent HMMs (phonetic,linguistic and prosodic context are taken into account) are trained from feature vectors. The feature vectors consists of spectrum (Mel-cepstral) and excitation (F_0) parts, extracted from the speech database. Each HMM has state duration probability density functions(PDFs) to model the temporal structure of speech. Accordingly, TTS models spectrum parameters excitation parameters and durations in a unified framework of HMM [10].

4.2. Training

Context dependent HMMs are trained with feature vectors which consists of spectrum and excitation.The spectrum part includes the spectral parameters and their delta and delta-delta coefficients. Excitation part consists of fundamental frequency ($\log F_0$), its delta and delta-delta coefficients. If spectrum and excitation are trained separately may occur inconsistency problems between them. The $\log F_0$ is composed of one -dimensional continuous (voiced) and zero-dimensional discrete symbol (unvoiced) values. To model such observation sequences Multi-space probability distribution (MSD) HMMS are used. The basic concept of MSD-HMM, is that they can model the sequence of observation vectors with variable dimensionality including zero-dimensional observations [11]. This special kind of HMMs are extremely useful for our work with HSM, because as explained above the harmonic part is composed of continues and discrete values, similar to $\log F_0$ (e.g., multi-dimensional for voiced, and zero-dimensional for unvoiced).

In common with most other continuous density HMM systems, HTS represents output distributions $\{b_{j(o_t)}\}$ by Gaussian Mixture Densities. However, a further generalization is made. Allows each observation vector at time t to be split into a number of S independent data streams O_{st}. The formula for computing $b_{j(o_t)}$ is then

$$b_{j(o_t)} = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} N(O_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\gamma_s} \quad (7)$$

where M_s is the number of mixture components in stream s, c_{jsm} is the weight of the m'th component and $N(;\mu,\Sigma)$ is a multivariate Gaussian with mean vector μ and covariance matrix Σ, that is

$$N(o;\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} e^{-\frac{1}{2}(o-\mu)'\Sigma^{-1}(o-\mu)} \quad (8)$$

where n is the dimensionality of o. The exponent γ_s is a stream weight. It can be used to give a particular stream more emphasis, however, it can only be set manually.

4.3. Synthesis

In the synthesis part an arbitrarily given text to be synthesized is converted to a context-base label sequence. Then according to the label sequence, a utterance HMM is constructed by concatenating context dependent HMMs. State durations of the sentence HMM are estimated maximizing the likelihood of the state duration densities. According to the duration densities that have been obtained the speech parameter generation algorithm generates the sequence of spectral and excitation parameters (voiced/unvoiced decisions) maximizing the output probabilities [12]. Finally a speech waveform is synthesized using the appropriate speech synthesis filter.

5. EXPERIMENTS AND RESULTS

The main objective of this work is to show the preliminary results of the integration of Harmonic plus Stochastic model in a HMM-based synthesis system.The different experiments depend on the structure of the spectral observation vectors, which can be split into S independent data streams weighted by a stream weight factor (7). Multiple data streams are used to enable separate modeling of multiple information.

The main goal of this specific research is to take advantage of this excellent property in combination with multi-space distribution HMMs to manage to separate as more precise as possible the Harmonic part from the stochastic part. The main problem of HSM modeling can be centralized on the voiced/unvoiced transitions where the separation of the harmonic part and stochastic part is not very precise. Using different streams to model them, in combination with MSD will resolve to a more independent modeling of each one. But while MSD utility of HTS, supports a multi-dimensional to zero dimensional variant vector, does not support multi-dimensional to multi-dimensional vector variability, which is necessary in this case.

To validate the performance of the proposed HSM method, it is compared to the results of the HTS system [6] under the same configurations and database[13]. Mel-cepstral and pitch analysis were substituted by HSM analysis, and MLSA synthesis filter was substituted by the HSM synthesis filter. As described above, for each speech frame k, to be analyzed the fundamental frequency (F_0) was estimated and a voiced/unvoiced frame decision was taken. LSF parameters were extracted for spectral modeling, and $\log F_0$ was used for excitation modeling. The feature vectors were modeled from context dependent HMMs as described for a general HMM system. At most of the experiments 14 LSF parameters were extracted for harmonic or stochastic spectral modeling plus one parameter for the Gain. Some experiments were conducted with a higher number of LSF. Excitation modeling is the same for all experiments and will not be discussed further.

According to the above, the different experiments that were held are:

1. One main vector of 93 features (HSM parameters with their derivatives, $\log F_0$ with its derivatives) was used. When a unvoiced frame is analyzed, a simple mean vector of all the harmonic parts of the voiced frames was used for the Harmonic part. The mean vector showed to perform better than a zero vector. Still some saturation on the synthesized utterances was present mainly at the voiced/unvoiced boundaries.

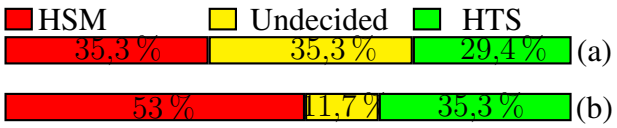


Figure 2. *Detailed results for the two parts of the perceptual test. (a) Similarity and (b) Quality.*

2. One main 30-dimensional feature vector containing static parameters only.The results as expected showed not smooth transitions at the phoneme boundaries, so high lack of natural continuity of the voice observed.
3. Two different vectors for the harmonic and the stochastic parts. The harmonic part was modeled with MSD (15 to 0 dimensions), and the stochastic part was modeled normally. The same experiment was conducted with a different Arctic database (CMU US KSP ARCTIC 0,95). An Indian-English male experienced speaker, and again the naturality of the synthetic voice of our model was significantly good.
4. Few additional experiments have been held using more poles to model spectrum envelope. These experiments kept the same structure as No 1 but 22 features are extracted for spectral modeling. Similar results were taken from this experiment so, the 14-vector size was kept to reduce the computational load and run time.

As expected best results were given by the $3'rd$ method, due to the best modeling. The performance of HSM due to different modeling approaches strengthens our starting point idea: MSD manage to better model Harmonic and stochastic parts and consequently better results are achieved. An perceptual test was given to 17 people where the same utterances were synthesized from HTS and HSM methods. The listeners have a variety of different backgrounds. Four of them are speech synthesis experts, ten listeners have speech processing background, and three listeners don't have experience in speech processing at all. Each listener evaluated 6 sentence pairs, which were presented to them in a random order. The test checks the quality of the synthesized sentences and the voice characteristics similarities to the original training voice. The listeners had to choose between five answers: "A clearly better than B", 'A a bit better than B", "i can't decide", "B clearly better than A","B a bit better than A". In the similarity test, listeners were asked to choose which of the two sentences, A or B, was more similar to the original one. Figure 2 shows the percentage of the number of times each method was preferred. The results show that the proposed method performed slightly better than the baseline HTS. Figure 2 as well shows the percentage of "i can't decide" choices, and actually at the 'similarity' test we can see that although the proposed system performs better, a high rate of the listeners couldn't distinguish the difference between the two systems.

6. CONCLUSIONS

In this work, a preliminary work to integrate an asynchronous Harmonic/Stochastic method for speech modeling, in HTS synthesis system was presented. A perceptual test was performed to compare the proposed system to the HTS system. The results show that the proposed model has good performance for speech synthesis by HMMs. As a future work we will try to use more specific configurations of the HMM-based system according to our model. Furthermore the highest attention will be given to extend the MSD property to manage to model Harmonic and stochastic part more precise. That means to be able to use MSD not only for variable feature vectors of multi-dimensional to zerodimensional but as well to multi-dimensional. We expect that by attempting this approach the performance of our model will improve.

7. ACKNOWLEDGMENT

This work was granted by the Spanish Government ref. AVIVAVOZ TEC2006-13694-C03. As well authors would like to thank the Nagoya Institute of Technology for providing Nitech-HTS synthesis system, giving us the opportunity to conduct an essential research on speech synthesis.

8. REFERENCES

[1] Erro Daniel, Moreno Asuncion, y Bonafonte Antonio, "Flexible harmonic/stochastic speech synthesis," *6th ISCA Workshop on Speech Synthesis (SSW6)*, vol. 6, pp. 194–199, Agosto 2007.

[2] Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, y Tadashi Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," *EUROSPEECH'99*, 1999.

[3] A.J. Hunt y A.W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 1, pp. 373–376 vol. 1, May 1996.

[4] Olivier Capp, "Ten years of hmms," mar 2001.

[5] Heiga Zen, Tomoki Toda, Masaru Nakamura, y Keiichi Tokuda, "Details of the nitech hmm-based speech synthesis system for the blizzard challenge 2005," *IEICE - Trans. Inf. Syst.*, vol. E90-D, no. 1, pp. 325–333, 2007.

[6] Heiga ZEN, Tomoki TODA, y Keiichi TOKUDA, "The Nitech-NAIST HMM-Based Speech Synthesis System for the Blizzard Challenge 2006," *IEICE Trans Inf Syst*, vol. E91-D, no. 6, pp. 1764–1773, 2008.

[7] J. Yamagishi, T.Ñose, H. Zen, T. Toda, y K. Tokuda, "Speaker-independent hmm-based speech synthesis system - hts-2007 system for the blizzard challenge 2007," 2007.

[8] Bennett Christina, L. y Black Alan, W., "The blizzard challenge 2006," *interspeech*, 2006.

[9] J. Yamagishi, T.Ñose, H. Zen, T. Toda, y K. Tokuda, "Performance evaluation of the speaker-independent hmm-based speech synthesis system "hts 2007"for the blizzard challenge 2007," *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 3957–3960, 31 2008-April 4 2008.

[10] Zen Heiga, Nose Takashi, Yamagishi Junichi, Sako Shinji, Masuko Takashi, Black Alan, W., y Tokuda Keiichi, "The hmm-based speech synthesis system (hts) version 2.0," *6th ISCA Workshop on Speech Synthesis (SSW6)*, vol. 6, pp. 294–299, Agosto 2007.

[11] K. Tokuda, T. Masuko, N. Miyazaki, y T. Kobayashi, "Hidden markov models based on multi-space probability distribution for pitch pattern modeling," *Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on*, vol. 1, pp. 229–232 vol.1, Mar 1999.

[12] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, y T. Kitamura, "Speech parameter generation algorithms for hmm-based speech synthesis," *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, vol. 3, pp. 1315–1318 vol.3, 2000.

[13] J. Kominek y Black Alan, W., "The cmu arctic speech databases," *SSW5, Pittsburgh, PA*, pp. 223–224, 2004.

FURTHER IMPROVEMENTS TO PRONUNCIATION BY ANALOGY

Tatyana Polyákova, Antonio Bonafonte

Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

ABSTRACT

The synthesis quality is influenced by many important factors, among which the correctness of the grapheme-to-phoneme conversion is one of the crucial ones. The globalization phenomenon makes it impossible to have a dictionary with all of the existing words for each language. Automatic letter-to-sound systems have been in the center of attention for the last decade. One of the most effective and promising methods resulted to be the so-called "pronunciation by analogy" method [8], based on the analogy in the grapheme context, allowing derivation of the correct pronunciation for a new word from the parts of similar words present in the dictionary. This paper aims at the study of this method's performance and comparison to authors' previous work, furthermore novel scoring strategies for determining the best pronunciations were proposed along with new ways of their combination. A word error rate reduction of 1.5-2.5 percent was obtained.

1. INTRODUCTION

The derivation of the pronunciation in English language given a letter string is a hard task for non-native speakers and it is even truer for automatic systems that are usually based on statistics.

The human brain handles statistics in a different way; humans use analogy to memorize how to pronounce words or word fragments in English and other languages with deep orthography.

When trying to read something, it takes time and extra effort to apply the pronunciation rules of the language, while the analogy matching that our brain performs in thunder fast. Either we say it or not correctly depend on the number of words with similar pronunciation rules that we have learned before. This is where the computer has a great advantage compared to, for example, English learners. For the computer, grasping all the examples from the dictionary and apply statistics-based analogy to derive pronunciation for the new words is a question of milliseconds. The pronunciation by analogy is an interesting technique similar to language learning that was successfully applied to derived pronunciation of out-of-vocabulary words [4,8,11].

Another important aspect of language learning is learning from errors. When a new word is pronounced erroneously a new word and corrected by a native speaker or a teacher, our brain learns not to commit the same error in a similar situation. The more examples of similar errors, given a similar error occurrence situation we have, the better we learn not to commit the same error again.

Combining these two methods used by language learners powered up by the computer CPU's learning and computing capacity we are able to improve the grapheme-to-phoneme module. The objective of this work was to compare the pronunciation-by-analogy system reported by Marchand and Damper [8]. This paper presents an interesting contribution to the research in speech synthesis due to the comparison of the grapheme-to-phoneme methods using the same dictionaries for training and testing of the systems. The possibilities of further improvement of the system's performance were explored from different perspectives. New scoring strategies were proposed and new ways to combine strategies by applying error-driven learning were studied.

2. PRONUNCIATION BY ANALOGY SYSTEM DESCRIPTION

For the first time, pronunciation-by-analogy (PbA) was proposed for reading studies by Glushko in 1979 [6] and later in 1986 Dedina and Nusbaum [4] introduced the use of this method to TTS applications. The latest and most successful implementation of the algorithm was published by Marchand and Damper [8] which we have reimplemented for our experiments. The system as well as the initial one, called PRONOUNCE [4] consists of four major components.

- Aligned lexicon (in one-to-one manner)
- Word matcher
- Pronunciation lattice (a graph that represents all possible pronunciations)
- Decision maker (chooses the best candidate among all present in the lattice)

In order to search for analogy between words that share similar substrings, in the first place it is necessary to make sure that there is a one-to-one match between the orthographic and phonetic strings, or, in other words, each letter has to be aligned to its corresponding phonetic representation. Finding the correct alignment is a challenge since the orthographic and phonetic representations of a word in English do not always have the same length. Due to its rather complex orthography, in English words there are usually more letters than sounds. In this case a null phone /_/ is inserted into the phoneme string, ex. *thing* / T _ i N _/, otherwise, if the number of phonemes is greater than that of letters, the phonemes corresponding to the same letter are joint together in one, e.g. *fox* /f A k_s/. The alignment is based on EM algorithm, and it is similar to that described in [3]. The alignment given by the system is not always the correct one and it can influence negatively on the results,

After the dictionary has been aligned in the operational phase the matcher, one of the most important components of the system, starts to search for common substrings between the input word and the rest of the dictionary entries. Before the matching starts each word in the dictionary and its pronunciation are added word beginning and end marks, for example *#thing#* #T _ i N _ #/. Every input word is then compared to all the words in the lexicon in order to find common "arcs". Let us call the substrings in the grapheme context letter arcs and the corresponding substring in the phoneme context phoneme arcs. All the possible letter arcs

with the minimum length of 2 letters and the maximum length equal to the input word length are generated and then searched for in the dictionary. For every letter arc from the input word, matching with the same letter arc from a dictionary word, the corresponding pronunciation or the phoneme arc is extracted. The frequency of appearance of each phoneme arc corresponding to the same letter arc is stored along with the start position is for each arc. As an example, we can assume that the word top is absent from our dictionary; the list of all possible letter arcs for this word can be given as *"#t, #to, #top, to, top, top#, op, op#, p#"*. Now let us suppose that in the lexicon we have the word "#topping#" with the pronunciation /# t A p _ I _ N #/, here the matcher finds the letter arcs *#t, #to, #top,* and op, with their corresponding phoneme arcs /# t/, /# t A/, /# t A p/, /A p/. Each time that for the same letter arc we find the same phoneme arc; the frequency of the phoneme arc is incremented. The matching phoneme arcs are entered into the pronunciation lattice that can be represented by nodes and connecting arcs. If an arc starts at a position *i* and ends at a position *j,* and if there is yet no arc starting or ending at position *i* , the nodes L_i and L_j are added to the graph. An arc is drawn between them. All the nodes are labeled with the corresponding "juncture" phoneme and its position in the word. The arcs are labeled with the remaining phonemes and the frequency of their appearance. An example of the lattice construction for the word top using the arcs found in the word *topping* is illustrated in Figure 1. All the arc frequencies are assumed to be equal to 1. Each complete path through the lattice is called "pronunciation candidate". We considered only the shortest paths through the lattice [8]. If there was unique shortest path, it was chosen as the best pronunciation and the algorithm stopped. In the usual case when there are several shortest paths through the lattice, it is necessary to choose the best pronunciation candidate among them. Therefore, the last but not least component of the algorithm is the decision making function.

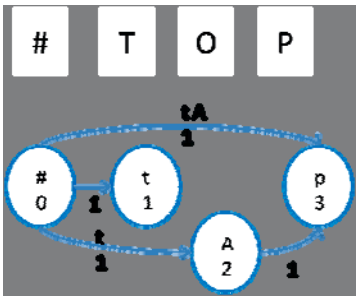


Figure 1. *Lattice construction for the word top.*

Each candidate can be represented as $C_j=\{F_j,D_j,P_j\}$, where $F_j = \{F_1,\ldots,F_n\}$ are the phoneme arc frequencies along the jth path, $D_j = \{d_1,\ldots,d_n\}$ are the arc lengths and $P_j = \{p_1,\ldots,p_l\}$ are the phonemes comprising the pronunciation candidate, being l is the pronunciation length.

Marchand and Damper in 2000 [8] proposed to use 5 scoring strategies in order to choose the best pronunciation. They will be explained with in more detail in the next section. In the same work two ways of strategy combination were introduced. Each strategy gives us a score for each candidate and based on its score each candidate is assigned a rank. According to the rank, each candidate is awarded points. If a strategy gives the same score for several candidates, they are given the same rank and the same number of points. There are two manners of determining the winner candidate; the first one is the sum rule, which chooses the candidate that has the largest value of the sum of points for all of the included strategies. The product rule chooses the candidate with the largest value of product of the points awarded by each of the included strategies. For NetTalk dictionary the best accuracy obtained was equal to 65.5% for words and 92.4% for phonemes, using all five strategies [8]. The sum and the product rule seemed to give the similar results.

3. MULTI-STRATEGY APPROACH

In our work we have extended the study of the scoring strategies implemented 6 new scoring strategies. All of the scoring strategies, the original ones and the proposed ones involve phoneme arc frequencies f_i, arc lengths d_i, and p_l, the phonemes of which the candidate consists.

The original 5 strategies [8] are:

1. Maximum arc frequency product (PF)

For each arc the corresponding arc frequencies are multiplied $PF(C_j) = \prod_{i=1}^{n} f_i$, n is the candidate length, or the number of arc of which the candidate consists. Rank 1 is given to the candidate scoring the maximum PF().

2. Minimum standard deviation of arc lengths (SDPS)

$SDPS(C_j) = \sqrt{\frac{\sum_{i=1}^{n}(d_i-\bar{d})^2}{n}}$, where $\bar{d}$ is the median arc length. Rank 1 is given to the candidate scoring the minimum SDPS().

3. Highest same pronunciation frequency (FSP)

The privilege is given to the candidates that share the same pronunciation with the others $FSP(C_j) = cand\{P_j | P_i = P_k\},\ j \neq k\ and\ k \in [1, N]$, rank 1 is given to the candidate scoring the maximum FSP().

4. Minimum number of different symbols (NDS)

This strategy gives preference to the candidates whose phonemes appear in the majority of other candidates. $DS(C_j) = \sum_{i=1}^{l}\sum_{k=1}^{N}\delta(P_{j,i}, P_{k,i})$, where l is the number of phonemes in a pronunciation, δ is the Kroneker delta, which is equal to 1 if $P_i \neq P_k$ and 0 otherwise, and N is the number of candidates, rank 1 is given to the candidate scoring the minimum NDS().

5. Weakest arc frequency (WL)

The candidate whose lowest arc frequency value is the highest $WL(C_j) = \min_i\{f_i\}$, rank 1 is given to the candidate scoring the maximum WL().

The proposed strategies are:

6. Weighted arc product frequency (WPF)

Similar to Strategy 1, but for each phoneme arc, A_k the frequency of its appearance is divided by *k*, the number of different phoneme arcs found in the dictionary for the corresponding letter arc, L_i. For example if our word, for which we are searching for the pronunciation is #infinity# and if in the pronunciation lattice we have a path that starts with a letter arc, $L_{I=}$ "# in" and a corresponding phoneme arc A_I=/# @ N/ , whose frequency is equal to 12, in order to obtain the weighted arc frequency , we have to divide 12 by the number of different phoneme arcs available in the dictionary for the

letter arc "#in". Let us say that besides A_1 we also found the following phoneme arcs: A_2 = /# I N/ and A_3= /# _ n/. Then the weighted frequency for A_1 is $WF(A_1)$= 12/3

7. Strongest first arc (SF)

The seventh strategy aims at capturing the analogy in prefixes. The candidate with the highest frequency score for the first arc is given rank 1.

8. Strongest last arc (SL)

This strategy is analogous to the previous one but for the suffixes. The candidate with the highest frequency score for the last arc is given rank 1.

9. Strongest longest arc (SLN)

The candidate who has at the same time the longest and the most frequent arc is given rank1. First the longest arc is chosen and if there is a tie the next step is to choose the most frequent one. The candidate that have the longest and arcs seem to be more reliable, and of course, the more frequent the arc is the stronger is the analogy.

10. Same symbols multiplied by arc frequency (SSPF)

The tenth strategy is similar to the fourth one (NDS), but on one hand when counting the common phonemes, we also take into consideration the phoneme arc frequencies.
For every candidate the pronunciation is compared phoneme by phoneme to other candidate pronunciations.
If a candidate has a common phoneme with other candidates, we give it a higher score, depending also on the number of times the phoneme arc containing that phoneme appears in the dictionary $SSPF(C_j) = \sum_{i=1}^{l}\sum_{k=1}^{N}\left(1-\delta(P_{j,i},P_{k,i})\right) * f_{arc(i)}$.

11. *Product frequency, same pronunciation (PFSP)*

The combination of first and third strategy, here all the candidates that share the same pronunciation obtain the same score, which is equal to the combination of the scores assigned to each one of the candidates by the first strategy $PFSP(C_j) = \sum_{\forall\, k, P_{k=P_j}} \sqrt[n]{PF(C_k)}$.

4. EXPERIMENTAL RESULTS

The experiments were performed on two dictionaries, NETtalk and LC-STAR dictionary, used by the authors in previous experiments.

The NETtalk has 20K of words, and it was manually aligned by Sejnowski and Rosenberg publicly available at [13]. The phonetic symbols used by Sejnowski and Rosenberg are left unchanged.

The LC-STAR is a public dictionary of U.S. English, created in the framework of LC-STAR project [7], we have used only the common words (about 50 K). The phone set used is SAMPA. [10]. No homonyms were considered for the experiments. As usual, 90 percent of the lexicons were used for training and 10 for test.

The first thing to do was to find out how each strategy performed. The strategy mask is a binary string, where one means the strategy is included in the final result and 0 otherwise.

The results for eleven strategies for both dictionaries are given in Table 1.

Strategy mask/ Dict	NETtalk		LC-STAR	
	Ph. acc.	W. acc.	Ph. acc.	W.acc.
10000000000	89.70%	57.48%	94.76%	73.59%
01000000000	88.00%	50.59%	92.68%	65.31%
00100000000	89.95%	59.06%	95.60%	79.34%
00010000000	90.27%	57.43%	95.53%	76.73%
00010000000	88.56%	53.75%	94.07%	71.44%
00000100000	89.69%	57.02%	94.96%	75.05%
00000010000	89.15%	55.84%	92.95%	66.17%
00000001000	87.92%	50.28%	94.46%	72.26%
00000000100	88.68%	54.01%	92.82%	65.23%
00000000010	89.99%	58.30%	94.95%	74.61%
00000000001	91.14%	**62.94%**	96.01%	**80.32%**

Table 1. *Results for each strategy for NETtalk and LC-STAR dictionaries.*

From the results above we can see that the strategies give different performance of different dictionaries. The best strategy is the proposed strategy 11 and the second best strategy is the original strategy 3 for both dictionaries. For NETtalk dictionary 2 proposed strategies and 3 original ones made it to the top five strategy list while for LC-STAR dictionary the top five strategies include 3 proposed and 2 original ones. As the next step we evaluated all possible strategy combinations, in the strategy combination mask 1 means the strategy is included in the final decision and 0 the strategy is left out.
For our implementation of the 5 original strategies the best results obtained for NETtalk lexicon were 63.04% words and 91.02% phonemes correct, given the combination of first and third strategies "10100" and 80.94% words and 96.07% phonemes correct for LC-STAR lexicon and the same strategy combination. These results are slightly different from those reported in [8], as well as the scores obtained for each original strategy with our system, but we believe that it is due to the implementation nuances. The best word accuracy obtained in [8] is 65.5% using all five strategies for NETtalk lexicon.The top five combination results are given in Tables 2 and 3.

S. combination	Ph. acc.	W. acc.
11110010011	91.28%	63.50%
01110110011	91.24%	63.40%
01100010001	91.30%	63.40%
01100010011	91.29%	63.35%
00100010001	91.31%	63.35%

Table 2. *Top five strategy combination results for NETtalk dictionary.*

As before, the top five results include proposed strategies. Eleventh strategy is present throughout Tables 2 and 3 and its contribution to improvement of overall score is the greatest for both lexicons.

The best strategy combination results obtained are higher than those previously obtained combining only the original strategies. The word error rate decreased from 36.96% to 36.5% for NETtalk and for LC-STAR from 19.06% to 18.78%. That's between 1.5 and 2.5 percent of error decrease.

S. combination	Ph. acc.	W.acc.
00101000001	96.13%	81.22%
01100001001	96.08%	81.12%
01111100001	96.11%	81.04%
01101001001	96.04%	81.04%
00101001001	96.09%	81.04%

Table 3. *Top five strategy combination results for LC-STAR dictionary.*

To further explore the possibilities of improvement for grapheme-to.-phoneme scores the transformation based learning was applied to strategy combination.

The transformation-based error-driven algorithm (TBL) originally invented by Eric Brill [2] consists in learning the transformation rules from the training data that is labeled with some initial classes. Using the TBL algorithm to correct the prediction previously obtained by another classifier allows us to capture the imperfections of previous approximations to the linguistic irregularities into a set of context-dependent transformation rules, where the context serves as the conditioning features.

In our case we took the best combination of original strategies as the initial prediction for LC-STAR dictionary to correct. The additional features were letters, phonemes and also each original strategy prediction per experiment. In order to obtain training predictions, we used n-fold evaluation, with n equal to the number of words in the dictionary. Each nth word was removed from the dictionary and input as the unknown word to the "pronunciation by analogy" system.

The best additional feature was found to be the "00100" or the third prediction standing alone, and it gave 81.46% words and 96.21% phonemes correct, using the 4-letter context and no constraints for correction. Constraints would limit the algorithm to correct the erroneous phonemes only by the ones previously seen in the training data. Using fourth and fifth predictions gave a slighter improvement up to 81.04% and 81.12% words correct correspondingly. These results should be compared to 80.94%, best result using only the original strategies. We have also used all five predictions as additional features but the improvements were not significant.

The results show that pronunciation by analogy captures very well all the regularities in English orthography, not leaving much room for improvement for the TBL method.

Comparing these results to previously obtained in [9], shown in Table 4 we can conclude that PbA is the best grapheme-to-phoneme method up to now.

Classifiers	baseline
DT	67.47%
FST	79.38%
HMM	47.54%
PbA	**80.94%**

Table 4. *Word accuracy for different grapheme-to-phoneme methods.*

The results above were obtained for the LC-STAR dictionary using decision trees (DT) [1], finite state transducers (FST) [5] and hidden Markov models (HMM) [10] and PbA classifiers.

5. CONCLUSIONS

This paper gives an overview of the pronunciation by analogy method used for g2p. New scoring strategies were proposed and the improvements were obtained based on these strategies. The 1.5-2.5% of error reduction was reached in comparison with the strategies used in [8]. The transformation-base learning algorithm was applied and the results were analyzed. New strategy combination methods were considered and slight improvements attained. The fact that applying rule-based error correction did not give important improvements allows concluding that the PbA methods is capable of capturing quite well the regularities in English orthography.

6. ACKNOWLEDGEMENTS

This work was sponsored by the Spanish Ministry of Education (*AP2005*-4526) and AVIVAVOZ project.

7. REFERENCES

[1] Black A.W., Lenzo K. and Pagel V., "Issues in building general letter to sound rules", In Proceedings of the Third ESCA workshop on speech synthesis, Jenolah Caves, W-S W, Australia, 1998

[2] Brill E., "Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging", Computational linguistics 21(4), pp. 543-565, 1995

[3] Damper R. I., Marchand Y., Marsterns J.-D. and Bazin A., "Aligning letters and phonemes for speech synthesis" in Proceedings of the 5thISCA Speech Syntesis Workshop, Pittsburgh, 209-214., 2004

[4] Dedina, M. and Nusbaum, H. "Pronounce: a program for pronunciation by analogy", Computer Speech and Language, Prentice-Hall, London, UK. 5:55—64, 1991

[5] Galescu L., J. Allen, "Bi-directional Conversion Between Graphemes and Phonemes Using a Joint N-gram Model", In Proc. of the 4th ISCA Tutorial and Research Workshop on Speech Synthesis, Perthshire, Scotland, 2001

[8] Glushko, R. J., "Principles for Pronouncing print: The psychology of phonography. Lesgold and Perfetti", 1981

[7] http://www.lcstar.org

[8] Marchand, Y. and Damper R.I. "A multi-strategy approach to improving pronunciation by analogy",Computational Linguistics26(2)pp. 195-219, 2000

[9] Polyakova T., Bonafonte A., "Learning from errors in grapheme-to-phoneme conversion", International Conference on Spoken Language Processing, Pittsburgh, USA, 2006.

[10] Taylor P., "Hidden Markov Models for grapheme to phoneme conversion", In Proc. of Interspeech 2005, Lisbon, Portugal, pp. 1973-1976, 2005

[11] Yvon, F., "Grapheme-to-phoneme conversion using multiple unbounded overlapping chunks", In Proc. NeMLaP'96, pp 218-228, 1996

[12] http://www.phon.ucl.ac.uk/home/sampa/

[13] ftp://svr-ftp.eng.cam.ac.uk/pub/comp.speech/

MODELO DE SÍNTESIS DE HABLA CON DISFLUENCIAS BASADO EN MODIFICACIONES LOCALES SOBRE FRASES CONSTITUYENTES

Jordi Adell[1]*, David Escudero-Mancebo*[2] *y Antonio Bonafonte*[1]

[1]TALP Research Center, Universitat Politècnica de Catalunya
[2]ECA-SIMM Laboratory, Universidad de Valladolid
{antonio,jadell}@gps.tsc.upc.edu[1], descuder@infor.uva.es[2]

RESUMEN

La síntesis de habla con disfluencias será necesaria en aplicaciones futuras, como puede ser el doblaje automático o la traducción voz-voz. Esta comunicación presenta un modelo para la generación de habla sintética con disfluencias, que se basa en la inserción de disfluencias dentro de locuciones fluidas. Las frases con disfluencias se generan empleando los modelos prosódicos que se emplearían para generar las frases sin disfluencias, pero añadiendo modificaciones locales que afectan sólo a las unidades adyacentes a la posición que ocupa la disfluencia. En esta comunicación se explica el modelo propuesto y su aplicación para modelar las variaciones locales relativas a la duración segmental en las fronteras de las disfluencias.

1. INTRODUCCIÓN

Hoy en día, los sistemas de síntesis de voz han alcanzado un alto nivel de naturalidad [1], principalmente debido al uso de técnicas como la síntesis basada en selección de unidades u otras tecnologías [2] que se basan en el análisis de grandes corpus de voz. Por ahora, la principal aplicación de la síntesis de voz está centrada en el habla leída, dado que se trata de un estilo muy generalista cuya extensión a otras situaciones se considera realista. Pero hoy en día, e incluso más aún en el futuro, las aplicaciones de conversión texto-voz (CTV) como por ejemplo el doblaje automático de películas, la robótica, los sistemas de diálogo o los informativos multilingües; demandan de una riqueza en los estilos muy superior.

Para integrar la voz sintética en las tecnologías enumeradas en el párrafo anterior, los sistemas CTV deben simular la manera de hablar, en lugar de simular la manera de leer, de los humanos. Ambos estilos difieren significativamente debido a la inclusión de un buen número de factores prosódicos, uno de ellos la presencia de disfluencias. Las disfluencias se definen como una interrupción en el flujo del habla que no añade ningún contenido proposicional a la frase [3]. A pesar de ello, las disfluencias ofrecen indicaciones sobre lo que se está diciendo [4] y son tremendamente frecuentes en el habla espontánea [3]. Debido a esto, su inclusión en el habla sintética parece una necesidad clara para el futuro.

El estudio de las disfluencias ha sido realizado desde diferentes perspectivas, principalmente la fonética [5], la psicolingüística [6] y el reconocimiento del habla [7]. Estas diferentes perspectivas modelan las disfluencias teniendo en cuenta sus intereses específicos. El uso de las disfluencias en sistemas CTV conlleva consideraciones adicionales que fuerzan la propuesta de un modelo alternativo. El modelo que proponemos, a diferencia de otras aproximaciones ya empleadas en sistemas CTV como [8] o [9], tiene en cuenta las frases fluidas asociadas a la frase disfluente que va a ser sintetizada, teniendo en cuenta las modificaciones locales que produce la inserción de dicha disfluencia. Estas modificaciones locales pueden afectar a la prosodia o a la cualidad de la locución original. En esta comunicación mostramos la importancia de estas modificaciones locales, comprobando el impacto en la duración de las sílabas que rodean las disfluencias.

Primero hacemos una introducción del modelo de generación de disfluencias. Después, se presenta el procedimiento a seguir para aplicar el modelo, mostrando las alteraciones en la duración de las sílabas que rodean la disfluencia. Por último se plantea el trabajo futuro a realizar para completar este trabajo y las conclusiones.

Trabajo parcialmente financiado por los proyectos de investigación AVIVO del Gobierno de España (TEC2006-13694-C03) y por el proyecto ACME de la Junta de Castilla y León (VA077A08)

2. INSERCIÓN DE DISFLUENCIAS EN SISTEMAS DE SÍNTESIS POR SELECCIÓN DE UNIDADES

La síntesis de disfluencias en el marco de la síntesis por selección de unidades presenta una serie de dificultades a tener en cuenta. Primero, la mayoría de los sistemas de selección de unidades que existen tienen un inventario cerrado de unidades que no contiene en absoluto disfluencias. Esto hace que los métodos de aprendizaje automático que se aplican para analizar la prosodia, no sean capaces de modelar automáticamente este fenómeno a partir de los datos. Además, no sólo los modelos prosódicos sino también los modelos de análisis del texto, como por ejemplo el etiquetado de *Part-of-Speech*, dependen mucho de que las frases de entrada tengan una estructura que se corresponda con una sintaxis correcta y un orden, en términos de acentos y de grupos de entonación, también correcto; lo cual no sucede en una frase con disfluencias, ya que cuando la fluidez de una frase se rompe, su estructura también se rompe. Por último, el habla sintética con disfluencias precisa del uso de nuevas unidades segmentales que no están definidas en las bases de datos convencionales, como pueden ser los fillers o los fonemas interrumpidos.

Nuestro modelo distingue tres elementos de cara a generar una frase dada que incluya disfluencias (*Disfluent Sentence* (DS)). Primero, la frase original que iba a ser pronunciada antes de que apareciera la disfluencia (*Original Sentence* (OS)). Después la frase objetivo (*Target Sentence* (TS)) que hubiera sido dicha si no hubiera habido ningún motivo que provocara la disfluencia. Tercero, el *Editing Term* (ET), que de acuerdo a la terminología defendida en [10] es la clave o indicador de la disfluencia (por ejemplo, el relleno de la pausa). Podemos ilustrar estos términos con un ejemplo tomado de [11]: *Go from left to mmm from pink again to blue* donde los elementos de la disfluencia se identifican como:

$$\textit{Go RM\{from left to\}} \overset{\text{IP}}{\downarrow} \textit{ET\{mmm\}, RR\{from pink again to\} blue.}$$

siendo RM (*Reparandum*), RR (*Repair*), ET (*Editing Term*) e IP (*Interruption Point*) los términos empleados en [10]. La frase del ejemplo (DS) está relacionada con las frases: *Go from left to right* y *Go from pink again to blue* que son las frases OS y TS respectivamente; y el ET es *mmm*. Estas relaciones se modelan como sigue:

$$\begin{aligned} DS &= PrevRM|RM|ET|RR|PostRR \\ OS &= PrevRM|RM|PostRM \\ TS &= PrevRM|RR|PostRR \end{aligned}$$

donde $PrevRM$ es la parte de la frase que precede el RM; $PostRM$ es la parte que sigue al RM y $PostRR$ es la parte que sigue al RR. El $PostRM$ existe sólo en OS, porque no se trata de una parte real de ninguna locución, sino que en su lugar; en la DS, se pronuncia RR. En el ejemplo presentado arriba, $PostRM$ sería $right$. Dado una DS, un sistema CTV de selección de unidades puede generar correctamente las correspondientes OS y TS. Si el inventario de unidades del CTV incluye un cierto número de frases con disfluencias, entonces el sistema CTV podrá elegir entre un conjunto de ETs. Además, existen evidencias de que la inclusión de disfluencias provoca modificaciones locales en las propiedades acústicas de los términos RM, $PostRM$ y $PrevRM$ [12, 9].

Nuestra propuesta de generación de disfluencias opera en tres etapas (Figura 1). Primero, utiliza OS para obtener los parámetros prosódicos relativos a $PrevRM$ y RM. También genera TS para obtener los parámetros prosódicos de RR y $PostRR$. Estos parámetros prosódicos se utilizan para guiar la búsqueda en el inventario de síntesis. En una segunda etapa, se obtiene el término ET desde el inventario. Finalmente, se aplican modificaciones locales a las sílabas adyacentes al término ET. Estas modificaciones se corresponden con las desviaciones locales que pueden aparecer en las fronteras de los elementos descritos en esta sección ($PrevRM$, RM, ...).

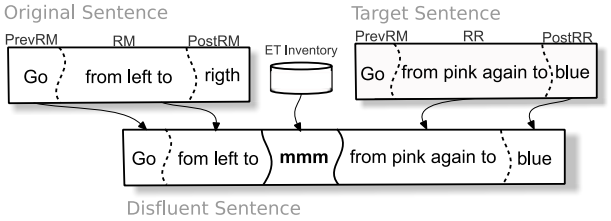


Figura 1: *Proceso de generación de habla con disfluencias aplicado a una frase de ejemplo*

El ritmo es una de las variables prosódicas que puede ser desviada con respecto a los valores predichos para las locuciones sin disfluencias. En la siguiente sección, vamos a mostrar como el ritmo de las frases con disfluencias sigue una tendencia general que es similar a la seguida por las frases sin disfluencias y que sufre una desviación significativa en las unidades próximas al ET. En este trabajo se consideran tres tipos de disfluencias: alargamientos, repeticiones y pausas rellenas. La tabla 1 describe los elementos de modelado para estos tres casos de estudio.

Tabla 1: *Elementos del modelo para diferentes tipos de disfluencias.* 1^a *y* 2^a *indican cada una de las realizaciones en una repetición.*

type	RM	ET	RR
alargamientos	$\emptyset$	$\emptyset$	$\emptyset$
pausas rellenas	$\emptyset$	filler (e.g. mmm)	$\emptyset$
repeticiones	1^a	$\emptyset$	2^a

3. MODIFICACIONES LOCALES DEL RITMO

En la literatura se encuentran dos categorías principales a la hora de describir el ritmo de las distintas lenguas: *Accent-timed* y *syllable-timed*. Ambas categorías están relacionadas con el principio de isocronía por el cual, las unidades del lenguaje tienden a estar equiespaciadas en el tiempo [13]. El español, al igual que otras lenguas procedentes del latín, se considera *syllable-timed* [14, 15]. Dado que en este trabajo vamos a centrarnos en el caso del español, parece apropiado medir el ritmo de las unidades suprasegmentales (como por ejemplo las frases) como la duración media de las sílabas en dichas unidades. Sirva la tabla 2 como referencia de los valores de duración media de las sílabas medidas para todo el corpus.

Tabla 2: *Duración media de las sílabas y límites de los intervalos de confianza al 99 %*

	mean	lower bound	upper bound
no-acentuadas	105ms	102ms	108ms
acentuadas	136ms	132ms	140ms
pre-Pausal	222ms	210ms	236ms
todas	123ms	120ms	125ms

El corpus empleado en este trabajo es una selección de frases del corpus desarrollado para el proyecto europeo LCSTAR. Se grabó en un laboratorio y recoge diálogos de dos personas a las que se pidió que completaran una determinada tarea por teléfono. La comunicación fue semi-duplex de manera que la base de datos está grabada en base a los turnos de intervención [16]. Aunque es habla de laboratorio, es espontánea porque los locutores no tenían ninguna guía en sus intervenciones. Los hablantes pronuncian las disfluencias de manera natural y además las disfluencias son muy frecuentes porque necesitaban planificar los turnos de intervención a la vez que realizaban sus tareas. Se han utilizado 100 frases de cuatro hablantes diferentes (3 hombres y 1 mujer). En total las disfluencias que se han analizado son: 133 pausas rellenas, 71 repeticiones y 65 alargamientos. La segmentación fonética se realizó automáticamente y fue corregida manualmente.

Se computan una serie de medidas para cada elemento descrito en la sección 2 (Tabla 3). D^{-1}_{syl} es la última sílaba de $PrevRM$ y D^{1}_{syl} es la primera sílaba de $PostRR$. Esperamos encontrar que el ritmo permanece constante a lo largo de las frases y que los cambios se producen en las fronteras de ET de acuerdo al modelo propuesto.

Tabla 3: *Lista de variables relacionadas con el ritmo.*

Variable	Definición
R_s	Duración media de las sílabas de una frase.
R^N_w	Duración media de las sílabas en la N-ésima palabra desde la disfluencia.
R_{DF}	Duración media de las sílabas en la disfluencia, o una parte de ella: $R_{hes}, R_{RM}, R_{RR}, R_f$
D^N_{syl}	Duración de la N-ésima sílaba desde la disfluencia.

El estudio se basa en diagramas tipo *boxplot* [17] que mostrarán las desviaciones en las fronteras de los ET. Para facilitar la interpretación, las variables se trazan en el orden en el que aparecen en las frases. Hemos utilizado el test estadístico conocido como *Least Significant Difference* (LSD) o *t-test* múltiple para comparar las medias de las distribuciones y así establecer diferencias [18]. Todos los niveles de significatividad presentados en esta comunicación son al 99 % de confianza.

Los alargamientos (en inglés *hesitations*) consisten en una extensión no prevista de una sílaba. En la figura 2 se observa que

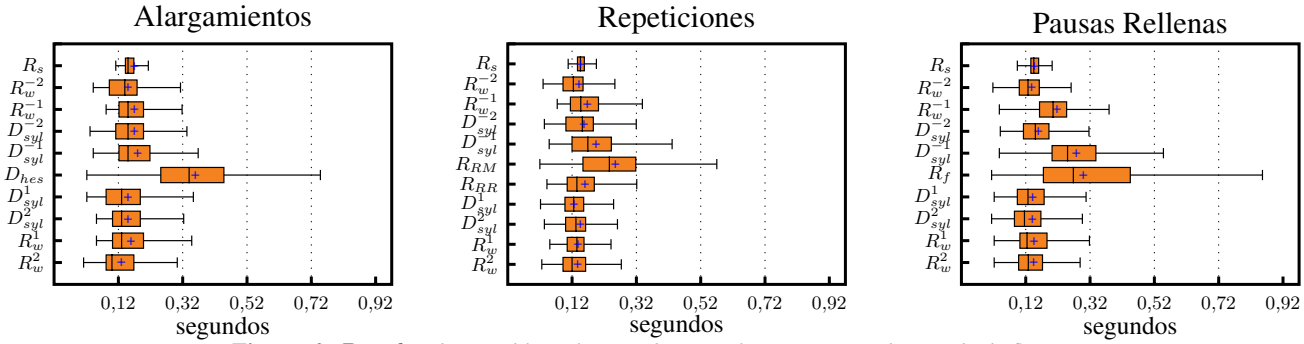


Figura 2: *Boxplot de variables relacionadas con el ritmo para cada tipo de disfluencia*

no hay una diferencia significativa entre el ritmo en las palabras que preceden al alargamiento (R_w^{-1} y R_w^{-2}) frente a R_s. Tampoco lo hay entre las palabras que siguen al alargamiento (R_w^1 y R_w^2). Sin embargo, la sílaba alargada (i.e. D_{hes}) tiene mayor rango y mayor valor medio. El test *LSD* muestra que la única diferencia significativa existente es la que existe entre D_{hes} y el resto de variables. Los intervalos del test *LSD* indican que el alargamiento de la sílaba está entre 141ms y 239ms con respecto a la duración media de las sílabas en la frase. Podemos resumir las modificaciones locales en el caso de los alargamientos como el alargamiento de la sílaba extendida, mientras que el resto de variables permanecen aparentemente inalteradas.

Las repeticiones se componen de dos elementos principales: las locuciones repetidas que corresponden a RM y RR. La figura 2 muestra que mientras R_{RR} es parecido a R_s, R_{RM} y D_{syl}^{-1} son más largos que R_s. Estas observaciones coinciden con las realizadas en trabajos previos [12, 19]. El test LSD muestra que existe una diferencia significativa entre los valores medios de R_{RM} y de D_{syl}^{-1} con respecto al resto de propiedades. Podemos concluir que para el caso de las repeticiones, RM parece pronunciarse más despacio que el resto de la frase y que la última sílaba de $PrevRM$ (D_{syl}^{-1}) es alargada sistemáticamente. LSD también indica que el valor medio de D_{syl}^{-1} es entre 10ms y 88ms más largo que el de R_s. También R_{RM} es entre 70ms y 148ms más largo que la duración media de las sílabas en la frase. Sólo el 17 % de las muestras contienen un silencio en ese punto. Si quitamos estos ejemplos de los datos, la diferencia sigue siendo significativa.

Las pausas rellenas (*filled pauses*) no pueden ser consideradas estrictamente como sílabas. Sin embargo, la duración del relleno (R_f) puede ser comparado con la duración de una sílaba. Puede verse que todas las medidas, excepto R_f, D_{syl}^{-1} y R_w^{-1}, siguen una distribución similar a la que sigue R_s (figura 2). Este hecho también se apoya en la observación de que el test LSD indica que no hay diferencia significativa con respecto a los valores medios del resto de variables. Según el test LSD, la sílaba pre-filler (D_{syl}^{-1}) –a diferencia de la sílaba pre-pausa– es entre 100ms y 161ms más larga que R_s. La duración de la pausa rellena tiene un valor medio de 300ms y una desviación estándar de 185ms lo que implica que la longitud de la pausa rellena supera claramente el rango de variación de las sílabas pre-pausales (ver Table 2). De nuevo, el ritmo de la frase global no se modifica y sólo se observan alteraciones locales en forma de alargamientos previos a la sílaba previa al relleno.

Para cada tipo de disfluencia analizada, se han identificado los elementos clave que las definen, observando una serie de modificaciones locales a la disfluencia que deberán ser reproducidas en la síntesis. Estas modificaciones deberán ser aplicadas una vez que las partes correspondientes de OS y TS han sido sintetizadas y ensambladas.

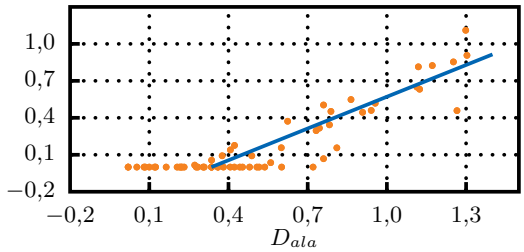
En experiencias previas, hemos construido para ello sistemas de reglas [20] o modelos de regresión [21]. En la figura 3 mostramos los modelos de regresión para el caso de los alargamientos. Estudios previos han mostrado que la duración percibida de un alargamiento es la compuesta por la duración de la sílaba más la duración del silencio posterior ($D_{ala} = D_{sil} + D_{syl}$) [9]. A partir de las observaciones del corpus, podemos inferir que la sílaba alargada solo puede serlo hasta un máximo y que sólo a partir de ese momento aparece un silencio. Para alargamientos superiores a este valor podemos modelarlos mediante reglas de regresión. Finalmente, el modelo propuesto puede expresarse como:

$$D_{sil} = \begin{cases} 0 & if D_{ala} <= 0{,}337 \\ 0{,}819 * D_{ala} - 0{,}174 & if D_{ala} > 0{,}337 \end{cases} \quad (1)$$

$$D_{syl} = \begin{cases} D_{ala} & if D_{ala} <= 0{,}337 \\ 0{,}181 * D_{ala} + 0{,}174 & if D_{ala} > 0{,}337 \end{cases} \quad (2)$$

El modelo propuesto ha sido implementado en nuestro sistema de CTV [22]. Se han realizado tests preceptuales informales que nos permiten afirmar que las modificaciones locales de las duraciones segmentales hacen que la inclusión de disfluencias no provoque una degradación significativa de la naturalidad general del sistema. También se ha observado que si no se aplican estas variaciones el resultado es menos natural.

(a) *Duración del silencio*



(b) *Duración de la sílaba*

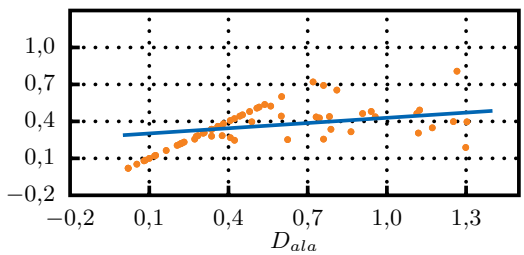


Figura 3: *Lineas de regresión de la duración de la sílaba y el silencio posterior en un alargamiento*

4. CONCLUSIONES Y TRABAJO FUTURO

En esta comunicación se ha presentado un modelo para la generación de disfluencias en sistemas CTV que se apoya en la consideración de tres frases constitutivas: La frase con disfluencias (DS), la frase original (OS) y la frase objetivo (TS). OS y TS proporcionan el contexto donde el *Reparandum* y el *Repair* respectivamente son pronunciados correctamente. El término ET se inserta en la frase final, utilizando la base de datos de síntesis y aplicando una serie de modificaciones locales a las sílabas adyacentes. La aportación principal es que este modelo puede ser aplicado en síntesis de voz utilizando los modelos prosódicos previamente entrenados con voz sin disfluencias.

Para estudiar las modificaciones locales en el ritmo, hemos considerado tres tipos de disfluencias: alargamientos, repeticiones y pausas rellenas. Para estos tres tipos de disfluencias, hemos comprobado que existen variaciones significativas con respecto al ritmo general de la frase, pero que dichas variaciones afectan exclusivamente al propio ET y a la parte final de *Reparandum*. Estos resultados apoyan la oportunidad del modelo propuesto.

Actualmente se están estudiando las variaciones locales provocadas en los contornos de $F0$. En el futuro no se descarta estudiar otros aspectos como la calidad de la voz o la energía. Nuestro propósito es el estudio de las variaciones de un conjunto de parámetros prosódicos que nos permita sintetizar las disfluencias, aprovechando los modelos previamente entrenados para voz sin disfluencias, aplicando a posteriori modificaciones locales que afecten sólo al entorno de la disfluencia.

Por otro lado, otro problema que debe ser tratado, se refiere a la predicción de la posición que deben ocupar las disfluencias. En este artículo sólo hemos abordado la cuestión de cómo sintetizar las disfluencias, y no hemos considerado donde incluir dichas disfluencias. Ésta es una tarea compleja que está fuera del ámbito de nuestra investigación por el momento. Otro aspecto importante a tener en cuenta es la elección de un $PostRM$ adecuado para las repeticiones. A pesar de la importancia de estos dos aspectos, hay que tener en cuenta que en aplicaciones de generación de voz en sistemas de diálogo, esta información puede ser aportada al sistema CTV en base a una serie de reglas. También hay otras aplicaciones como la traducción voz-voz donde la posición de las disfluencias puede venir dada de acuerdo a la locución que debe ser traducida.

5. BIBLIOGRAFÍA

[1] A. Aaron, E. Eide, , y J.F. Pitrelli, "Conversational computers," *Scientific American*, vol. 292, no. 6, pp. 64–69, June 2005.

[2] Mark Fraser y Simon King, "The blizzard challenge 2007," in *Proceedings of the Blizzard Challenge*, 2007.

[3] Jea E. Fox Tree, "The effects of false starts and repetitions on the processing of subsequent words in spontaneous speech," *Journal of Memory and Language*, vol. 34, no. 6, pp. 709–738, December 1995.

[4] Michiko Watanabe, Keikichi Hirose, Yasuharu Den, y Nobuaki Minematsu, "Filled pauses as cues to the complexity of following phrases," in *Proc. of Eurospeech*, September 2005, pp. 37–40, Lisbon, Portugal.

[5] Shu-Chuan Tseng, *Grammar, Prosody and Speech Disfluencies in Spoken Dialogues.*, Ph.D. thesis, Dpt. Linguistics and Literature, University of Bielefeld, April 1999.

[6] Herbert H. Clark, "Speaking in time," *Speech Communication*, vol. 36, no. 1-2, pp. 5–13, January 2002.

[7] Masataka Goto, Katunobu Itou, y Storu Hayamizu, "A real-time filled pauses detection system for spontaneous speech recognition," in *Porc. of EUROSPEECH*, Budapest, Hungary, 1999, pp. 227–230.

[8] Shiva Sundaram y Shrikanth Narayanan, "An empirical text transformation method for spontaneous speech synthesizers," in *Proc. of EUROSPEECH*, Geneva, Switzerland, September 2003, pp. 1221–1224.

[9] R. Carlson, K. Gustafson, y E. Strangert, "Cues for hesitation in speech synthesis," in *Proceedings of Interspeech 06*, Pittsburgh, USA, 2006.

[10] Elizabeth Ellen Shriberg, *Preliminaries to a Theory of Speech Disfluencies*, Ph.D. thesis, Berkeley's University of California, 1994.

[11] William Levelt y A. Cutler, "Prosodic marking in speech repair'," *Journal of Semantics*, pp. 205–217, 1983.

[12] Elizabeth E. Shriberg, "Acoustic properties of disfluent repetitions," in *Proc. of International Conference on Phonetic Sciences (ICPhS)*, 1995, vol. 4, pp. 384–387, Stockholm, Sweden.

[13] Manuel Almeida, "Organización temporal del español: el principio de isocronía," *Revista de Filología Románica*, vol. 1, no. 14, pp. 29–40, 1997, Madrid.

[14] Guillermo Andrés Toledo, *El ritmo en el español : estudio fonético con base computacional*, Number 361 in Biblioteca románica hispánica ; II. Estudios y ensayos. Gredos, 1988.

[15] Mar Carrió y Antonio Ríos, "Compensatory shortening in spanish spontaneous speech," in *Proceedings of ESCA Workshop on Phonetic and Phonology of Speaking Styles.*, September 1991, vol. 16, pp. 1–5, Barcelona, Spain.

[16] David Conejero, Jesús Giménez, Victoria Arranz, Antonio Bonafonte, Neus Pascual, Núria Castell, y Asunción Moreno, "Lexica and corpora for speech-to-speech translation a trilingual approach.," in *Proc .of Eurospeech*, September 2003.

[17] Michael Frigge, David C. Hoaglin, y Boris Iglewicz, "Some implementations of the boxplot," *The American Statistician*, vol. 43, no. 1, pp. 50–54, February 1989.

[18] D.J. Saville, "Multiple comparison procedures: The practical solution," *The American Staticions*, vol. 44, no. 2, pp. 174–180, May 1990.

[19] Jordi Adell, Antonio Bonafonte, y David Escudero, "Disfluent speech analysis and synthesis: a preliminary approach," in *in Proc. of 3th International Conference on Speech Prosody*, May 2006, Dresden, Germany.

[20] Jordi Adell, Antonio Bonafonte, y David Escudero, "Filled pauses in speech synthesis: towards conversational speech.," *10th International Conference on Text, Speech and Dialogue (LNAI)*, vol. 1, pp. 358–365, September 2007, Springer Verlag.

[21] Jordi Adell, Antonio Bonafonte, y David Escudero, "Statistical analysis of filled pauses' rhythm for disfluent speech synthesis," in *Proc. of the 6th IWSS*, Bonn, Germany, August 2007.

[22] Antonio Bonafonte, Pablo Daniel Agüero, Jordi Adell, Javier Pérez, y Asunción Moreno, "Ogmios: The UPC Text-to-Speech synthesis system for spoken translation," in *TC-STAR Workshop on Speech-to-Speech Translation*, June 2006.

NUEVO MÓDULO DE ANÁLISIS PROSÓDICO DEL CONVERSOR TEXTO-VOZ MULTILINGÜE DE TELEFÓNICA I+D

M. Á. Rodríguez[1], J. G. Escalada[1], A. Armenta[1] y J. M. Garrido[2]

[1]División de Tecnología del Habla
Telefónica Investigación y Desarrollo
Emilio Vargas, 6, 28043 Madrid
Via Augusta, 177, 08021 Barcelona

[2]Departament de Traducció i Filologia, UPF/
Barcelona Media Centre d'Innovació
Ocata, 1, 08003 Barcelona

RESUMEN

Este artículo presenta un nuevo módulo de análisis prosódico para un conversor texto-voz (CTV) que se ocupa de predecir y caracterizar los límites prosódicos en la lectura de un texto. Los límites tratados son tanto pausas como frases entonativas, y se emplean para mejorar la generación de otros parámetros prosódicos (duración de los sonidos y contorno de F0) y aumentar la naturalidad de la voz sintética. El funcionamiento del módulo de análisis prosódico no sólo tiene en cuenta características lingüísticas generales propias de un idioma determinado, sino que también se adapta al modo particular de hablar de un locutor humano de referencia.

1. INTRODUCCIÓN

En este artículo se presenta una evolución del módulo de inserción de pausas (módulo estructurador-pausador) del CTV multilingüe de Telefónica I+D [1], orientada a mejorar tanto la naturalidad y corrección de las pausas generadas, como la generación del resto de parámetros prosódicos. El antiguo módulo pausador sólo consideraba límites de grupo fónico (pausas) que se decidían por regla sobre el resultado de una agrupación en sintagmas. Las mejoras se han centrado fundamentalmente en dos aspectos: por un lado, la detección de pausas se ha enriquecido con la inserción de otro tipo de límites prosódicos, con lo que ya no cabe hablar tanto de un módulo pausador sino de un módulo de análisis prosódico, encargado de identificar en los textos de entrada unidades prosódicas de distinto ámbito; y por otro, se ha intentado modelar la variación estilística interlocutor que se da en la segmentación prosódica de los enunciados.

Empieza a ser un hecho generalmente aceptado que los enunciados de las lenguas presentan una estructura prosódica más compleja que la simple segmentación en grupos fónicos o entonativos. Desde hace tiempo se han propuesto otras unidades prosódicas de nivel inferior, como el grupo acentual [2, 3], el grupo tónico [4] o la frase intermedia [5, 6]. Estas unidades formarían toda una estructura jerárquica que determinaría, entre otros aspectos, la asignación de las pausas y de los contornos entonativos [7, 8, 9]. Este tipo de aproximación jerárquica está presente ya en algunas implementaciones de analizadores prosódicos para conversión texto-voz [10, 11].

De acuerdo con esta idea, en el módulo de análisis prosódico del CTV de Telefónica se han distinguido dos tipos distintos de límites prosódicos: límites de nivel 1, que se corresponderían con los límites de grupo fónico o entonativo en la terminología lingüística tradicional, y que se realizan en el nivel fonético con la inserción de una pausa de una duración específica y un movimiento de F0 determinado; y límites de nivel 2, de ámbito inferior al anterior, más o menos equivalentes a las frases intermedias del modelo autosegmental (aquí proponemos la denominación frase entonativa), y que se manifiestan fonéticamente con un movimiento específico de F0, pero sin inserción de pausa.

La asignación de límites prosódicos de nivel 1 y 2 se realiza teniendo en cuenta tres factores: organización sintáctica de los enunciados (*chunks*); información sobre el número de sílabas desde el último límite; e información sobre la velocidad de elocución. La organización sintáctica y el número de sílabas ya se empleaban en el antiguo módulo pausador (si bien de una manera mucho menos elaborada que en la versión actual); la velocidad de elocución se ha incorporado en esta nueva versión.

Por otro lado, el antiguo módulo pausador no admitía variación interlocutor en el pausado. Todos los locutores sintéticos de un mismo idioma pausaban exactamente igual el mismo texto. En el nuevo módulo de análisis prosódico se ha intentado recoger la variación interlocutor, mediante la creación de modelos específicos para cada locutor sintético. Los datos sobre la variación estilística propia del locutor para la construcción de sus modelos se extraen del análisis prosódico del corpus grabado por cada locutor de referencia. La creación de modelos para la segmentación prosódica a partir del análisis automático

de datos reales no es nueva en conversión texto-voz ([12], por ejemplo), aunque en este caso se ha aplicado a la creación de modelos específicos para cada locutor.

Este nuevo procedimiento de predicción de límites prosódicos se comenzó a aplicar a los locutores desarrollados para español castellano, pero luego se ha aplicado también a los locutores del resto de los idiomas incorporados en el CTV multilingüe de Telefónica.

2. ESTRUCTURA Y FUNCIONAMIENTO DEL MÓDULO DE ANÁLISIS PROSÓDICO

En la Figura 1 se incluye un diagrama que ilustra el proceso completo que sigue el módulo de análisis prosódico.

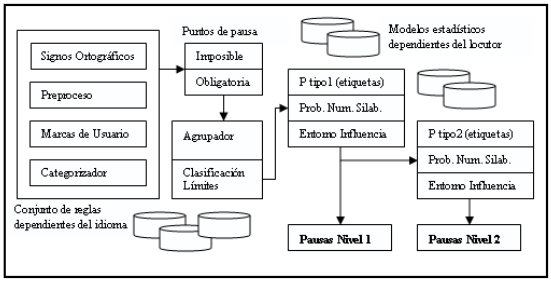


Figura 1. *Diagrama del módulo de análisis prosódico*

De forma simplificada, el proceso de segmentación del texto de entrada en unidades prosódicas implica dos fases:

- La identificación y etiquetado de los puntos en el texto que son candidatos a recibir una marca de límite (límites potenciales). Entre ellos están los signos ortográficos (se considera que no siempre implican pausa).
- La selección de los límites definitivos entre los límites potenciales.

Para la selección definitiva de los límites se utiliza, además del conjunto de límites potenciales ya etiquetados y caracterizados, una serie de datos sobre el comportamiento prosódico de cada locutor de referencia, extraídos del análisis automático del corpus grabado. Estos datos son los que permiten obtener la variación interlocutor en el proceso de pausado automático.

En los siguientes apartados se describen con más detalles estos dos procesos.

2.1. Detección y clasificación de los límites potenciales

Para la localización de los límites potenciales se lleva a cabo en el CTV un análisis morfológico del texto, como paso previo a la agrupación de palabras en sintagmas *(chunking)*. Estas tareas se realizan en los módulos categorizador y agrupador. El primero es el encargado de asignar una categoría a las palabras del texto, y de determinar sus propiedades morfológicas (género, número, etc.), y el segundo se ocupa de agrupar las palabras categorizadas en algo parecido a sintagmas (pseudosintagmas). Este análisis se lleva a cabo mediante un sistema de reglas y de diccionarios específicos para cada idioma.

El siguiente paso es caracterizar los límites potenciales con una etiqueta que se empleará para el proceso de selección de los límites definitivos. Al igual que en el caso del análisis morfosintáctico, esta tarea la realiza un conjunto de reglas y diccionarios específicos de cada idioma, que en función del contexto sintáctico asignan una etiqueta a cada límite potencial identificado. Se han definido más de 170 etiquetas diferentes para el etiquetado de los límites, que intentan reflejar, fundamentalmente, el contexto sintáctico del límite potencial.

2.2. Selección de los límites definitivos

El proceso de selección de límites se ha organizado en dos fases secuenciales. En la primera fase se deciden los límites definitivos de nivel 1 entre todos los límites potenciales. En la segunda fase, se seleccionan los límites definitivos de nivel 2 entre los límites potenciales restantes.

Ambas fases funcionan de una forma semejante, si bien emplean datos diferentes durante su proceso.

2.2.1. Selección de límites definitivos de nivel 1

Los datos del locutor humano de referencia que se emplean en esta fase son:

• Probabilidades de realizar límite de nivel 1 para cada tipo (etiqueta) de límite. Reflejan el hecho de que las distintas etiquetas de los límites potenciales no tienen la misma probabilidad de inducir un límite prosódico. Estas probabilidades se obtienen encontrando el número de veces que cada etiqueta ha sido realizada por el locutor de referencia como límite de nivel 1, y dividiendo por el número de ocurrencias de esa etiqueta en el corpus de grabación.

• Probabilidad acumulada por número de sílabas transcurridas. Esta tabla de valores refleja el hecho de que a medida que aumenta el número de sílabas desde un límite de nivel 1, aumenta la probabilidad de introducir un nuevo límite de nivel 1. Para su cálculo, primero se obtiene la probabilidad de encontrar en el corpus grabado grupos fónicos (delimitados por límites de nivel 1) de determinado número de sílabas (una, dos, tres...). Una vez asociadas las probabilidades a cada número de sílabas, se acumulan para obtener una especie de función de densidad de probabilidad (probabilidad de que un grupo fónico tenga x sílabas o menos). Al hacer este cálculo no se tienen en cuenta los límites asociados a signos ortográficos, pues se considera que su realización como límites definitivos no está tan condicionada por la longitud de los grupos fónicos.

Respecto a las probabilidades asociadas a cada etiqueta de límite, es bien conocido que la bondad de determinada etiqueta para ser escogida como límite definitivo no depende únicamente de su identidad sino también del contexto en el que se encuentra. El volumen de la combinatoria entre distintos códigos de pausa hace inviable obtener valores de probabilidad para todos los casos de combinaciones. Por esta razón se ha buscado una aproximación alternativa que permita tener en cuenta la influencia del contexto en la realización de un límite como definitivo. Esta alternativa consiste en modificar los valores de probabilidad inicial asociados a cada etiqueta, resultando unas probabilidades incentivadas (o penalizadas, si el incentivo resulta negativo). Para cada límite potencial se localizan los límites anterior y siguiente cuyas probabilidades iniciales sean mayores que las del límite dado. Esto define un entorno de influencia del código de pausa. Cuanto más extenso sea el entorno de influencia, más preferible será la pausa. El incentivo (o penalización) de probabilidad se calcula a partir de los datos de probabilidades acumuladas por número de sílabas transcurridas.

Una vez obtenidas las probabilidades incentivadas correspondientes a cada límite, se pasa a la etapa final del procedimiento de selección, que consiste en un algoritmo de programación dinámica tipo Viterbi que obtiene la secuencia óptima de valores seleccionado / no-seleccionado correspondiente a la secuencia de límites potenciales de una frase de texto. En cada punto del camino (cada límite) se obtienen dos valores de suma de probabilidad: uno correspondiente al caso de seleccionar ese límite, y otro correspondiente al caso de no seleccionarlo. En el punto final del camino se escoge la alternativa de mayor suma de probabilidades acumulada, se va siguiendo de qué alternativa del límite anterior procede (seleccionado / no-seleccionado), y se reconstruye hacia atrás el camino óptimo.

2.2.2. Selección de límites definitivos de nivel 2

La selección de límites de nivel 2 sigue el mismo procedimiento descrito anteriormente, teniendo en cuenta que se parte de los límites de nivel 1 ya seleccionados. En este caso, los datos que se emplean procedentes del análisis del corpus de grabaciones son los siguientes:

• Probabilidades de realizar límite de nivel 2. Se calculan encontrando el número de veces que cada etiqueta ha sido realizada por el locutor de referencia como límite de nivel 2, y dividiendo por el número de ocurrencias de esa etiqueta en el corpus de grabación, sin tener en cuenta los casos en que esa etiqueta se realizó como límite de nivel 1.

• Probabilidad acumulada por número de sílabas transcurridas. Para su cálculo, primero se obtiene la probabilidad de encontrar en el corpus grabado frases entonativas (limitadas por la izquierda por un límite de nivel 1 o de nivel 2, y limitadas por la derecha por un límite de nivel 2) de determinado número de sílabas (una, dos, tres,...). Como en el caso de los límites de nivel 1, una vez asociadas las probabilidades a cada número de sílabas, se acumulan para obtener una especie de función de densidad de probabilidad.

3. EVALUACIÓN

El verdadero valor de la predicción y caracterización de límites prosódicos está en la mejora que aporta en la corrección y naturalidad en la lectura de textos por parte del CTV. Desde este punto de vista, la mejor forma de evaluación sería mediante pruebas subjetivas con ejemplos de voz sintetizada, que tendrían que ser escuchadas y evaluadas por un número suficiente de personas para expresar y cuantificar sus preferencias. Estas pruebas son complejas y costosas. Además, los cambios introducidos en nuestro sistema han llevado aparejados otros cambios que afectan a los módulos generadores de otros parámetros prosódicos (duración y F0), por lo que la evaluación subjetiva no permitiría discernir el efecto individual del nuevo módulo de análisis prosódico. Por último, cuando se habla de estructura prosódica, el concepto de corrección es algo que, en gran medida, está por definir, y seguramente también sometido a variación estilística.

Por ello, hemos optado por realizar una evaluación de tipo objetivo. Asumiendo como referencia de corrección la segmentación prosódica realizada por los locutores de referencia en la lectura del corpus. Esta evaluación tiene sus limitaciones, pero nos permite comprobar si el módulo de análisis prosódico consigue sus objetivos: realizar una correcta predicción de límites prosódicos, y reflejar las peculiaridades de un hablante determinado en esta tarea. Hemos tomado como referencia las grabaciones realizadas por dos locutores masculinos en español castellano (identificados como JOSÉ y NACHO), que presentan una forma bastante diferente de realizar límites prosódicos al leer textos.

Se han escogido 10 oraciones con suficiente longitud como para inducir un buen número de límites internos de nivel 1 (pausas) y de nivel 2 (frases entonativas), y con una estructura compleja y variada. El conjunto de esas frases contiene un total de 292 lugares en los que se podría introducir un límite (espacios entre palabras), descontando las pausas finales correspondientes a cada oración.

Para comparar el funcionamiento del módulo de análisis prosódico ajustado a cada uno de los dos locutores de referencia, presentamos las tablas 1 y 2, que recogen los siguientes datos: lugares en los que tanto el locutor de referencia como su correspondiente locutor sintético coinciden en no hacer ningún límite (CNLI); coincidencia total (lugar y nivel) en hacer un límite (CTOT); coincidencia parcial (coincide el lugar pero no el nivel) en hacer un límite (CPAR); límites realizados por el locutor pero no predichos por el sistema (NPRE); y límites predichos por el sistema pero no realizados por el locutor (falsas predicciones, FPRE).

CNLI	CTOT	CPAR	NPRE	FPRE	Total
197 casos	37 casos	16 casos	33 casos	9 casos	292 casos
67,47%	12,67%	5,48%	11,3%	3,08%	100%

Tabla 1. *Comparación de la asignación de límites del locutor sintético JOSÉ con su correspondiente locutor humano de referencia teniendo en cuenta todas las posibilidades de hacer límite en los textos*

CNLI	CTOT	CPAR	NPRE	FPRE	Total
219 casos	39 casos	7 casos	16 casos	11 casos	292 casos
75%	13,36%	2,4%	5,48%	3,77%	100%

Tabla 2. *Comparación de la asignación de límites del locutor sintético NACHO con su correspondiente locutor humano de referencia teniendo en cuenta todas las posibilidades de hacer límite en los textos*

CNLI y CTOT recogen el funcionamiento estrictamente correcto (JOSÉ: 80,14%, NACHO: 88.36%). CPAR recoge un margen añadido de funcionamiento aceptable (JOSÉ: 5,48%, NACHO: 2,4%). NPRE y FPRE se pueden considerar errores de funcionamiento, aunque no se refieren a límites objetivamente mal ignorados o mal predichos por el sistema. La lectura resultante puede ser correcta o al menos aceptable para el texto de entrada, pero los hemos considerado errores de funcionamiento en tanto en que se apartan de lo que hizo el locutor de referencia

La diferencia de comportamiento entre los dos locutores de referencia frente a los mismos textos aparece en la tabla 3, donde se recogen los límites que realizaron: de nivel 1, distinguiendo los relacionados con signos ortográficos (N1OR) y los no relacionados (N1NO), y los de nivel 2 (NIV2). También se muestran las coincidencias totales (lugar y nivel) entre los dos locutores (CTOT). Se puede ver que el número total de límites realizados por uno y otro cambia bastante, y que el grado de coincidencia es relativamente bajo, excepto en el caso de las pausas relacionadas con signos ortográficos.

	N1OR	N1NO	NIV2	Total
JOSÉ	18	15	53	86
NACHO	20	13	29	62
CTOT	18	8	17	43

Tabla 3. *Diferencias de comportamiento entre los locutores de referencia JOSÉ y NACHO*

4. CONCLUSIONES

En este trabajo se ha presentado un nuevo módulo de análisis prosódico para el CTV de Telefónica. Además de mejorar la corrección del pausado automático con respecto a la versión anterior, el principal objetivo ha sido hacer la tarea de segmentación prosódica automática dependiente del locutor, con la idea de obtener resultados distintos para un mismo texto en función del locutor sintético que se utilice para leerlo.

Los resultados de la evaluación llevada a cabo muestran un grado aceptable de funcionamiento del sistema, adaptado a distintos locutores. Además, en una evaluación subjetiva informal de la voz generada por el CTV, la voz sintética resulta más natural y, en particular, menos monótona y predecible en su entonación. También se ha podido apreciar que la inserción de límites con distintos locutores sobre el mismo texto no coincide en muchos casos.

10. BIBLIOGRAFÍA

[1] M. Á Rodríguez, J. G. Escalada y D. Torre, "Conversor texto.voz multilingüe para español, catalán, gallego y euskera", Procesamiento del Lenguaje Natural, Revista nº 23 SEPLN, pp. 16-23, 1998.

[2] N. Thorsen, "Interpreting Raw Fundamental-Frequency Tracings of Danish", Phonetica, 36, pp. 57-78, 1979.

[3] N. Thorsen, "Intonation contours and stress group patterns in declarative sentences of varying length in ASC Danish", ARIPUC 14, pp. 1-29, 1980.

[4] T, Navarro, *Manual de entonación española*, Guadarrama, Madrid. 1944.

[5] M. Beckman y J.B. Pierrehuumbert, "Intonational structure in English and Japanese", Phonology Yearbook, 3, pp. 255-310, 1986.

[6] P. Prieto, "Phonological phrasing in Spanish" en: Optimality-Theoretic Advances in Spanish Phonology, ed. by Sonia Colina and Fernando Martínez-Gil, pp. 39-60. John Benjamins, Amsterdam/Philadelphia, 2006.

[7] E. O Selkirk*, Phonology and Syntax*: *The relation between Sound and Structure*, Cambridge, MA, The MIT Press, 1984.

[8] M. Nespor y I. Vogel,. *Prosodic Phonology*, Dordrecht, Foris, Studies in Generative Grammar, 28, 1986.

[9] D.R. Ladd, "Intonational phrasing: the case for recursive prosodic structure", Phonology Yearbook, 3, 311-340, 1986.

[10] B. Gili Fivela y S. Quazza, "A Prosodic Parser for an Italian Text-to-Speech System", Actas del XII Congreso de la SEPLN, Sevilla, septiembre de 1996. Procesamiento del Lenguaje Natural, Revista 19: pp. 189-200, 1996.

[11] B. Gili Fivela y S. Quazza, "Text-to-prosody parsing in an Italian synthesizer", in Proceedings of the 5th European Conference On Speech Communication and Technology (EuroSpeech), pp. 987-990, 1997.

[12] J. Hirschberg. y P. Prieto, "Training intonational phrasing rules automatically for English and Spanish text-to-speech", Speech Communication 18,3, pp. 283-292, 1996.

SESIÓN DE POSTER 2

ADQUISICIÓN DE UN CORPUS DE DIÁLOGOS PARA UN DOMINIO DE RESERVAS DE INSTALACIONES DEPORTIVAS

E. Segarra, M.J. Castro, I. Galiano, F. García, J. A. Gómez, D. Griol, L.F. Hurtado, E. Sanchis, F. Torres, F. Zamora

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València, 46022 València, Spain
esegarra@dsic.upv.es

RESUMEN

La adquisición de un corpus de diálogos hablados es un proceso complejo y costoso. Con el objetivo de facilitar este proceso proponemos un método para llevar a cabo la adquisición de un corpus de diálogos para una tarea de requerimientos a un sistema de información; en particular, en este trabajo el usuario interacciona con un sistema de información de reservas de pistas deportivas. En este trabajo presentamos una descripción de los diferentes componentes del sistema de adquisición.

1. INTRODUCCIÓN

El desarrollo de sistemas de diálogo hablado ha recibido un fuerte impulso en los últimos años en el marco de las tecnologías del lenguaje hablado. El desarrollo de estos sistemas es un proceso complejo que comporta el diseño, implementación y evaluación de varios módulos en los que intervienen diversas fuentes de conocimiento. Actualmente, una de las aproximaciones adoptadas con más éxito es la basada en modelos estadísticos, que representan de forma probabilística los diferentes procesos implicados en los módulos y en la que los correspondientes modelos se estiman a partir de corpus de diálogos persona-máquina [1, 2, 3, 4]. El éxito de las aproximaciones estadísticas depende principalmente de la calidad de los modelos y, por tanto, de los corpus a partir de los cuales son entrenados. Es por ello que la adquisición de corpus adecuados y la definición de una adecuada representación semántica para su etiquetado son procesos claves.

La adquisición de un corpus puede resultar un proceso complejo y costoso por lo que es útil cualquier ayuda que permita facilitar este proceso. Con este objetivo en la adquisición de un corpus de diálogos para la tarea EDECAN-SPORT de reservas de pistas deportivas en el marco del proyecto Edecán [5] hemos seguido el proceso que describimos a continuación.

Trabajo parcialmente subvencionado por el gobierno español y los fondos FEDER con el proyecto TIN2005-08660-C04-02 , y por el Programa de Apoyo a la Investigación y Desarrollo PAID-05-08 de la Universitat Politècnica de València.

En primer lugar, hemos analizado un conjunto de diálogos persona-persona facilitados por el servicio de reservas de pistas deportivas de nuestra universidad, que constituye el dominio de trabajo de la tarea EDECAN-SPORT. A partir de estos diálogos se ha definido la semántica de la tarea en términos de frames de los turnos de usuario y de los turnos de sistema, y se han etiquetado estos diálogos iniciales. Con ello disponemos de un corpus inicial muy reducido para la tarea EDECAN-SPORT. A partir de este pequeño corpus etiquetado se ha obtenido una versión preliminar de un gestor de diálogo que se estima a partir de datos [3]. Este gestor de diálogo se utiliza en el proceso de adquisición de un corpus más amplio de esta tarea con la ayuda de la técnica del Mago de Oz (las adquisiciones se llevan a cabo con usuarios reales y un sistema de diálogo simulado en parte).

En segundo lugar, como el corpus inicial no tiene la suficiente tamaño para entrenar modelos adecuados para el módulo de comprensión del habla, no disponemos de una versión preliminar de este módulo para el proceso de adquisición con el Mago de Oz. Nuestra propuesta consiste en el uso de un Mago de Oz específico para el módulo de comprensión diferente del Mago de Oz que supervisa el gestor de diálogo. La idea de usar más de un Mago de Oz fue propuesta por [6].

Para que este Mago específico de comprensión actúe de la forma más parecida a un sistema automático, va acompañado de un proceso que realiza una simulación de errores de reconocimiento y comprensión. Esta simulación [7] está basada en el análisis de los errores de reconocimiento y comprensión generados cuando nuestros módulos de reconocimiento y comprensión son aplicados a un corpus adquirido en anteriores proyectos [8], también para una tarea de consulta a un sistema de información.

En este trabajo presentamos una descripción de los diferentes componentes del sistema de adquisición para la tarea EDECAN-SPORT. En la sección 2 describimos el corpus inicial, y en la seccióón 3 la representación semántica tanto de los turnos de usuario como de sistema. En la sección 4 describimos la arquitectura global del sistema de adquisición y en la sección 5 los datos globales del corpus que se está adquiriendo.

S1: Bienvenido al servicio de reservas de instalaciones deportivas. ¿En qué puedo ayudarle?
U1: Quiero reservar una pista de pádel para el próximo miércoles.
S2: ¿A qué hora quiere jugar?
U2: A las tres de la tarde.
S3: El día 8 de diciembre a las 15:00 horas están disponibles las pistas de pádel número 1 y 4. ¿Cuál quiere reservar?
U3: La pista de pádel número 1.
S4: Le acabo de reservar la pista de pádel número 1 para el día 8 de diciembre a las 15:00 horas. ¿Quiere algo más?
U4: Sí, quiero cancelar una pista para mañana por la tarde.
S5: ¿Quiere cancelar la reserva de la pista de tenis número 3 del día 7 de diciembre a las 19:00?
U5: Sí, por favor.
S6: Acabo de cancelar la pista señalada. ¿Quiere algo más?
U6: No, gracias.
S7: Gracias por usar este servicio de reservas. Adiós.

Figura 1. *Un ejemplo de diálogo para la tarea EDECAN-SPORT.*

2. DIÁLOGOS PERSONA-PERSONA

Se dispone de un pequeño conjunto de diálogos persona-persona reales de la aplicación de reservas de pistas deportivas, gracias a la colaboración del personal del servicio de deportes de la universidad. Estos diálogos han facilitado la definición del dominio semántico de la tarea EDECAN-SPORT: consultas de horarios disponibles, consultas de reservas realizadas anteriormente, peticiones de reservas de pistas deportivas y, finalmente, peticiones de cancelaciones de reservas anteriores. Se ha obtenido un total de 150 diálogos que comprenden 837 turnos de usuario. La Figura 1 muestra un ejemplo de estos diálogos.

3. DEFINICIÓN DE LA SEMÁNTICA DE LA TAREA EDECAN-SPORT

La definición de la semántica de la tarea se llevó a cabo a partir de los anteriores diálogos y teniendo en cuenta las diferentes prestaciones del servicio real ofrecido por la universidad.

3.1. Representación semántica de los turnos de usuario

Los actos de diálogo de usuario representan la interpretación semántica de los turnos de usuario en términos de frames (conceptos y atributos). Para la tarea EDECAN-SPORT hemos identificado seis conceptos:

- Cuatro conceptos dependientes de la tarea y relacionados con los diferentes tipos de interacción del usuario con el sistema (requerimientos al sistema de información): disponibilidad de pistas *Availability*, reserva de pistas *Booking*, consulta de pistas reservadas *Booked*, y cancelación de reservas *Cancellation*.
- Dos conceptos independientes de la tarea: *Acceptance* y *Rejection*.

Han sido identificados un total de seis atributos. Están relacionados con la información que el usuario debe proporcionar al sistema para completar los requerimientos al sistema de información. Estos atributos son: *Sport*, *Hour*, *Date*, *Court-Type*, *Court-Number*, and *Order-Number*.

A continuación se muestra un ejemplo de la representación semántica de un turno de usuario:

Turno de usuario: *Quiero reservar una pista de tenis para mañana por la tarde*

Representación semántica: (Booking)
Sport: tenis
Date: mañana
Hour: tarde

3.2. Representación semántica de los turnos de sistema

Los turnos de sistema han sido también etiquetados. Los conceptos se han dividido también en dependientes e independientes de la tarea. Se ha definido un conjunto de 15 conceptos dependientes de la tarea:

- Cuatro conceptos principales que informan al usuario acerca del resultado de un requerimiento: acerca de la disponibilidad de pistas *Availability*, acerca de la reserva de pistas *Booking*, acerca de las reservas actuales *Booked* y acerca de la cancelación de reservas *Cancellation*.
- Cuatro conceptos para pedir al usuario los valores de atributos necesarios para llevar a cabo la consulta al sistema de información: *Sport*, *Date*, *Hour*, and *Court-Type*.
- Cuatro conceptos para confirmar los cuatro conceptos principales y un concepto *Confirmation* para confirmar los diferentes valores de atributos.
- Otros conceptos: violación de las normas de reserva del servicio de la universidad *Rule-Info* y la petición de selección de una de las pistas disponibles *Booking-Choice*.

Han sido identificados seis atributos: cinco relacionados con atributos del turno de usuario (*Sport*, *Court-Type*, *Court-Number*, *Date*, *Hour*), y un atributo relacionado

con el número de pistas que satisfacen el requerimiento del usuario (*Availability-Number*). Por otra parte, ha sido identificado un conjunto de seis conceptos independientes de la tarea: *Opening*, *Closing*, *Non-Understood*, *New-Query*, *Acceptance* y *Rejection*. A contiuación se presenta un ejemplo de etiquetado de turno de sistema:

Turno de sistema: *¿Quiere reservar la pista de squash número 1 en el pabellón para el 25 de junio de 20:00 a 20:30 horas?*

Representación semántica: (Confirmation-Booking)
Sport: squash
Date: 25-06-2008
Hour: 20:00-20:30
Court-Type: pabellón
Court-Number:1

4. LA ARQUITECTURA PARA LA ADQUISICIÓN DEL CORPUS PARA LA TAREA EDECAN-SPORT

Siguiendo las principales contribuciones de la literatura en el área de los sistemas de diálogo hablado, realizamos la adquisición del corpus para la tarea EDECAN-SPORT usando la técnica del Mago de Oz. En esta técnica, una persona sustituye al sistema de diálogo en la mayor parte de sus funciones, principalmente: escucha al locutor, construye el requerimiento al sistema de información y construye el frame de sistema como respuesta al locutor. Un generador de lenguaje natural construye la respuesta al locutor en base a este frame. Finalmente, se genera el audio correspondiente a esta frase en lenguaje natural.

Usualmente estas tareas son realizadas por una única persona. Nuestra propuesta [7] para esta adquisición estriba en trabajar con dos Magos de Oz: un simulador del proceso de comprensión del habla y un simulador del gestor de diálogo. El primero escucha al locutor y simula la acción de los módulos de reconocimiento y comprensión del habla y proporciona la representación semántica del turno de usuario (el frame/s correspondiente/s). A partir de dicho frame, el segundo Mago actúa como un gestor de diálogo. La arquitectura que se propone se muestra en la Figura 2.

Además de simplificar las funciones del Mago de Oz, la separación de la comprensión y del gestor de diálogo permite simular con más fidelidad lo que ocurre en un sistema totalmente automático. El gestor de diálogo no escucha al locutor, sino que recibe la información proporcionada únicamente por el módulo de comprensión. Con ello conseguimos similares condiciones experimentales en el proceso de adquisición que en el proceso de evaluación y prueba del sistema de diálogo.

4.1. El simulador de comprensión

El Mago de Oz de comprensión, utilizando el llamado editor de comprensión, traduce la intervención del locutor en su correspondiente frame (que nos va a servir como frame de referencia en el etiquetado de los turnos de usuario del corpus). A partir de este frame, el simulador de errores introduce ciertos errores en el frame de referencia generando el frame simulado, que será el que se proporcionará el segundo Mago. Este proceso se muestra en la Figura 3. El simulador de errores reproduce el comportamiento de nuestros módulos de reconocimiento y comprensión desarrollados en el marco del proyecto DIHANA [8].

4.2. El simulador del gestor de diálogo

Hemos desarrollado una aproximación para la gestión del diálogo haciendo uso de un modelo estadístico que se estima a partir de corpus [3] en el marco del proyecto DIHANA [8]. En esta aproximación, la respuesta del gestor en un momento del diálogo se selecciona en base a un proceso de clasificación que tiene en cuenta la información suministrada por el usuario a lo largo del diálogo y la última intervención del mismo. Este modelo de gestor se estima automáticamente a partir de un corpus de diálogos de entrenamiento etiquetado en términos de actos de diálogo. Este gestor almacena la información proporcionada por el usuario en una estructura de datos que continene también medidas de confianza para cada uno de los elementos de esta estructura (conceptos y atributos).

Se ha adaptado esta aproximación para su uso en el marco del proyecto EDECÁN, en particular para la tarea de reservas de instalaciones deportivas EDECÁN-SPORT [9]. Esta adaptación tiene en cuenta los nuevos requerimientos introducidos por esta tarea, lo que incluye el uso de un manejador de aplicación que interacciona con la base de datos y que verifica si el requerimiento del usuario cumple con las normas definidas para el uso del servicio de reservas. Como ya se ha comentado en la introducción, a partir de los diálogos persona-persona etiquetados en términos de frames se ha estimado una versión preliminar de gestor de diálogo para la tarea EDECÁN-SPORT.

Esta versión está implementada en el sistema de adquisición, y la misión del Mago de Oz de diálogo consiste en supervisar su funcionamento. Esta supervisión se lleva a cabo a través de dos aplicaciones, en una se supervisa (se corrige cuando el sistema automático ha fallado) el funcionamiento del gestor de diálogo y en la otra aplicación se supervisa el funcionamiento del manejador de la aplicación.

5. LA ADQUISICIÓN

Se están adquiriendo 240 diálogos en los que intervienen 18 locutores (9 hombres y 9 mujeres) de diferentes procedencias geográficas (las 4 sedes de los equipos investigadores del consorcio EDECÁN). Las lenguas implicadas en la adquisición son tres: castellano, catalán y eusquera. Se ha definido un conjunto de 15 tipos de escenarios. A continuación se muestra un escenario.

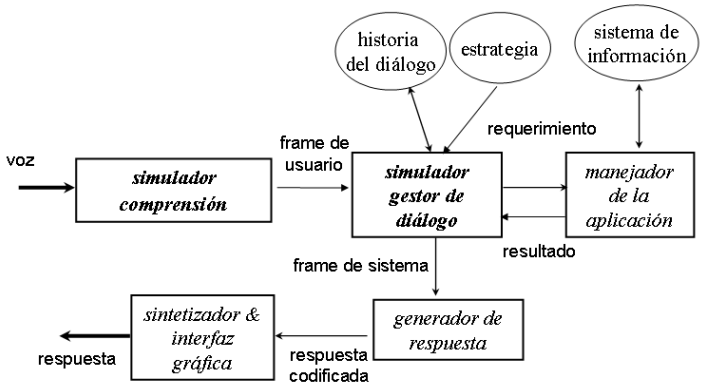


Figura 2. Esquema de la adquisición.

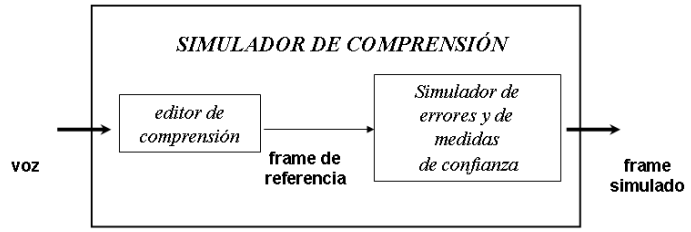


Figura 3. Simulador de comprensión.

Objective: Conocer la disponibilidad y reservar.
Sport: Padel.
Date: miércoles, jueves o viernes.
Hour: tarde.

6. BIBLIOGRAFÍA

[1] A. Potamianos, E. Ammicht, y E. Fosler-Lussier, "Modality tracking in the Multimodal Bell Labs Communicator," in *Proc. of ASRU 03*, St. Thomas, U.S. Virgin Islands, 2003, pp. 192–197.

[2] F. Torres, L.F. Hurtado, F. García, E. Sanchis, y E. Segarra, "Error handling in a stochastic dialog system through confidence measures," *Speech Communication*, vol. 45, pp. 211–229, 2005.

[3] L.F. Hurtado, D. Griol, E. Segarra, y E. Sanchis, "A Stochastic Approach for Dialog Management based on Neural Networks," in *Procs. of InterSpeech'06*, Pittsburgh, USA, 2006, pp. 49–52.

[4] J. Williams y S. Young, "Partially Observable Markov Decision Processes for Spoken Dialog Systems," in *Computer Speech and Language 21(2)*, 2007, pp. 393–422.

[5] Eduardo Lleida, Encarna Segarra, M. Inés Torres, y J. Macías-Guarasa, "EDECÁN: sistEma de Diálogo multidominio con adaptación al contExto aCústico y de AplicaciónN," in *IV Jornadas en Tecnologia del Habla*, Zaragoza, Spain, 2006, pp. 291–296.

[6] Daniel Salber y Joëlle Coutaz, "Applying the wizard of oz technique to the study of multimodal systems," in *EWHCI '93: 3rd International Conference on Human-Computer Interaction*, London, UK, 1993, pp. 219–230, Springer-Verlag.

[7] F. Garcia, L.F. Hurtado, D. Griol, M. Castro, E. Segarra, y E. Sanchis, "Recognition and Understanding Simulation for a Spoken Dialog Corpus Acquisition," in *TSD 2007*, vol. 4629 of *LNAI*, pp. 574–581. Springer, 2007.

[8] José-Miguel Benedí, Eduardo Lleida, Amparo Varona, María-José Castro, Isabel Galiano, Raquel Justo, Iñigo López de Letona, y Antonio Miguel, "Design and acquisition of a telephone spontaneous speech dialogue corpus in Spanish: DIHANA," in *Proceedings of LREC 2006*, Genoa (Italy), May 2006, pp. 1636–1639.

[9] Lluís F. Hurtado, David Griol, Encarna Segarra, y Emilio Sanchis, "Adapting a Statistical Dialog for a New Domain," in *Proc. of DECALOG'07*, Rovereto (Italy), 2007, pp. 171–172.

ADQUISICIÓN Y EVALUACIÓN DE UN CORPUS DE DIÁLOGOS MEDIANTE UNA TÉCNICA DE GENERACIÓN AUTOMÁTICA DE DIÁLOGOS

David Griol, Lluís F. Hurtado, Emilio Sanchis, Encarna Segarra

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València. E-46022 València, Spain
{dgriol,lhurtado,esanchis,esegarra}@dsic.upv.es

RESUMEN

En este trabajo presentamos una aproximación para adquirir un corpus de diálogos mediante la interacción de un simulador de usuario y un simulador de gestor de diálogo. Inicialmente se define una selección aleatoria de respuestas para el funcionamiento de ambos módulos, evaluándose automáticamente el diálogo adquirido mediante la definición de un conjunto de condiciones de parada. Las probabilidades de las respuestas seleccionadas tras simular con éxito un diálogo se incrementan previamente a una nueva simulación. De este modo, es posible obtener un modelo de diálogo sin la necesidad de disponer de un corpus de diálogos para la tarea. En el artículo se resumen los resultados de la aplicación de esta metodología para la adquisición de un corpus de diálogos para el proyecto EDECÁN.

1. INTRODUCCIÓN

El aprendizaje de modelos estadísticos que permitan desarrollar los diferentes módulos de un sistema de diálogo ha despertado durante la última década el interés de la comunidad científica [1] [2]. Aunque en la literatura pueden encontrarse modelos para el diseño de gestores de diálogo basados en la definición por parte de un experto de un conjunto de reglas, durante los últimos años se han desarrollado aproximaciones basadas en el aprendizaje de un modelo estadístico que define el comportamiento del gestor de diálogo [3] [4] [5].

En este campo, hemos desarrollado una aproximación para gestionar el diálogo utilizando un modelo estadístico aprendido a partir de un corpus de diálogos [6]. Recientemente se ha llevado a cabo la adaptación de este modelo para desarrollar un gestor de diálogo en el marco del proyecto EDECÁN [7].

El éxito de las aproximaciones estadísticas depende del tamaño y la calidad del corpus de diálogos utilizado para realizar el aprendizaje del modelo. La adquisición y etiquetado de un corpus de diálogos con un número suficiente de diálogos para entrenar un buen modelo requiere un esfuerzo considerable. Una solución para este problema consiste en el desarrollo de un módulo que simule las respuestas del usuario. En este campo se han desarrollado durante los últimos años diferentes técnicas para modelizar el comportamiento del usuario [8] [9] [10] [11].

En este artículo, presentamos una aproximación para adquirir un corpus de diálogos mediante la interacción de un simulador de usuarios y un simulador de gestor de diálogo. La aproximación propuesta se basa en la selección aleatoria de las respuestas del usuario y del sistema. Los únicos parámetros que se requieren para la adquisición son la definición de la semántica de la tarea (es decir, el conjunto de posibles actos de diálogo de usuario y de sistema) y un conjunto de condiciones que permitan descartar automáticamente los diálogos que no alcanzan el objetivo definido. Hemos utilizado esta técnica para adquirir un corpus para una de las tareas definidas en el proyecto EDECÁN. La tarea EDECÁN-UPV consiste en el diseño de un interfaz oral para reservar y proporcionar información sobre las instalaciones deportivas en nuestra universidad.

El corpus obtenido mediante la técnica de simulación automática se ha evaluado utilizándolo para el entrenamiento de nuestro gestor de diálogo estadístico. El gestor de diálogo aprendido se ha evaluado utilizando un conjunto de diálogos persona-persona proporcionados por el personal del Área de Deportes de la Universidad Politécnica de Valencia. Este corpus está compuesto por 150 diálogos (873 turnos de usuario). De este modo, en estos diálogos han participado usuarios que deseaban realmente realizar las diferentes consultas que proporcionará el sistema de forma automática.

2. DEFINICIÓN DE LA SEMÁNTICA DE LA TAREA EDECAN-UPV

De forma similar a la utilizada en muchos otros sistemas de diálogo, la notación seleccionada en la tarea EDECAN-UPV para la representación de los turnos de usuario y de sistema se basa en la utilización de actos de diálogo.

Este trabajo se ha desarrollado en el marco del proyecto EDECÁN subvencionado por el MEC y FEDER número TIN2005-08660-C04-02.

2.1. Actos de diálogo de usuario

Los actos de diálogo de usuario se representan mediante la notación clásica de frames (atributos y conceptos). Para la tarea EDECÁN se han definido un total de siete conceptos:

- Cuatro conceptos dependientes de la tarea, correspondientes a los tipos de consultas que puede solicitar el usuario: conocer la disponibilidad de pistas (*Availability*), realizar una reserva (*Booking*), saber las reservas que tiene vigentes (*Booked*) o cancelar alguna de ellas (*Cancellation*).

- Tres conceptos independientes de la tarea: *Acceptance*, *Rejection* y *Not-Understood*.

Se han definido un total de seis atributos, relativos a la información que debe aportar el usuario para completar las diferentes consultas contempladas por el sistema. Los atributos definidos son el deporte que se desea practicar (*Sport*), el horario para el que se desea la consulta (*Hour*), la fecha (*Date*), el tipo de pista polideportiva (*Court-Type*), el identificador de pista (*Court-ID*) y el número de orden correspondiente a la pista que se desea reservar (*Order-Number*).

A continuación se muestra un ejemplo de la interpretación semántica de una intervención del usuario:

Turno de usuario: *Quiero reservar una pista de squash para mañana por la tarde.*
Interpretación semántica:
(Booking)
Sport: squash
Date: mañana
Hour: tarde

2.2. Actos de diálogo de sistema

El etiquetado de los turnos de sistema se ha realizado de forma similar al de las intervenciones del usuario. Los conceptos definidos pueden clasificarse igualmente en dependientes de la tarea e independientes de la misma. Se han detallado un total de 18 conceptos dependientes de la tarea:

- Conceptos utilizados para informar al usuario del resultado de una determinada consulta: sobre disponibilidad de pistas (*Availability*), sobre la realización de una reserva (*Booking*), sobre las reservas actuales del usuario (*Booked*) o sobre la anulación de una reserva (*Cancellation*).

- Conceptos definidos para requerir al usuario los atributos necesarios para una determinada consulta: deporte (*Sport*), fecha (*Date*), hora (*Hour*) y tipo de pista (*Court-Type*).

- Conceptos utilizados para la confirmación de conceptos (*Confirmation-Availability*, *Confirmation-Booking*, *Confirmation-Booked*, *Confirmation-Cancellation*) y de atributos (*Confirmation-Sport*, *Confirmation-Date*, *Confirmation-Hour*, *Confirmation-Court-Type*).

- Conceptos relativos al gestor de aplicación: infracción de la normativa de reservas (*Rule-Info*) o para indicar la necesidad de seleccionar alguna de las pistas disponibles (*Booking-Choice*).

Se han definido un total de seis atributos, correspondientes a los cinco detallados para el etiquetado de los turnos de usuario (*Sport*, *Court-Type*, *Court-ID*, *Date* y *Hour*) y un atributo relativo al número de pistas que satisfacen los requerimientos del usuario (*Num-Courts*).

Seguidamente se muestra un ejemplo del etiquetado de una respuesta del sistema:

Turno de Sistema: *¿Le reservo la pista de squash 1 del pabellón para el 25 de julio de 20:00 a 20:30?*
Etiquetado:
(Confirmation-Booking)
Sport: squash
Date: 25-07-2008
Hour: 20:00-20:30
Court-Type: pabellón
Court-ID:1

3. TÉCNICA PARA LA ADQUISICIÓN AUTOMÁTICA DE DIÁLOGOS

Como se ha comentado en la introducción, nuestra aproximación para la adquisición de un corpus de diálogo se basa en la interacción de un módulo simulador de usuario y un simulador de gestor de diálogo. Ambos módulos realizan una selección aleatoria de una de las posibles respuestas definidas para la semántica de la tarea (actos de diálogos de usuario y de sistema). Al principio de la adquisición, el conjunto de respuestas de sistema se define como equiprobable. Cada vez que se simula un diálogo con éxito, las probabilidades de las respuestas seleccionadas por el gestor de diálogo durante dicho diálogo se incrementan antes de simular un nuevo diálogo.

El simulador de usuarios proporciona conceptos y atributos que representan la intención del turno de usuario. De este modo, el simulador lleva a cabo las funciones de los módulos de reconocimiento automático del habla y de comprensión del lenguaje.

La semántica seleccionada para el gestor de diálogo se corresponde con las 25 posibles respuestas definidas para el sistema en la tarea. La selección de las posibles respuestas de usuario se llevó a cabo teniendo en cuenta la semántica definida para para el módulo de comprensión.

Adicionalmente, se ha desarrollado un módulo que realiza la generación de errores y la incorporación de medidas de confianza. Esta información modifica los frames

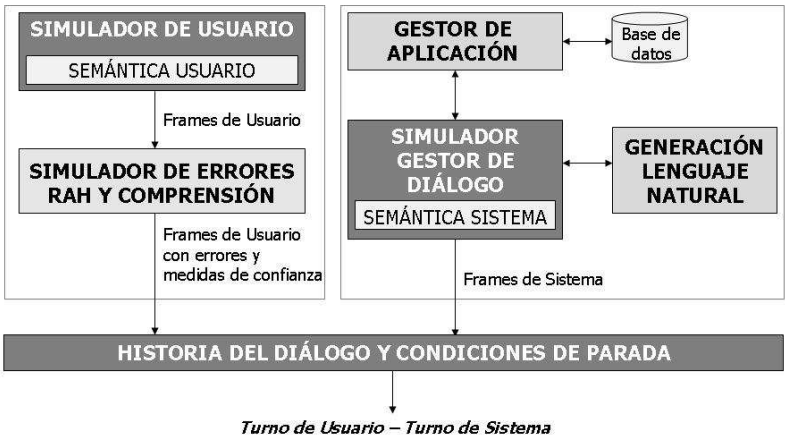


Figura 1. Esquema de módulos que componen la técnica de simulación de diálogos propuesta

generados por el simulador de usuario. De forma experimental, hemos detectado 2,7 errores por diálogo de un análisis de un corpus adquirido para la tarea DIHANA mediante la utilización de un reconocedor del habla y un módulo de comprensión reales. Este valor se puede modificar para adaptar el módulo simulador de errores con respecto al funcionamiento de cualquier módulo de RAH y de comprensión. La Figura 1 muestra la arquitectura de la metodología de adquisición automática de diálogos desarrollada.

Se selecciona una solicitud por parte del usuario para finalizar el diálogo una vez el sistema ha proporcionado la información definida en el objetivo del mismo. Los diálogos que cumplen esta condición antes de un número máximo de turnos se consideran exitosos. El gestor considera que el diálogo no ha alcanzado el objetivo prefijado cuando alguna de las siguientes condiciones tiene lugar:

- El diálogo excede del máximo de turnos.
- La respuesta seleccionada por el gestor de diálogo se corresponde con una consulta no requerida por el simulador de usuarios.
- El módulo gestor de la aplicación genera un mensaje de error debido a que el simulador del usuario no ha proporcionado la información obligatoria necesaria para llevar a cabo una consulta a la base de datos.
- El generador de respuestas proporciona un mensaje de error cuando la respuesta seleccionada por el gestor implica el uso de datos no proporcionados por el simulador del usuario.

La Tabla 1 resume las estadísticas de la adquisición de un corpus de diálogos para la tarea EDECÁN-UPV. Se definió un conjunto de 15 escenarios que especifican los objetivos de los diálogos, simulándose un total de 100.000 diálogos.

Diálogos simulados	100.000
Diálogos con éxito	2.521
Diálogos diferentes	1.973
Número de turnos de usuario por diálogo	4,2

Tabla 1. Estadísticas de la adquisición del corpus mediante la técnica de simulación de diálogos

4. EVALUACIÓN

El corpus descrito en la sección previa se ha utilizado para aprender un gestor de diálogo estadístico para la tarea EDECÁN-UPV de acuerdo con la metodología presentada en [7]. El corpus proporcionado por el Área de Deportes de nuestra universidad se ha utilizado como conjunto de evaluación para evaluar así el comportamiento del gestor de diálogo con un corpus adquirido con usuarios reales.

Hemos definido tres medidas para evaluar el funcionamiento del gestor. Estas medidas se calculan comparando turno a turno la respuesta automáticamente generada por el gestor de diálogo con respecto a la respuesta de referencia anotada para dicho turno en el corpus proporcionado por el Área de Deportes. La primera medida es el porcentaje de respuestas que coinciden con la respuesta de referencia en el corpus ($\%exacta$). La segunda medida es el porcentaje de respuestas que son coherentes con el estado actual del diálogo ($\%correcta$). Finalmente, la tercera medida es el porcentaje de respuestas que se consideran erróneas con respecto al estado actual del diálogo y ocasionarían el fallo del mismo ($\%error$). Estas dos ultimas medidas se han obtenido tras la revisión manual de la respuestas proporcionadas por el gestor de diálogo.

La Tabla 2 muestra los resultados de la evaluación del gestor de diálogo. Los resultados obtenidos tras la experimentación muestran que el gestor de diálogo aprendido se adapta correctamente a los requisitos de la tarea

EDECÁN-UPV, proporcionando un 89,8 % de respuestas que son coherentes con el estado actual del diálogo. Además, el 75,3 % de las respuestas coinciden con la de referencia en el corpus. No obstante, el porcentaje de respuestas proporcionadas por el gestor que pueden ocasionar el fallo del diálogo es considerable (3,9 %). Además, hay un 6,3 % adicional de respuestas que no suponen el fallo del diálogo, pero no son coherentes con el estado actual del mismo (por ejemplo, respuestas que requieren al usuario información que ya ha proporcionado previamente). Queremos reducir ambos porcentajes mediante la incorporación de nuevos diálogos al corpus inicial.

	$\%exacta$	$\%correcta$	$\%error$
Respuesta del gestor	75,3 %	89,8 %	3,9 %

Tabla 2. Resultados de la evaluación del gestor de diálogo estadístico desarrollado para el proyecto EDECÁN

5. CONCLUSIONES

En este artículo, hemos descrito una aproximación para adquirir automáticamente un corpus de diálogos mediante la interacción de un simulador de usuarios y un simulador de gestor de diálogo. Para el desarrollo de ambos módulos definimos la semántica de las posibles respuestas de sistema y un conjunto de condiciones de parada que nos permiten evaluar automáticamente si el diálogo es exitoso o no. De este modo, el esfuerzo necesario para adquirir y etiquetar un corpus de diálogos se reduce notablemente.

El corpus que se ha obtenido mediante la aplicación de esta técnica a la tarea EDECÁN-UPV se ha utilizado para aprender un gestor de diálogo para dicha tarea, utilizando un modelo de diálogo estadístico. Hemos utilizado un corpus adquirido con usuarios reales para evaluar este gestor de diálogo. Los resultados de la evaluación muestran que es posible utilizar el modelo de diálogo aprendido para el desarrollo de un gestor de diálogo, generado sin mucho esfuerzo y con un alto rendimiento.

Actualmente, nuestro objetivo es llevar a cabo la evaluación de los diferentes módulos que componen el sistema de diálogo EDECÁN con usuarios reales. Esta evaluación se está llevando a cabo de forma supervisada, utilizando el gestor de diálogo presentado en este trabajo. Los diálogos que se adquieran se utilizarán además para mejorar el modelo de diálogo inicial.

6. BIBLIOGRAFÍA

[1] S. Young, "The Statistical Approach to the Design of Spoken Dialogue Systems," Tech. Rep., CUED/F-INFENG/TR.433, Cambridge University Engineering Department, Cambridge (Reino Unido), 2002.

[2] E. Levin, R. Pieraccini, y W. Eckert, "A stochastic model of human-machine interaction for learning dialog strategies," in *IEEE Transactions on Speech and Audio Processing*, 2000, vol. 8(1), pp. 11–23.

[3] J. Williams y S. Young, "Partially Observable Markov Decision Processes for Spoken Dialog Systems," in *Computer Speech and Language*, 2007, vol. 21(2), pp. 393–422.

[4] H. Cuayáhuitl, S. Renals, O. Lemon, y H. Shimodaira, "Learning Multi-Goal Dialogue Strategies Using Reinforcement Learning with Reduced State-Action Spaces," in *Proc. of the 9th International Conference on Spoken Language Processing (Interspeech/ICSLP)*, Pittsburgh (Estados Unidos), 2006, pp. 469–472.

[5] F. Torres, E. Sanchis, y E. Segarra, "Development of a stochastic dialog manager driven by semantics," in *Proc. of European Conference on Speech Communications and Technology (Eurospeech'03)*, Ginebra (Suiza), 2003, pp. 605–608.

[6] David Griol, Lluís F. Hurtado, Encarna Segarra, y Emilio Sanchis, "A statistical approach to spoken dialog systems design and evaluation," in *Speech Communication*, 2008, vol. 50, pp. 666–682.

[7] D. Griol, L.F. Hurtado, E. Sanchis, y E. Segarra, "Adaptación de un Gestor de Diálogo Estadístico a una Nueva Tarea," in *Revista de la Sociedad Española de Procesamiento del Lenguaje Natural*, 2007, vol. 39, pp. 231–238.

[8] K. Scheffler y S. Young, "Corpus-based Dialogue Simulation for Automatic Strategy Learning and Evaluation," in *Proc. of NAACL-2001. Workshop on Adaptation in Dialogue Systems*, Pittsburgh (Estados Unidos), 2001.

[9] O. Pietquin y T. Dutoit, "A probabilistic framework for dialog simulation and optimal strategy learning," in *IEEE Transactions on Speech and Audio Processing, Special Issue on Data Mining of Speech, Audio and Dialog*, 2005, vol. 14, pp. 589–599.

[10] K. Georgila, J. Henderson, y O. Lemon, "Learning user simulations for information state update dialogue systems," in *Proc. of the 9th European Conference on Speech Communication and Technology (Eurospeech'05)*, Lisboa (Portugal), 2005, pp. 893–896.

[11] H. Cuayáhuitl, S. Renals, O. Lemon, y H. Shimodaira, "Reinforcement learning of dialogue strategies with hierarchical abstract machines," in *Proc. of IEEE/ACL Workshop on Spoken Language Technology (SLT)*, Palm Beach (Aruba), 2006, pp. 182–186.

ARQUITECTURA DISTRUBUIDA PARA EL DESARROLLO DE SISTEMAS DE DIÁLOGO HABLADO, EDECÁN.

José Enrique García, Alfonso Ortega, Antonio Miguel y Eduardo Lleida.

Grupo de Tecnologías de las Comunicaciones (GTC)
I3A, Universidad de Zaragoza
{jegarlai,ortega,amiguel,lleida}@unizar.es

RESUMEN

El proyecto EDECÁN (TIN2005-08660-C04, *www.EDECÁN.es*) tiene como objetivo aumentar la robustez de un sistema de diálogo de habla espontánea a través del desarrollo de tecnologías para la adaptación y personalización del mismo a los distintos contextos acústicos y de aplicación en los que pueda encontrarse. Con ese objetivo, se propone el uso de una arquitectura distribuida y flexible que permita la cooperación entre diferentes sistemas (módulos de comprensión, reconocedores, gestores de diálogo, etc. desarrollados para diferentes entornos de uso). EDECÁN hace uso de una interfaz de comunicaciones entre módulos capaz de soportar cualquier tipo de servicio en los sistemas operativos Windows y Linux. En este trabajo se describe la arquitectura, los elementos constitutivos de la misma, los procedimientos de gestión, los protocolos y los servicios desarrollados bajo este paradigma.

1. INTRODUCCIÓN

El desarrollo de aplicaciones basadas en las tecnologías del habla ha crecido enormemente en los últimos años. Junto a este crecimiento ha aumentado también la complejidad del software de las mismas, llevándose a cabo un conjunto de iniciativas para el desarrollo de plataformas y arquitecturas que facilitasen su construcción. Como ejemplos, podemos citar la arquitectura GALAXY [1] del MIT bajo la cual se han desarrollado los sistemas JUPITER (información meteorológica) [2], VOYAGER, (información urbana), u ORION (asistente personal) y que ha servido de plataforma para el proyecto DARPA COMMUNICATOR (información de viajes en avión, hotel y alquiler de coches) [3]. Basadas en los principios de GALAXY podemos encontrar otros sistemas como el CU Sonic Spine System de la Universidad de Colorado [4] o el propio sistema presentado, EDECÁN.

La filosofía de la interfaz de comunicaciones EDECÁN consiste en proveer al desarrollador de sistemas distribuidos de diálogo hablado de un conjunto de herramientas sencillas y multiplataforma, que permitan la interacción entre los distintos módulos de la aplicación. Así, es posible el desarrollo de un nuevo módulo integrante de un sistema de diálogo hablado para enviar/recibir datos a través de escrituras/lecturas en una determinadas posiciones de memoria, sin tener que preocuparse del uso de librerías de comunicaciones propias del sistema operativo y del lenguaje de programación.

Al igual que GALAXY, la arquitectura EDECÁN presenta un nodo central configurable que permite un control flexible de las interacciones entre módulos de la aplicación de diálogo a través del establecimiento de un protocolo de comunicaciones y gestión. Además, ofrece un conjunto de librerías para el rápido desarrollo de los módulos que componen un determinado sistema.

El presente artículo está organizado del siguiente modo: En la Sección 2 se realiza una breve descripción de la arquitectura EDECÁN. La Sección 3 se dedica a la presentación del protocolo de comunicaciones, la gestión y las librerías EDECÁN. En la Sección 4 se ofrece una descripción de servicios disponibles y en la Sección 5 se apuntan las líneas futuras de evolución. Por último, en la Sección 6 se presentan las conclusiones.

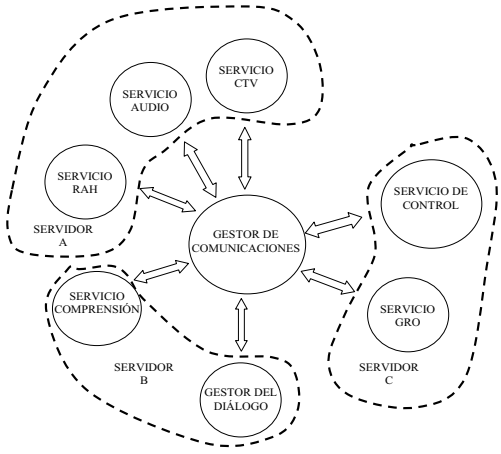


Figura 1. *Esquema de un sistema con arquitectura EDECÁN con 7 servicios repartidos en 3 servidores.*

2. LA ARQUITECTURA EDECÁN

La arquitectura EDECÁN como se muestra en la Figura 1, está formada por un nodo central que permite la comunicación entre servidores conectados a él a través del envío de mensajes. Este nodo central actúa

Este trabajo ha sido parcialmente financiado a través del proyecto TIN2005-08660-C04.

como una batería de clientes y hace funciones de *gestor de comunicaciones*, iniciando los distintos módulos que componen el sistema y llevando a cabo el enrutado de paquetes entre ellos con dos modos de funcionamiento, a través de encaminamiento forzado, si la cabecera del paquete cuenta con un destino prefijado o mediante encaminamiento por defecto a través de la creación de una tabla de rutas configurable. El gestor de comunicaciones puede ser controlado mediante otro módulo del sistema, el cual se conecta a él como cliente y le envía comandos para realizar modificaciones sobre el sistema, como conectar servicios, desconectarlos, dar de alta nuevos servicios, o modificar la tabla de rutas.

Cada uno de los componentes del sistema (*service*) es un servicio genérico sobre el que puede instanciarse cualquier tipo de módulo (RAH, comprensión, gestor de diálogo,…). Son capaces de enviar y recibir datos y están implementados de tal manera que la máquina en la que residen (*servidor*) dispone de un proceso de fondo o demonio (*super-server*) que se ejecuta de forma continua en segundo plano y acepta las conexiones entrantes del gestor, lanzando cada uno de los servicios en un nuevo proceso y dejando la responsabilidad de la gestión de la conexión al propio servicio.

Figura 2. *Ejemplo de fichero de servicios disponibles y ejemplos de configuración de algunos servicios.*

Los servicios disponibles en una determinada máquina se definen a partir de un fichero de configuración en formato XML. Esta lista de servicios potenciales será consultada por el servicio genérico *service* para llevar a cabo la instanciación del servicio concreto. De este modo, cada módulo del sistema de diálogo distribuido se define a partir de una librería dinámica y un fichero de configuración en formato XML. Tanto el fichero de servicios disponibles como ejemplos de ficheros de configuración de servicios se pueden ver en la Figura 2. La librería dinámica contendrá las funciones necesarias para que el módulo en cuestión sea capaz de realizar su cometido y el fichero de configuración contiene el conjunto de comandos a los que el servicio es capaz de responder y el nombre de la función asociada a dicha respuesta.

3. PROTOCOLO DE COMUNICACIONES, GESTIÓN Y LIBERÍAS EDECÁN.

EDECÁN está construida a partir de conexiones TCP/IP aunque para determinados servicios también está previsto el uso del protocolo UDP. El modo principal de funcionamiento es a través del envío de paquetes de texto en formato XML aunque también dispone de un modo de transmisión binario para el envío de información que demande un ancho de banda elevado como puede ser la voz.

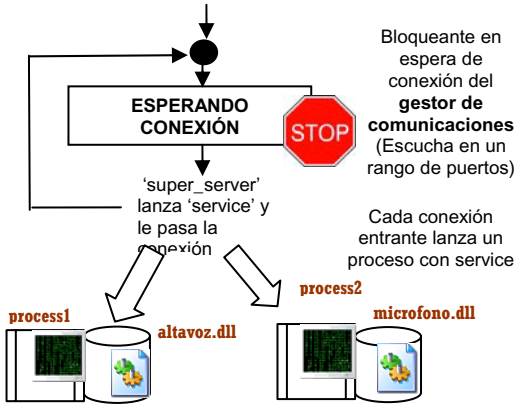


Figura 3. *Diagrama de flujo del funcionamiento de un servidor EDECÁN.*

3.1. Protocolo de Comunicaciones.

Fundamentalmente, se establece un protocolo de comando-respuesta entre los diferentes módulos de modo que el servicio receptor ejecutará una determinada acción como respuesta a la recepción de un determinado comando. Con el objetivo de simplificar al máximo la creación de módulos, la arquitectura cuenta con el proceso *service* que gestionara las comunicaciones de cada uno de los servicios haciendo transparente el envío y recepción de paquetes desde o hacia los servicios. Cada módulo se establece como un servicio genérico que, tras la recepción del primer paquete de inicialización, se instancia como un servicio concreto a través de la carga de la librería dinámica que contiene las funciones de dicho servicio concreto. Este proceso aparece ilustrado en la Figura 3.

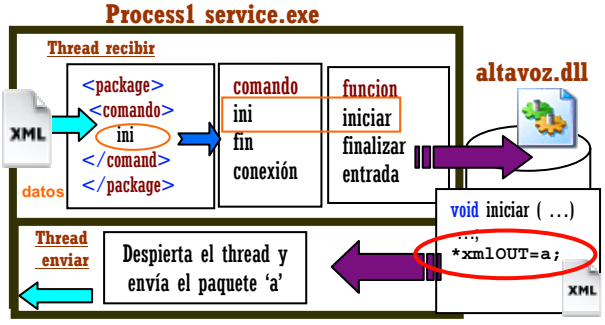


Figura 4. *Proceso de recepción de comandos y ejecución de funciones en un servicio.*

Una vez instanciado el servicio su modo de funcionamiento sigue el patrón mostrado en la Figura 4. En modo recepción, se comunicará con las funciones de la librería dinámica a través del paso de información mediante el uso de memoria compartida. Al recibir un paquete de otro módulo, este es interpretado y tras extraer el comando que contiene, se ejecuta su función asociada. Después se realiza el paso del resto de la información contenida en el paquete a través de estructuras de memoria compartida. En cuanto a la transmisión de paquetes, el proceso encargado de la gestión de las conexiones realiza una continua revisión de los paquetes que resultan de la ejecución de alguna de las funciones de la librería procediendo al envío de aquellos que se encuentran pendientes.

3.2. Biblioteca de Funciones.

El Software Development Kit (SDK) de EDECÁN cuenta también con una biblioteca de funciones que hace más simple la construcción, el envío y la recepción e interpretación de paquetes XML. En ella se encuentran definidos un conjunto de paquetes genéricos de uso muy común como pueden ser los de conexión, desconexión, etc. junto con un conjunto de funciones para crear paquetes más específicos

3.3. Gestión del Sistema Distribuido.

Cualquier sistema distribuido construido a partir de la arquitectura EDECÁN puede ser gestionado a través de una intuitiva y sencilla interfaz gráfica de usuario (GUI) mostrada en la Figura 5, a través de la cual pueden enviarse órdenes de configuración al gestor de comunicaciones, para que éste las ejecute. Las principales órdenes que puede dar son aquellas relacionadas con el montaje/desmontaje de un nuevo sistema, alta/baja o conexión/desconexión de un servicio, modificación la tabla de rutas del sistema o de un servicio, peticiones de información de los servicios del sistema o de las rutas del sistema. La GUI desarrollada también permite visualizar el estado de los diferentes módulos y la tasa de transmisión de subida/bajada de cada uno de ellos.

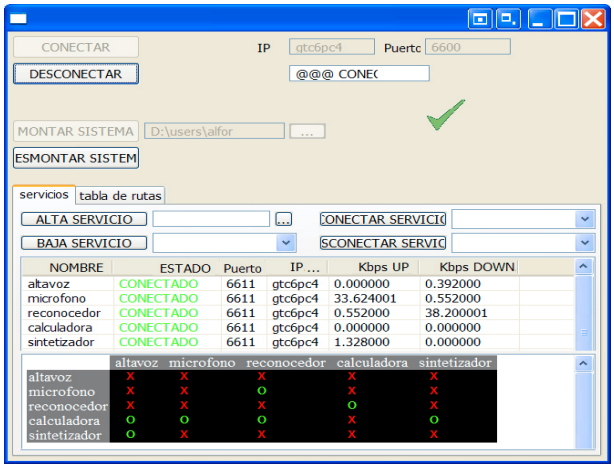


Figura 5. *Interfaz Gráfica de Usuario para control y monitorización de una aplicación.*

Figura 6. *Ejemplo de fichero de configuración de un sistema con la arquitectura EDECÁN.*

La construcción de un nuevo sistema, una vez implementados los diferentes módulos que lo componen, es tan simple como el envío de un fichero en formato XML que contenga todos los datos necesarios (IP, puerto, tipo de servicio, etc.) de cada uno de los módulos constituyentes, como se ilustra en la Figura 6. Junto a esta información, se envía la tabla de rutas a través de la cual se llevará a cabo el encaminamiento por defecto en el gestor de comunicaciones. Dicho fichero será enviado al gestor de comunicaciones encargado del montaje del sistema. A partir de ese momento, se encargará de enviar los diferentes paquetes de conexión e inicialización de los diferentes servicios a las máquinas en las que residirán estos.

4. LISTADO DE SERVICIOS.

Dado que el objetivo último del desarrollo del SDK EDECÁN es la investigación, desarrollo y creación de sistemas de diálogo hablado, los servicios implementados hasta el momento se han centrado principalmente en este ámbito. Sin embargo, las potencialidades del *Midleware* desarrollado trascienden este ámbito y podrán ser desarrollados módulos en cualquier otro campo de manera fácil y sencilla.

4.1. Servicios Básicos para Sistemas de Diálogo Hablado.

En este sentido, se han desarrollado módulos constitutivos de un sistema de diálogo hablado como un elemento capaz de capturar audio a partir del dispositivo de sonido de la máquina en la que se encuentra y transmitirlo a cualquier otra ubicación haciendo uso de un amplio abanico de modos de codificación. Por otro lado se ha implementado el servicio dual encargado de la recepción y posterior reproducción del audio. Con estos dos servicios pueden desarrollarse sencillas aplicaciones de Voz sobre IP (VoIP). Asimismo, se cuenta con sistemas de reconocimiento automático del habla, módulos de comprensión, de gestión del diálogo y de generación de respuestas orales que operan en este paradigma definido. También se ha creado un servicio

que, a partir de un motor de conversión texto-voz, envía las muestras de voz codificadas a otros módulos del sistema. Además, se cuenta con servicios que hacen posible la comunicación entre lugares remotos vía texto o módulos capaces de realizar las tareas de gestión y monitorización de un sistema ya montado.

4.2. EDECÁN *Mobile*.

La estructura EDECÁN cuenta con un servicio especial que permite la extensión de sus usos a dispositivos portátiles (PDA, teléfonos móviles,...). Este tipo de dispositivos reciben una consideración especial debido a su limitación computacional y a la posibilidad de contar con un ancho de banda reducido. Con el objetivo de salvar estos inconvenientes, se estableció el uso de paquetes binarios mediante conmutación de paquetes en circuito virtual. Por otro lado, se desarrollaron servicios específicos como puede ser un Front-End de extracción y compresión de parámetros acústicos para arquitecturas de coma fija a partir del estándar ETSI ES 201 108.

Otro inconveniente que habitualmente presentan estos dispositivos es su conexión a través de políticas DHCP. Esto hace que la dirección IP asignada no siempre sea la misma por lo que no procede que éstos actúen como servidor. Para solucionar esto, se desarrolló un servicio puente que permite la conexión con servicios instalados en dispositivos móviles sin modificar la estructura ni la filosofía EDECÁN. De este modo, el servicio puente residente en una máquina con dirección IP estable actúa como servidor de cara al gestor de comunicaciones pero a su vez posibilita que el dispositivo móvil se conecte a él como cliente. Así, no es necesario el conocimiento previo de la dirección IP asignada al dispositivo móvil. Una vez establecidas las conexiones, el servicio puente sólo deberá reenviar los paquetes recibidos hacia o desde el cliente móvil, siendo así transparente su funcionamiento.

4.3. Otros Servicios Disponibles.

Entre los servicios desarrollados para esta arquitectura, cabe destacar un módulo que detecta la presencia de una persona gracias al uso de una *web-cam* mediante un software de detección de caras, algo que puede resultar de gran utilidad para saber si alguien tiene la intención de dirigirse al sistema de diálogo hablado. También se ha desarrollado un completo *web-browser* capaz de mostrar aquellas páginas web que un usuario, a través del envío de comandos orales mediante un sistema de diálogo basado en la arquitectura EDECÁN, vaya solicitando. Asimismo, se cuenta con un módulo que realiza una adaptación de información proveniente en formato XML a una plantilla predefinida con XSLT para generar un documento HTML que podrá ser visualizado por cualquier *web-browser* con el formato deseado. Por último se han implementado módulos de adaptación de los modelos acústicos al locutor capaces de personalizar cualquiera de las aplicaciones basadas en voz que sean implementadas bajo la arquitectura EDECÁN y módulos de normalización de vectores de características para la adaptación al entorno acústico de acuerdo con el algoritmo MEMLIN [5].

5. EVOLUCIONES FUTURAS.

Como posibles evoluciones futuras se plantea la modificación de los gestores de comunicaciones para que éstos sean capaces de dar soporte a su interconexión. También se llevará a cabo la implementación de un gestor de comunicaciones más robusto, que pueda gestionar las desconexiones eventuales, de modo que sea capaz de reconectar a los servicios cuando estos se caigan.

Por último, se prevé el desarrollo de una nueva figura, un *Directorio Público de Servicios* (DPS), que aglutine los nombres, funcionalidades, direcciones IP, puertos y toda la información necesaria para implementar sistemas bajo la arquitectura EDECÁN. Así, cuando un gestor de comunicaciones solicita un determinado servicio, el DPS le proporciona la lista de todos los disponibles, junto con sus características, su carga de trabajo actual, etc., indicando igualmente el servidor correspondiente donde se hallan.

6. CONCLUSIONES

En el presente trabajo se ha presentado la arquitectura desarrollada dentro del proyecto EDECÁN. Se trata de una arquitectura que permite la cooperación entre diferentes sistemas (módulos de comprensión, reconocedores, gestores de diálogo, etc. desarrollados para diferentes entornos de uso) de un modo fácil y flexible para el desarrollo de sistemas distribuidos de diálogo hablado aunque es posible su extensión a cualquier otro tipo de tarea. Se ha presentado su interfaz de comunicaciones multiplataforma y descrito sus elementos constitutivos, procedimientos de gestión, protocolos y algunos de los servicios desarrollados.

Los autores desean agradecer a todos los investigadores del proyecto coordinado EDECAN su colaboración y aportaciones a este trabajo.

7. BIBLIOGRAFÍA

[1] Seneff, S. et al. “Galaxy-II: a reference arquitecture for conversational systems development”. ICSLP, 931-934. 1998.

[2] V. Zue, et al., "JUPITER: A Telephone-Based Conversational Interface for Weather Information," IEEE Trans. on Speech and Audio Proc., Vol. 8 , No. 1, Jan. 2000

[3] Walker M., Hirschmann L., Aberdeen J. “Evaluation for DARPA COMMUNICATOR Spoken Dialog Systems”. LREC, 2000.

[4] Hacioglu, K. and Pellom, B. “A Distributed Architecture for Robust Automatic Speech Recognition”. ICASSP, 328-331. 2003.

[5] L. Buera, E. Lleida, A. Miguel, A. Ortega, O. Saz, “Cepstral vector nomalization based on stereo data for robust speech recognition”, IEEE trans on Audio, Speech and Language Processing, vol.15, pp.1098-1113. Marzo 2007.

ARQUITECTURA MULTIMODAL CONTROLADA POR VOZ: REVISIÓN DE METÁFORAS DE INTERACCIÓN

David Escudero-Mancebo, Héctor Olmedo-Rodríguez, Valentín Cardeñoso-Payo
ECA-SIMM Laboratory, Universidad de Valladolid[1]
Campus Miguel Delibes s/n. 47008 Valladolid
{descuder, holmedo, valen}@infor.uva.es

RESUMEN

En esta comunicación presentamos una plataforma para el desarrollo de aplicaciones multimodales dirigidas por voz. Se presenta un lenguaje de especificación de escenas multimodales y la arquitectura soporte. El lenguaje y la arquitectura se validan revisando la cobertura de una serie de metáforas básicas que tienen que ver con la interacción gráfica, interacción vocal y la combinación de ambos modos. Esta evaluación pone en evidencia las capacidades de la plataforma y apunta el trabajo futuro que debe realizarse.

1. INTRODUCCIÓN

Una de las aplicaciones más interesantes de las tecnologías del habla (TH) es su uso para implementar interfaces multimodales [1]. La voz puede ser un complemento a la interacción gráfica en el uso de determinados terminales tipo kiosco o en terminales móviles. Otro ámbito donde las TH pueden ser un complemento importante es el de las aplicaciones de entretenimiento o de entrenamiento (AEE). Uno de los ámbitos más utilizados para el desarrollo de aplicaciones de (AEE) son los entornos 3D.

A pesar del espectacular crecimiento tanto de las aplicaciones 3D como de la investigación en TH el estado del arte en aplicaciones multimodales que combinen 3D con sistemas de diálogo se caracteriza por la presencia de un número pequeño de prototipos. Aunque hay propuestas de estandarización, los prototipos existentes parecen soluciones ad-hoc en un ámbito reducido y con aplicación limitada (con excepción quizá de las aplicaciones de visual speech). En este artículo presentamos una plataforma para el desarrollo de aplicaciones 3D que incluyan interacción vocal cuyo objetivo es ofrecer un marco genérico para la programación conjunta de escenas y personajes 3D y de diálogos teniendo en cuenta los estándares disponibles en ambos ámbitos.

La plataforma es descrita en más detalle en [2]. En esta comunicación se plantea además una propuesta para la evaluación de la plataforma propuesta. Esta evaluación se realiza en base a un análisis de cobertura de una serie de metáforas de interacción básicas encontradas en la bibliografía básica. Este análisis permite destacar los méritos de la plataforma y apuntar los aspectos donde debe ser mejorada.

Primero presentaremos nuestra propuesta de interacción multimodal (gráfica y vocal) con espacios 3D. Seguidamente enumeraremos las metáforas de interacción gráfica y vocal así como los tipos de cooperación entre éstas a la vez que revisamos las posibilidades de la plataforma para dar cobertura a dichas metáforas. Las conclusiones destacan el trabajo futuro a realizar para mejorar la plataforma.

2. LA PLATAFORMA MULTIMODAL

En este apartado describiremos el lenguaje de especificación y la arquitectura propuesta para definir aplicaciones que permiten interactuar de manera multimodal con entornos 3D.

2.1. El lenguaje XMMVR

El eXtensible markup language for MultiModal interaction with Virtual Reality worlds o XMMVR es un lenguaje de marcas para especificar escenas, comportamientos e interacción. Cada mundo es modelado por un elemento XMMVR, usando la metáfora de película cinematográfica. Es un lenguaje de marcas híbrido porque en éste quedan embebidos otros lenguajes como VoiceXML o X+V para interacción vocal y X3D o VRML para describir las escenas 3D. Los ficheros XML válidos para el DTD de XMMVR incluyen enlaces a los programas y ficheros necesarios para ejecutar el mundo 3D definido. Nuestro sistema es dirigido por eventos, por ello se requiere definir una mínima lista de eventos para sustituir la línea de tiempo.

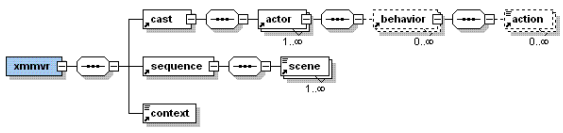


Fig. 1. *Elementos del lenguaje XMMVR*

[1] Trabajo parcialmente financiado por el proyecto de la Junta de Castilla y León (VA077A08)

La Figura 1 muestra la estructura de un documento XMMVR. Cualquier elemento *xmmvr* está formado por un reparto de actores llamado *cast* y una secuencia de escenas llamada *sequence* determinando la evolución temporal del mundo. El elemento *context* se reserva para uso futuro.

Al usuario se le considera un miembro de la audiencia. Es capaz de interactuar con los actores del mundo aunque no es considerado como un actor del mundo.

Cada *actor* del reparto es un elemento con apariencia gráfica descrita en un fichero VRML y un comportamiento *behaviour* que especifica la interacción del usuario. Cada comportamiento está definido como un par <*evento, lista de acciones*>. Las acciones son ejecutadas cuando una condición *condition* se cumple.

El usuario genera eventos utilizando la interacción gráfica *GUI* o la interacción vocal *VUI*. Existen también eventos del sistema para definir la interacción con otros actores del mundo (*eventos ACT*) o para interactuar con el sistema (eventos *SYS*). La *lista de acciones* es un conjunto de acciones a ejecutarse cuando ocurre un evento. Las acciones pueden ser de tipo GUI, VUI, ACT o SYS. Las acciones GUI modifican la apariencia gráfica del mundo 3D. Las acciones VUI son diálogos. Las acciones ACT son mensajes enviados entre actores. Acciones SYS son navegaciones entre escenas.

El elemento *sequence* planifica las escenas *scenes* del mundo. Por defecto las escenas se muestran en el mismo orden en el que están escritas en el documento. Los eventos y acciones SYS permiten navegar entre escenas cambiando el orden secuencial por defecto. Debe definirse por lo menos una escena en el mundo xmmvr. La interacción es sólo posible si se ha definido al menos un actor.

De acuerdo a estas premisas, se ha definido un DTD [3] de manera que es posible desarrollar aplicaciones multimodales escribiendo un fichero XML válido para el DTD de XMMVR. Las escenas 3D, los actores, sus comportamientos y la interacción con el usuario se definen en el mismo fichero de marcas. Este fichero se usa por la arquitectura del sistema para ejecutar la aplicación como se describirá a continuación.

2.2. La plataforma XMMVR

Hemos creado un marco para desarrollar aplicaciones de interacción persona-ordenador multimodales donde el flujo de la interacción en entornos 3D está conducido por diálogos hablados. Para construir una aplicación utilizando nuestro marco el desarrollador tiene que especificar el mundo virtual, la secuencia de diálogos y la lista de acciones a lanzarse cuando los eventos son generados por los usuarios. En la sección anterior hemos descrito un lenguaje para especificar estos elementos en un documento XMMVR común. En esta sección describimos la arquitectura del sistema responsable de analizar sintácticamente documentos XMMVR y ejecutar la correspondiente aplicación multimodal.

Una de nuestras metas fue construir una aplicación web multimodal, por lo que desarrollamos un sistema embebido en un navegador web. Los diálogos están programados utilizando VoiceXML y las escenas 3D y actores están descritos en VRML. Hemos desarrollado un gestor de mundo para planificar las acciones a ejecutarse en el mundo 3D utilizando un applet Java.

2.2.1. Gestión del diálogo vocal

La Figura 2 muestra los componentes que gestionan la interacción vocal. Existe un componente embebido en el navegador web que inicia la ejecución solicitando un diálogo al dispatcher. Recupera el documento VXML correspondiente de un repositorio y lo envía al intérprete. El intérprete VXML ejecuta el diálogo utilizando la plataforma vocal. El interfaz entre el intérprete VXML y la plataforma vocal son prompts (a ser sintetizados) y fields (la salida del reconocedor de diálogo automático) de acuerdo al estándar VoiceXML.

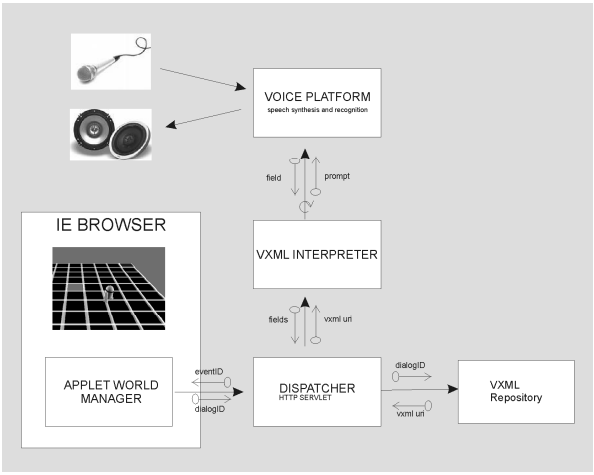


Fig. 2. *Componentes de la arquitectura para la gestión del diálogo hablado*

Hemos desarrollado un sistema que utiliza un applet Java en un navegador web Internet Explorer. Utiliza el navegador VRML CORTONA [4] para mostrar el estado del mundo. La interacción GUI se basa en el API EAI [5]. Utilizamos un servlet sobre un servidor Apache Tomcat para alimentar el navegador vocal de nuestro sistema de diálogo. Creamos nuestro propio intérprete VXML que utiliza la plataforma vocal ATLAS de Ibervox [6]. Como los componentes vocales están distribuidos sobre diferentes servidores, las aplicaciones multimodales se pueden ejecutar en un PC convencional con audio y los navegadores apropiados.

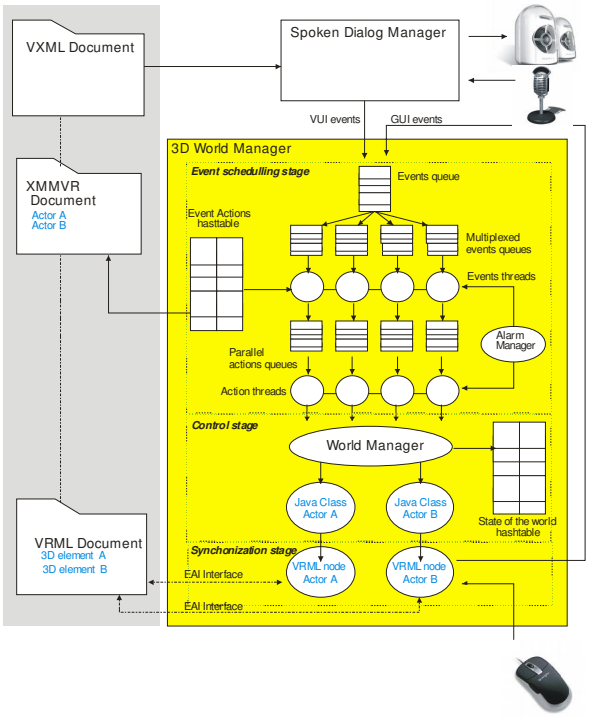


Fig. 3. *Componentes de la arquitectura del gestor del mundo 3D*

2.2.2. Gestión del mundo 3D

Los componentes necesarios para hacer al mundo 3D reaccionar a eventos se muestran en la Figura 3. Los eventos VUI se despachan por el gestor de diálogo hablado y los eventos GUI se generan típicamente clickando los elementos 3D. El sistema obtiene la información acerca de las escenas 3D y los actores del documento XMMVR. El mismo documento especifica el comportamiento de los actores como pares <*evento, lista de acciones*> y enlaza con el documento VRML que especifica la escena 3D y los actores. La parte izquierda de la Figura 3 muestra la relación entre dos actores XMMVR llamados actor A y actor B y dos elementos 3D VRML llamados nodo A y nodo B. En respuesta a eventos, el sistema hace cambios en los nodos de acuerdo a especificaciones del documento XMMVR.

La primera etapa del sistema es la tarea de Gestión de eventos, donde los eventos son encolados y multiplexados en un número de colas paralelas (en este caso cuatro colas, pero este parámetro es programable). Cada cola es atendida por el hilo correspondiente, y esto permite la ejecución concurrente de acciones en la escena. Los hilos acceden a la tabla hash que contiene las acciones correspondientes con los eventos, como se describe en el documento XMMVR. Si alguna de las acciones no es elemental, el hilo se encarga de descomponerla, produciendo una nueva cola de acciones elementales preparada para ser ejecutada por el Gestor del mundo en el próxima etapa. Hemos programado un Gestor de alarmas para añadir anti-acciones que eliminan varias acciones a ser canceladas cuando ocurren eventos excepcionales (alarmas).

Cada elemento 3D VRML tiene dos clases Java asociadas: Control e Interfaz. Control ejecuta acciones elementales e interacciona con las variables correspondientes de la tabla hash "Estado del mundo" e Interfaz manipula los gráficos del nodo 3D. Entre la clase Java Interfaz y los nodos 3D hay un interfaz (escenario *synchronization* en la Figura 3) que es responsable de utilizar el EAI para manipular los elementos VRML y recibir los eventos GUI y direccionarlos a la entrada del sistema. En la Figura 3, el nodo A es manipulado a través de la clase Java Interfaz Actor A. La diferencia entre el nodo A y el nodo B es que el nodo B puede potencialmente enviar eventos GUI tales como clicks de ratón en el nodo 3D VRML B.

El Gestor del mundo lee las acciones elementales y llama a los métodos Java correspondientes. Además, actualiza la tabla hash "Estado del mundo" que almacena el estado de las propiedades del mundo. La información de esta tabla hash es utilizada para asignar valores a los parámetros de los métodos Java. El gestor es también responsable de actualizar esta tabla de acuerdo a las acciones ejecutadas.

3. REVISIÓN DE METÁFORAS

En este apartado presentaremos las características y clasificación de las metáforas de interacción gráfica y vocal así como los tipos de cooperación entre ellas para evaluar las capacidades de nuestra propuesta de cara a su resolución.

3.1. Interacción gráfica

Empleamos las referencias [7] y [8] que clasifican los diferentes modos de interacción gráfica de acuerdo a metáforas fundamentales (teatro y locomoción) y metáforas navegacionales (elevador, vehículo, deslizamiento, silla voladora y tele transporte). Se ha comprobado que el lenguaje XMMVR tiene expresividad suficiente para atender estas metáforas. No entramos en detalle en este artículo porque estamos más interesados aquí en la interacción vocal.

3.2. Interacción vocal

Oviatt [9] distinguió, entre las características básicas para la interacción multimodal, la necesidad de restringir el lenguaje por las limitaciones de los sistemas de diálogo para entender el lenguaje natural y la necesidad de realimentación para adaptar el lenguaje a nuevas situaciones o cambios de estado.

El uso de VXML impone diálogos restringidos y permite la realimentación.

McGlashan [10] distingue cuatro metáforas de interacción vocal:

Proxy o Delegado: El usuario puede tomar control de varios agentes (cambiar de agente, selección en la acción) en el mundo virtual e interactuar con el mundo virtual a través de ellos, por ejemplo: pintor, ¡pinta la casa de rojo!
Divinity: El usuario actúa como un dios y controla el mundo directamente, por ejemplo: que la casa sea roja!
Telekinesis: Los objetos y agentes en el mundo virtual pueden ser interlocutores de diálogo del usuario, por ejemplo: casa, ¡píntate de rojo!
Interface Agent: El usuario se comunica con un agente, separado del mundo virtual, que ejecuta sus comandos.

El sistema está preparado para resolver las metáforas de *Proxy* e *Interface Agent* ya que los diálogos están vinculados vía el elemento *behavior* a los distintos actores. Sin embargo, el uso de la metáfora *Proxy* exigiría la disponibilidad de un intérprete VXML que dispusiera de diferentes voces para asociar a los diferentes agentes que fueran a intervenir.

Las metáforas *Telekinesis* y *Divinity* podrían simularse empleando un actor sin apariencia gráfica que fuera responsable de dialogar con el usuario e interactuar con el resto de agentes vía eventos de tipo ACT.

3.3. Cooperación entre modalidades

Existen cinco tipos básicos de cooperación entre modalidades según [11]:
Transferencia: Parte de la información producida por una modalidad es usada por otra modalidad.
Equivalencia: Ambos modos podrían tratar la misma información.
Especialización: Un determinado tipo de información es siempre procesada por la misma modalidad.
Redundancia: La misma información es procesada por ambas modalidad.
Complementariedad: Diferentes partes de información son procesadas por cada modalidad pero tienen que ser combinadas.

La *transferencia* está garantizada por el mantenimiento de un contexto tanto en el lenguaje XMMVR como en el *Gestor del mundo* de la arquitectura (*hash* de Estado del mundo). Uno de los modos puede provocar la ejecución de una acción que modifique el valor de estas variables de contexto y esta información ser usada por el otro modo.

La *complementariedad* también puede ser programada usando el contexto en el lenguaje XMMVR.

La *equivalencia* también puede darse ya que la misma secuencia de acciones puede ser ejecutada como consecuencia de un evento vocal o gráfico. La *especialización* es responsabilidad del programador de escenas que puede delegar determinadas tareas a uno u otro modo.

Por último, la *redundancia* es el caso más problemático. Cuando los dos modos introducen información que se refiera a lo mismo pueden darse situaciones de bloqueo dado que las acciones pueden ejecutarse de forma concurrente. Esto está previsto con la inclusión del elemento *Alarm manager* que se espera pueda anular acciones de las colas. Sin embargo existen situaciones que van a precisar modificaciones en la arquitectura como pueden ser la cancelación de diálogos cuando el modo gráfico introduzca una información redundante.

5. CONCLUSIONES Y TRABAJO FUTURO

Hemos presentado nuestra propuesta para definir e implementar aplicaciones que permitan interacción multimodal con entornos 3D. Hemos revisado el estado del arte referente a metáforas de interacción gráfica y vocal así como los tipos de cooperación entre distintas modalidades. Tras ello, hemos evaluado nuestra propuesta concluyendo que nos permite implementar de manera notable metáforas de interacción gráfica estructurales y navegacionales. Se hace más difícil la implementación de las distintas metáforas de interacción vocales así como los tipos de cooperación entre modalidades.

De estas deficiencias que detectamos en nuestro sistema podemos definir nuestro trabajo futuro que podemos resumir en: adaptar la solución propuesta a resolver cada metáfora de interacción gráfica y vocal y cada tipo de cooperación entre distintas modalidades.

6. BIBLIOGRAFÍA

[1] Jaimes A., Sebe N. Multimodal human-computer interaction: A survey**.** IEEE IW on HCI & ICCV 2005.
[2] Olmedo H. et al. Conceptual and practical framework for the integration of multimodal interaction in 3D worlds, New Trends on HCI. Springer (en prensa).
[3] DTD de XMMVR. http://www.xmmvr.info/.
[4] Cortona: http://www.parallelgraphics.com/.
[5] Phelps A.M. Introduction to the External Authoring Interface, EAI. Rochester Inst. of Technology, Dep. of Information Technology, 1999.
[6] Atlas Ibervox: http://www.verbio.com/.
[7] Contigra. http://www.contigra.com/.
[8] Dachselt R., Action Spaces - A metaphorical concept to support navigation and interaction in 3D interfaces; User Guidance in Virtual Environments. Workshop "Usability Centred Design and Evaluation of Virtual 3D Environments", 2000.
[9] Oviatt S., Cohen P., Multimodal interfaces that process what comes naturally. Com. of the ACM, 2000.
[10] McGlashan S., Axling T., Talking to Agents in Virtual Worlds. UK VR-SIG Conference, 1996.
[11] Martin, J. C. TYCOON: theoretical and software tools for multimodal interfaces, AAAI Press., 1998.

AUTOMATIC WORD STRESS MARKER FOR PORTUGUESE TTS

Daniela Braga[1] and Luis Coelho[2]

[1]MLDC – Microsoft Language Development Center, [2]Instituto Politécnico do Porto - ESEIG

ABSTRACT

In this paper, a linguistically rule-based word stress marker for European and Brazilian Portuguese is described. The main goals that led us to develop this application were to increase the grapheme-to-phone performance and to automatically provide lexical stress information to train a Hidden Markov Models based Speech Synthesis System for European Portuguese. The system was implemented and tested giving rise to 99.59% of word accuracy rate for European Portuguese and 99.60% of word accuracy rate for Brazilian Portuguese. This system was also tested with Galician texts and 98.52% of word accuracy rate was obtained.

1. INTRODUCTION

Stress marking has a major impact in two modules of a Text-to-Speech (hereafter TTS) system: on the one hand, in grapheme-to-phone(me) conversion (and syllabification module), and on the other hand in the prosody module. Stress is also part of the phonetic information of the lexicon used as input for the text analysis of a dictionary-based Text-to-Speech system. Stress information used in Hidden Markov Models-based Speech Synthesizers (HTS) offline training has also proved to improve synthetic voice intelligibility and naturalness [1]. Although word stress in Portuguese is widely studied in literature [2], there is not much work published on automatic word stress marking for Portuguese. In the early 90's, Oliveira et al. [3] pointed out the importance of stress marking and refer that the DIXI version uses 18 rules. More recently, Teixeira et al. [4] described a stress marker algorithm with only 3 rules, followed by a table of exceptions (although not published), while Barros & Weiss [5] presented a maximum entropy-based stress model which was trained with a 4219 word stress annotated corpus. In [4], no performance rates of this rule-based system are presented. In [5], the accuracy rate of the proposed statistical method is 85.57%. In this paper, we present a tool for automatic word stress marking for European and Brazilian Portuguese. This tool represents the latest development of a preliminary version presented in a previous work [6], which was designed only for Brazilian Portuguese language and whose results were 98.58% of accuracy rate. Our current version has not only largely overcome the initial performance results, but it has also turned to be more flexible, supporting both European and Brazilian Portuguese varieties. This paper is structured as following: in section 2, the rule-based automatic word stress marker for Portuguese is presented; in section 3, the tests are described and the results are discussed; in section 4, the application of this work to Galician is presented and discussed; in section 5, main conclusions are summarized and future work is foreseen.

2. AUTOMATIC WORD STRESS MARKER

The proposed word stress marker is composed by 31 rules and is based on the analysis of context around the last graphemes of each word. After the text is separated into sentences and the sentences are separated into words, the system checks word by word in order to find non stressed words, which are pre-defined and receive no stress mark. According to literature [7], non stressed words are monosyllabic high frequent function words such as monosyllabic definite and indefinite articles (<o, a, os, as, um, uns>); clitics (<me, te, se, o, a, os, as, lo, la, los, las, no, na, nos, nas, lhe, lhes, nos, vos>) and their contractions (<mo, ma, mos, mas, to, ta, tos, tas, lho, lha, lhos, lhas, no-lo, no-la, no-los, no-las, vo-lo, vo-la, vo-los, vo-las>); relative pronoun <que>; monosyllabic prepositions (<a, com, de, em, por, sem, sob>) and their contractions (<do, da, dos, das, ao, à, aos, às, no, na, nos, nas, num, nuns>); and monosyllabic conjunctions (<e, mas, nem, ou, que, se>). The last grapheme of the word, which receives the position number zero ^(0), is the starting point for each rule. Then, the left context of this zero position is analyzed grapheme by grapheme and the stressed vowel is predicted according to the different combinations of the graphic patterns. The symbol set used in the rules design is shown in Table 1. In Table 2, the word stress marker complete algorithm is displayed. Most of the rules are repeated (e.g. rules 5 and 6, rules 7 and 8, etc.), bearing in mind the adjustment of the graphemes' position, so as to predict plurals of nouns and adjectives. The stressed vowel is marked with a digit (<1>) and not with an apostrophe, in order to avoid ambiguities in the TTS text normalization module.

symbol	Meaning
^(0)	Word last grapheme
^(1)	Word penultimate grapheme
^(2)	Word antepenultimate grapheme
^(3)	Word third last grapheme

^(4)	Word fourth last grapheme
T	Position occupied by the stressed vowel
T=1	Stressed vowel is the penultimate grapheme
/	Except
→	Then
{x}	Grapheme x
{ }	Space
1	Stressed vowel

Table 1. *Symbols used in the automatic word stress marker algorithm for EP and BP.*

#	Rule	Exemple
1	List of non stressed words.	por, um, se
2	If there is an orthographic accent[1], the accented vowel is the stressed one. The acute or circumflex accents have precedence over the tilde[2].	órgão, órgãos, bênção, bênçãos
3	If the word has only one vowel → T= vowel	tem, vem, bem, vi
4	If ^(0) = {r, l, z, x} → T = 1	propor, juiz
5	If ^(0) = {m} and ^(1) = {i,o,u} → T = 1	pudim, bombom, comum
6	If ^(0) = {s} and ^(1) = {n} and ^(2) = {i,o,u} → T = 2	pudins, comuns
7	If ^(0) = {i} and ^(1) = {u,} and ^(2) = {q,g} → T = 0	caqui, aqui, sagüi
8	If ^(0) = {s} and ^(1) = {i} and ^(2) = {u,} and ^(3) = {q,g} → T = 1	caquis, sagüis
9	If ^(0) = {i,u} and ^(1) é vogal → T = 1	caju, grau, pneu
10	If ^(0) = {i,u} and ^(1) is not a vowel → T = 0	caju, javali
11	If ^(0) = {s} and ^(1) = {i,u} and ^(2) is not a vowel → T = 1	cajus, javalis
12	If ^(0) = {s} and ^(1) = {i,u} and ^(2) is a vowel → T = 2	andais, pauis, graus.
13	If ^(0) = {and} and ^(1) = {u} and ^(2) = {q, g} and ^(3) is vowel/{u} → T = 3	Alambique, Henrique, obrigue
14	If ^(0) = {s} and ^(1) = {e} and ^(2) = {u} and ^(3) = {q, g } and ^(4) is vowel/{u} → T = 4	alambiques, Henriques, obrigues
15	If ^(0) = {e} and ^(1) = {u} and ^(2) = {q, g} and ^(3)={u} → T = 4	açougue, azougue, tougue
16	If ^(0) = {s} and ^(1) = {e} and ^(2) = {u} and ^(3) = {q, g} and ^(4)={u} → T = 5	açougues, azougues, tougues
17	If ^(0) = {e} and ^(1) = {u} and ^(2) = {q, g } and ^(3) ={r} → T = 4	embarque, marque, morgue
18	If ^(0) = {s} and ^(1) = {e} and ^(2) = {u} and ^(3) = {q, g } and ^(4) ={r} → T = 5	embarques, marques, morgues
19	If ^(0) = {e} and ^(1) = {u} and ^(2) = {q, g} and ^(3) ={n}→ T = 4	sangue, , manque,
20	If ^(0) = {s} and ^(1) = {e} and ^(2) = {u} and ^(3) = {q, g} and ^(4) = {n} → T = 5	exangues, manques, palanques
21	If ^(0), ^(1), ^(2) are vowels, if ^(1) = {i,u} and if ^(3) is a consonant, { } → T = 2	meia, seio, apoio, aia, gaia, papagaio
22	If ^(0) = {s, m} e ^(1), ^(2), ^(3) are vowels, if ^(2) = {i,u} and if ^(4) is a consonant, { } → T = 3	meias, seios, gaias, papagaios
23	If ^(0) and ^(3) are vowels, and ^(1) is a consonant and ^(2) = {i, u} and ^(4) ≠ vowel/ {u} → T = 3	cadeira, queima, louco, estrangeiro
24	If ^(0) = {s} and ^(1) and ^(4) are vowels, and ^(2) is consonant and ^(3) = {i, u} and ^(5) ≠ vowel/{u} → T = 4	cadeiras, queimas, loucos, estrangeiros
25	If ^(0) = {a,e,o} and ^(1) is consonant and ^(2) = {n} and ^(3) = {i,u} and ^(4) is vowel→ T = 3	ainda, caindo, incluindo, oriundo
26	If ^(0) = {s} and ^(1) = {a,e,o} and ^(2) is consonant and ^(3) = {n} and ^(4) = {i,u} and ^(5) is vowel→ T = 4	Oriundos
27	If ^(k)[3] = penultimate vowel and ^(k) = {i,u} and ^(k+1) is a vowel and^(k-1) is not a vowel and ^(k+2) is not {q,g} → T = k+1	outro, claustro
28	If ^(0) = {m} and^(1) = {e} and ^(2) = {u} and ^(3) = {q} → T = 1	Quem
29	If ^(0) = {a,o, e} and ^(1) = {i, u} and ^(2) is a cons. or {u} → T = 1	inicie, assobio, continua, rua
30	If ^(0) = {s, m} and ^(1) = {a,o, e} and ^(2) = {i, u} and ^(3) is a consonant or{u} → T = 2	academias, continuam, inicием
31	If none of the above rules occur → T = penultimate vowel of the word	casa, homem, guerra

Table 2. *Rule set for word stress marker in European and Brazilian Portuguese.*

3. IMPLEMENTATION, TESTS AND RESULTS

The word stress marker was programmed in C/C++ for Windows. A graphic interface was built in Borland Delphi with the purpose of testing the performance of this application. The stress output is combined with the syllabification output, as can be seen in Figure 1. The automatic syllabification application shown in Figure 1 was already described in [8]. Syllabification and stress prediction make stress information be envisaged in a syllable unit context and not only in a vowel context. This way, vocalic stress prediction can be extended to syllabic stress prediction. The word stress marker is part of a larger application which is basically the Portuguese HTS front-end, presented in [9]. Two tests were

[1] Exception to this rule: in the following words <àquele, àqueles, àquela, àquelas, àqueloutro, àqueloutra, àqueloutros, àqueloutras>, the orthographic accent <`> should not be considered as an accent. This is an accent that marks the contraction between two words, not a phonological stress.

[2] Exceptions to this rule occur in words ending by the suffixes <-inho>,<- inha>,<- inhos>,<- inhas>,<-zinho>, <-zinha>,<-zinhas>,<- zinhos> (e.g. pãezinhos, sotãozinho) or <-mente> (e.g. cristãmente), in which the stressed vowel becomes the penultimate syllable, although the accented vowel still has a secondary accent.

[3] (k) is a variable, a given grapheme.

conducted in order to assess the performance of the automatic word stress marker with corpora from both varieties of Portuguese (European and Brazilian). The first test was carried out using 1000 sentences as input, randomly extracted from Cetem-Público European Portuguese (EP) newspaper corpus [10], containing 8052 words and 41156 characters without spaces. The second test was conducted using 500 sentences, randomly extracted from Ceten-Folha Brazilian Portuguese newspaper corpus [11] and composed by 5372 words and 28633 characters without spaces. The overall results show a very similar performance of the word stress marker, giving rise to 99.59% of accuracy rate for European Portuguese and 99.60% of accuracy rates to Brazilian Portuguese. Table 3 shows the results of the word stress marker using Cetem-Público European Portuguese corpus. The word error rate (hereafter WER) is 0.41%, from which 0.35% occur in foreign words. In fact, foreign words are the main cause for errors in our system. In Figure 2, a detailed display of errors according to their origin can be seen. English origin words in Portuguese language have the highest percentage of errors (0.11%), as shown in Figure 2, and occur in words such as <internet> (in_te1r_net) or <cocktail> (coc_kta_i1l). This result is explained by the high frequency of English origin words in the Portuguese vocabulary.

Type of error	# errors	% errors
Foreign words	28	0.35
Portuguese words	5	0.06
Total	**33**	**0.41**

Table 3. *Results of word stress marker using European Portuguese corpora.*

With the same number of errors (0.11%), we can find foreign words from other origins, of which <jihad> (ji1_had) or <Arafat> (a_ra1_fat) are examples. Italian, French and Latin origin words are responsible for 0.04% of errors each, and can be found in brands <Lamborghini> (lam_bor_ghi_ni1), proper names <Pausini> (pau_si_ni1), <Jacques> (jac_qu1_es), <Chirac> (chi1_rac) and compound expressions <ex-libris> (e1x li_bri1s). The Portuguese words' category includes errors in stress prediction of the words <Coimbra> (co1im_bra) and <Quercus> (quer_cu1s), which were repeated several times in the corpus. The comparison of our results with other accuracy rates reported in literature using statistical methods (85.57% stated in [5]) seems to demonstrate the better performance of a linguistically rule-based approach when tackling the word stress prediction. In Table 4, the results of the word stress marker using Ceten-Folha Brazilian Portuguese corpora are presented. Once more, foreign words such as <Corinthians> (co_rin_thi1_ans) are the major cause for the system WER (being responsible for 0.31% of the errors). The word <porque> is the second error cause. Although in European Portuguese this word is stressed in the second last syllable (<porque>), in Brazilian Portuguese this word is stressed in the last syllable (<porque>). This difference will be included in the system and treated as an exception. One error occurred in a readable acronym <Telesp> (te1_lesp), because it shows an unpredicted final graphic pattern. These results represent a great improvement when compared with others, previously described in literature (98.58% of accuracy rate in [6]).

Type of error	# errors	% errors
Foreign words	17	0.31
<porque>	4	0.07
Acronyms	1	0.01
Total	**22**	**0.40**

Table 4. *Results of word stress marker using Brazilian Portuguese corpora.*

4. APPLICATIONS TO GALICIAN

The common historical origin between Portuguese and Galician and their linguistic proximity led us to test the proposed word stress marker with Galician corpora. The selected corpus was composed by 300 sentences, 2627 words and 12250 characters without spaces, randomly extracted from CORGA - Corpus de Referencia do Galego Actual [12]. This corpus is a collection of different sources (oral and written) and genders (newspapers, literature, magazines, etc.). No adaptation of the here described word stress marker was made to Galician language, except in the non stressed words' list. The analysis of the results using Galician texts was based in the requirements presented in [13]. In Table 5, it can be seen that the accuracy rate of the automatic Portuguese word stress marker when tested with Galician texts is 98.52%. This encouraging result not only demonstrates a very similar phonological structure between Portuguese and Galician, but also proves the high applicability of this module to a different romance language without any algorithm adaptation.

Type of error	# errors	% errors
For lack of accent in Galician	34	1.29
Foreign words	2	0.08
Others	3	0.19
Total	**39**	**1.48**

Table 5. *Results of word stress marker for Galician.*

Most of the errors shown in Table 5 are due to the fact that in Galician, because of the Spanish orthography influence, there is no graphical accent in words ending with diphthongs like /jo/ or /ja/ (e.g. <contrario, media, principio, Emilio, circunstancia>), because these words are considered to be stressed in the penultimate syllable. However, the same words exist in Portuguese but are considered to be stressed in the antepenultimate syllable, which means that the Galician diphthongs are considered to be two syllables in fact. Hence, these

words in Portuguese receive a graphical accent (e.g. <contrário, média, princípio, Emílio, circunstância>). Therefore, the high rate of errors (1.29%) in Table 5 can be explained because the graphical accent is essential to the identification of the tonic syllable and Galician doesn't have it in the same contexts as Portuguese does. Anyway, according to [14], there is a trend in Galician oral language to pronounce these words like in Portuguese, in other words, separating these final diphthongs in two syllables. In order to solve these errors in a great extent we could propose the following rules, as shown in Table 6:

#	Rule	Example
1	If ^(0) = {a,o} and ^(1) = {i} and ^(2) is consonant and ^(3) = V → T = 3	contrario, media, principio
2	If ^(0) = {s} and ^(1) = {a,o} and ^(2) = {i} and ^(3) is consonant and ^(4) = V → T = 4	contrarios, medias, principios
3	If ^(0) = {a,o} and ^(1) = {i} and ^(2) is consonant and ^(3) = {m,n} and ^(4) = V → T = 4	circunstancia
4	If ^(0) = {s} and ^(1) = {a,o} and ^(2) = {i} e ^(3) is consonant and ^(4) = {m,n} and ^(5) = V → T = 5	circunstancias

Table 6. *Stress marking rules to solve words ending by diphthongs /jo/ and /ja/ in Galician.*

The proposed rules in Table 6 would be able to raise the current accuracy rate to 99.81%. Other errors occur in foreign words (0.08%), similarly to Portuguese, in words like <Madrid> (ma1_drid) and <chofer> (cho_fe1r) and in abbreviations, such as <mili (abbreviation of "servicio militar")> (mi_li1). These encouraging results and small refinements allow us to conclude that our system and approach are highly applicable to other languages in general and to romance languages in particular.

5. CONCLUSIONS

In this article, a linguistically rule-based automatic word stress marker for European and Brazilian Portuguese was described, implemented and tested. The purpose of this work was to provide stress information to the front-end part of the TTS system (syllable boundary marker and grapheme-to-phone(me) transcriber) and to the training corpora used by the HTS back-end, since it was proved in [1] that this information improves synthetic naturalness. The proposed automatic word stress marker deals with 31 rules and starts analyzing the last grapheme of a word. The goal is to identify the tonic vowel of each word. Combined with the automatic syllabification information, the stress information can affect the entire syllable and not only the stressed vowel. This approach proved to be very efficient giving rise to very encouraging accuracy rates when tested with real text corpora: 99.59% with European Portuguese corpora and 99.60% with Brazilian Portuguese corpora. This system was also experimented with Galician corpora with a small adaptation in the non stressed word list, giving rise to 98.52% of accuracy rate. A refinement of these results based on the errors' analysis was proposed. Due to the success of application of the system presented in this paper to European Portuguese, Brazilian Portuguese and Galician, we believe that this approach can be easily adapted to other languages. The application of this work to Catalan was already done with similar success [15] and other languages are envisaged.

6. REFERENCES

[1] Maia, R.: Speech Synthesis and Phonetic Vocoding for Brazilian Portuguese Based on Parameter Generation form Hidden Markov Models. PhD thesis. Department of Computer Science and Engineering, Nagoya Institute of Technology, Nagoya, Japan (2006)

[2] Mateus, M., Andrade, E. *The Phonology of Portuguese*. Oxford University Press, Oxford, 2000.

[3] Oliveira, L., Viana, M., Trancoso, I."DIXI - Portuguese Text-to-Speech System", Proceedings of EUROSPEECH'91 - 2nd European Conference on Speech Communication and Technology, pp.1239-1242. Genoa, Italy, 1991.

[4] Teixeira, J. P., Freitas, D."MULTIVOX- Conversor Texto-Fala para Português", Lima, V. (eds.) III Encontro para o Processamento Computacional do Português Escrito e Falado (PROPOR'98), pp. 88-98. Porto Alegre, RS, Brazil, 1998.

[5] Barros, M., Weiss, C. "Maximum Entropy Motivated Grapheme-to-Phoneme, Stress and Syllable Boundary Prediction for Portuguese Text-to-Speech", IV Jornadas en Tecnologías del Habla, pp. 177-182. Zaragoza, España, 2006.

[6] Silva, D., Lima, A., Maia, R., Braga, D., Moraes, J. F., Moraes, J. A., Resende Jr., F. "A rule-based grapheme-phone converter and stress determination for Brazilian Portuguese natural language processing", VI International Telecommunications Symposium (ITS2006), pp.550-554. Fortaleza-CE, Brazil, 2006.

[7] Cunha, C., Cintra, L. *Nova gramática do português contemporâneo*. Sá da Costa, Lisboa, 1992.

[8] Braga, D., Resende Jr., F. G. V.: "Módulos de Processamento de Texto Baseados em Regras para Sistemas de Conversão Texto-Fala em Português Europeu", XXI Encontro da Associação Portuguesa de Linguística, pp.141-156. Coimbra, Portugal, 2007.

[9] Braga, D. *Algoritmos de Processamento da Linguagem Natural para Sistemas de Conversão Texto-Fala em Português*. PhD Thesis. Universidade da Coruña, España, 2008.

[10] Cetem-Público,http://www.linguateca.pt/CETEMPublico/

[11] Ceten-Folha, http://www.linguateca.pt/CETENFolha/

[12] Corpus de Referencia do Galego Actual, http://corpus.cirp.es/corga/

[13] Real Academia Galega/ Instituto da Língua Galega: Normas ortográficas e morfolóxicas do idioma galego. Real Academia Galega/ Instituto da Língua Galega, Vigo, España, 2003.

[14] Freixeiro Mato, X. R. *Manual de Gramática Galega*. Edicións a Nosa Terra, Vigo, 2006.

[15] Rustullet, S.; Braga, D.; Nogueira, J.; Dias, M. "Automatic Word Stress Marking and Syllabification for Catalan TTS", Proceedings of Interspeech 2008, Brisbane, Australia, 2008.

DIALOG ACT LABELING IN THE DIHANA CORPUS USING PROSODY INFORMATION

Vicent Tamarit, Carlos-D. Martínez-Hinarejos

Instituto Tecnológico de Informática, Universidad Politécnica de Valencia
Camino de Vera s/n, 46022, Valencia, Spain

ABSTRACT

We propose a dialog act classification based on the prosody of the audio signal in combination with the course of the dialog. The work is applied to the Spanish corpus DIHANA. As far as we know, it is the first experiment made with prosody in this corpus. To do the labeling, we used two features that had been extracted from the user speech (pitch and energy) in a HMM classifier combined with an n-gram of dialog acts. The results shows a sightly improvement in the tagging when prosody is included in the classification.

1. INTRODUCTION

In a speech based dialog system, it is necessary to recognize the user's speech and to understand the true meaning of the uttered sentence. Both of them are used by the machine to generate an appropriate response. When obtaining the relevant information that aids the system, the speech is segmented into utterances (the minimal significant unit from the dialog viewpoint) each of which is labeled with a dialog act (DA). Dialog acts typically represent types of sentences or communicative intentions. Common dialog acts are: question, answer, response,... One of the research fields in dialog systems is the identification of dialog acts.

One way to extract dialog acts from speech is using the automatic speech transcription, so the recognition accuracy may affect the correct labeling. To improve the tagging other authors have proposed the use of some prosody features that can identify different types of sentences. On the one hand, this method has one important advantage: it could be used before the speech recognition, and the dialog act identification may aid the speech recognizer in the recognition of the words; but, on the other hand, the signal is more difficult to interpret than the transcription.

Some studies have proved the influence of prosody in dialog acts identification. In [1], results are presented for the SwitchBoard corpus, based on spontaneous conversations between English speakers. These results show an improvement on the DA identification when using prosodic features. The CallHome Spanish corpus, with telephonic conversation in Latin American Spanish, has been used in a similar test [2]. In this last work, pitch and energy features are computed to classify acts through Support Vector Machines (SVM).

In this article we describe the results of dialog act labeling in the Spanish corpus DIHANA, using pitch and energy values. This corpus is recorded only by Spanish speakers, seeking those who do not have a strong accent. Instead of SVM or K-Nearest Neighbours (K-NN) techniques, that do not capture the continuity of the features, we used Hidden Markov Models (HMM) with gaussian output distributions, in a similar way to the speech recognition process. Furthermore, we improved the prosodic classification with a dialog act n-gram.

2. DESCRIPTION OF THE CORPUS

The Spanish corpus DIHANA [3] is composed of 900 dialogs about a telephonic train information system. It was acquired by 225 different speakers (153 male and 72 females), with small dialectal variants. There are 6,280 user turns and 9,133 system turns. The vocabulary size is 823 words. The total amount of speech signal was about five and a half hours.

The acquisition of the DIHANA corpus was carried out by means of an initial prototype, using the Wizard of Oz (WoZ) technique [4]. This acquisition was only restricted at the semantic level (i.e., the acquired dialogs are related to a specific task domain) and was not restricted at the lexical and syntactical level (spontaneous-speech). In this acquisition process, the semantic control was provided by the definition of scenarios that the user had to accomplish and by the WoZ strategy, which defines the behaviour of the acquisition system.

The annotation scheme used in the corpus is based on the Interchange Format (IF) defined in the C-STAR project [5]. Although it was defined for a Machine Translation task, it has been adapted to dialog annotation [6]. The three-level proposal of the IF format covers the speech act, the concept, and the argument, which makes it appropriate for its use in task-oriented dialog.

Based on the IF format, a three-level annotation scheme of the DIHANA corpus utterances was defined in [7].

WORK SUPPORTED BY THE EC (FEDER) AND THE SPANISH MEC UNDER GRANT TIN2006-15694-CO2-01.

This DA set represents the general purpose of the utterance (first level), as well as more precise semantic information that is specific to each task (second and third levels).

All of the dialogues are segmented in turns (User and System), and each turn is also segmented into utterances. Finally, each utterance is labelled with a three-level label. Obviously, more than one utterance can appear per turn. In fact, an average of 1.5 utterances per turn was obtained.

Only the first level contains linguistic information that can be learned by a prosodic classifier, so we preprocessed the audio corpus to cut the turns into first level utterances. The final tags we used were: Afirmación (Yes-answer), Negación (No-answer), Pregunta (Question), Respuesta (Generic Answer), Cierre (End dialog), Indefinida (No tagged). The $42\,\%$ of the first level utterances are questions; thus, our baseline classification error is $58\,\%$. There are 7,373 first level utterances. We used, on average, 6,118 for train and 1,255 for test.

3. MODELING

Other authors have used some different techniques to classify DAs from prosodic cues, like decision trees [1], neural networks [8], and SVM[2]. We investigated the performance of HMMs for classification based on prosody.

3.1. Feature extraction

There is a classification of sentences in Spanish based on the speaker intonation [9]. The intonation of the sentences are quite different, i.e., for a question or for a statement. In Figures 1 and 2, we show the pitch evolution for one sentence with different intonations.

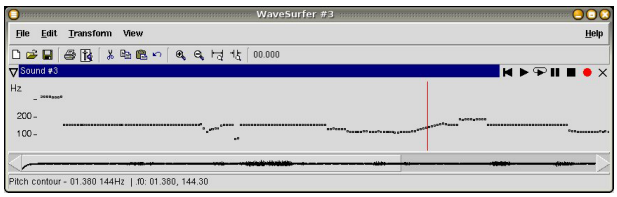


Figure 1. Pitch evolution for the sentence ”Se puede ir desde Santurce a Bilbao” recorded like a statement.

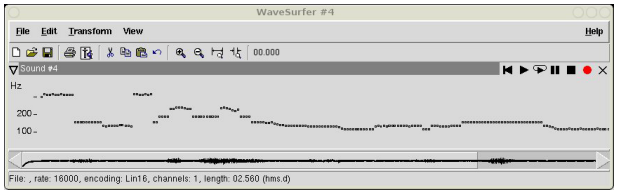


Figure 2. Pitch evolution for the sentence ”Se puede ir desde Santurce a Bilbao” recorded like a question.

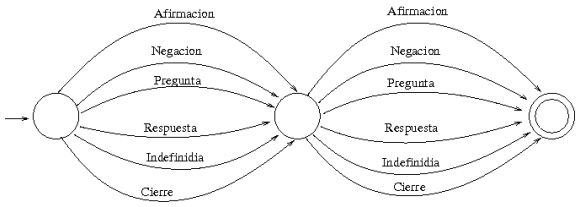


Figure 3. Finite State Machine used as language model in the task.

The feature extraction is based in only two features: energy measure and pitch. For every 10 ms of signal, we computed the frame energy and an estimation of the F0. We used the Snack library [10], developed by the Department of Speech, Music and Hearing, in the Royal Institute of Technology in Sweden, to estimate the fundamental frequency. The purpose of the library is to develop in a short time sound tools using scripting languages such as Tcl/TK or Python. In addition to these two values, we computed the first and second derivative of the features. Therefore, we obtained vectors with six elements.

3.2. Hidden Markov Models

The HMMs with gaussian distributions in the states are used in speech recognition to model the sound units, usually phonemes. Their structure allows them to model the time-variation of the features so they can represent the prosodic variation in time.

We used a three-state HMM for each dialog act. This structure was selected due to the minimum number of vectors obtained from the audio signal. They were trained using the HTK software [11].

The decodification process was made using iATROS. This recognition software was developed in the PRHLT Group in the Instituto Tecnológico de Informática. It is based on the Viterbi algorithm and uses three models:

- Acoustic models: Each model represents a phonetic unit as a continuous HMM. In our task, we have one acoustic model for each dialog act which represents the prosodic variation.
- Lexical model: Each word is described as a Finite State Machine (FSM), that defines the acoustic models that compose the word. In our case, no words are actually defined. Therefore, each lexical model correspond to an only acoustic model.
- Language Model: Defines the relations between the words. We used a FSM to model the structure of a turn. Figure 3 shows the model we used. It has three states, since the utterances have two dialog acts at most. The transition probabilities between states are equal for all the edges.

```
Pregunta|Respuesta Indefinida 0.02 -3.9121
Pregunta|Respuesta Afirmacion 0.22 -1.5141
Pregunta|Respuesta Pregunta 0.34 -1.0788
Pregunta|Respuesta Respuesta 0.27 -1.3093
Pregunta|Respuesta Cierre 0.07 -2.6569
Pregunta|Respuesta Negacion 0.07 -2.6569
```

Figure 4. Some estimations of the 3-gram used in this task. The second number is the log-probability.

Obviously, some turns have only one tag. We used partial decodification to solve this problem and allow decodification with only one utterance.

3.3. Word Graphs

A word graph (WG) is a direct graph, no cyclical and with weights, where each node represents a discrete point in time. The edges of the graph are a set [w, s, e], where w is the hypothetic word from node s to e. The weights are scores associated to the edges. The best path from the initial state to the final state is the most likely hypothesis. In short, a WG is like a "picture" of the recognition process.

The word graph is necessary to add n-gram information to the model. The n-gram is estimated using the sequence of dialog acts in the dialogues, i.e., using system and user turns. In Figure 4 is showed a example of 3-gram. However, recognition is only performed on user turns (audio records from system utterances does not exist). Therefore, after the recognition process we obtain a WG with the calculated acoustic probabilities and equal language model probabilities, which includes all possible dialog act sequences. We can incorporate the n-gram information by changing the language model probabilities in the WG by the corresponding n-gram probabilities. After this change, we searched the best path in the WG using the combination of the acoustic and the new language model probabilities.

In Figure 5 there is an example of a word graph for this task. In this case there is only one dialog act, and the graph shows us the probabilities for each class. In the shown WG, lets assume that the previous dialog acts for that turn were "Pregunta" and "Respuesta". The new WG with n-gram probabilities (using the probabilities of Figure 4) is showed in Figure 6.

4. RESULTS

To obtain significant results in the labeling task with the DIHANA corpus, a cross-validation approach was adopted and 5 different partitions were used. Each of them had 720 dialogues for training and 180 for testing. The statistics for the corpus are presented in Table 1.

For each partition we combined the prosody classification with a 3-gram trained with the utterances' evolution within the dialogs. The 3-grams included all the utterances (user and system), because in a real dialog system we always know the tag of the previous system turn.

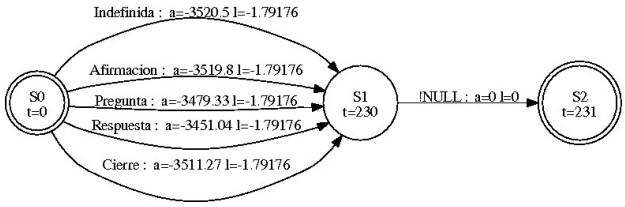


Figure 5. Example of word graph for this task. Each edge is represented with the label and the log-probability of the acoustic model (a) and language model (l).

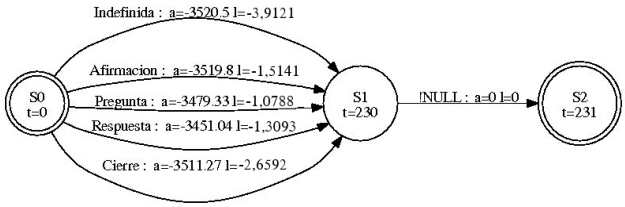


Figure 6. The original language model probabilities are replaced by the n-gram ones.

Table 2 shows the results of the experiments. We included the tagging using only the calculated 3-grams, and the combination with prosody. In this table, the Word Error Rate (WER) measures the accuracy of the act labeling, whereas the Sentence Error Rate (SER) shows the accuracy of the whole turn tagging.

Labeling acts using the 3-gram produced an improvement of 20 points from the baseline (that we fixed in 58 %). The inclusion of prosody information reduces the WER only in one point.

	Training		
	User	System	Total
Dialogues	720		
Turns	5,024	7,206	12,330
Running words	42,806	119,807	162,613
Vocabulary	762	208	832

	Test		
	User	System	Total
Dialogues	180		
Turns	1,256	1,827	3,083
Running words	10,815	29,950	40,765
Vocabulary	417	174	485

Table 1. DIHANA corpus statistics (average of the five cross-validation partitions).

WER/SER	3-gram	Combined
Partition 1	41.4/40.2	40.5/39.3
Partition 2	43.5/42.2	42.8/41
Partition 3	40.8/39	39.8/37.6
Partition 4	40.9/39.5	40.6/39.4
Partition 5	34.5/33.2	32.4/30.7
Total	40.3/38.9	**39.3/37.6**

Table 2. Results for the experiments in the five partitions.

5. CONCLUSIONS AND FUTURE WORK

The corpus DIHANA has a labeling oriented to the human-machine interaction. This tagging is useful for the system to understand the requests and generate a response, but it is not based on the intonation of the sentence. This task-oriented labeling could be the reason of the little improvement in the classification using our prosody-based classifier. As far as we know this is the first time prosody is used in the dialog act classification in the DIHANA corpus, so we can not conclude that prosody does not improve the dialog act tagging, as more experiments should be performed.

Future work is directed to improve the intonation extraction, as can be seen in [12], and test new prosody features in this corpus, such as those proposed in [13], as well as other classification techniques like K-NN, neural networks or decision trees, which are proved in other corpora, but not in DIHANA. The classification structure based on HMMs could be applied on other corpora like CallHome or SwitchBoard. These corpora are annotated with a different set of dialog acts that could be more suitable for the prosody-based classifier. The use of a Spanish corpus annotated with labels based on the intonation of the sentences may help us to determine the utility of the prosody in Spanish. Restructuring the dialog acts in DIHANA is another possibility.

6. REFERENCES

[1] E. Shrinberg, R. Bates, P. Taylor, A. Stolcke, D. Jurafsky, K. Ries, N. Coccaro, R. Martin, M. Meteer, and C. Van Ess-Dykema, "Can prosody aid the automatic classification of dialog acts in conversational speech?," *Language and Speech. Special Issue on Prosody and Conversation*, vol. 41 (3-4), pp. 439–487, 1998.

[2] Raul Fernandez and Rosalind W. Picard, "Dialog act classification from prosodic features using support vector machines," *Speech Prosody 2002, International Conference*, pp. 291–294, 2002.

[3] J.-M. Benedí, E. Lleida, A. Varona, M.-J. Castro, I. Galiano, R. Justo, I. López de Letona, and A. Miguel, "Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: Dihana," *Fifth International Conference on Language Resources and Evaluation (LREC)*, pp. 1636–1639, May 2006.

[4] M. Fraser and G. Gilbert, "Simulating speech systems," *Computer Speech and Language*, , no. 5, pp. 81–89, 1991.

[5] Lavie A., L. Levin, P. Zhan, M. Taboada, D. Gates, M. M. Lapata, C. Clark, M. Broadhead, and A. Waibel, "Expanding the domain of a multi-lingual speech-to-speech translation system," *Proceedings of the Workshop on Spoken Language Translation, ACL/EACL-97*, 1997.

[6] T. Fukada, D. Koll, A. Waibel, and K. Tanigaki, "Probabilistic dialogue act extraction for concept based multilingual translation systems," *ICSLP 98*, pp. 2771–2774, 1998.

[7] N. Alcácer, J. Benedí, F. Blat, R. Granell, C. D. Martínez, and F. Torres, "Acquisition and labelling of a spontaneous speech dialogue corpus," *Proceeding of 10th International Conference on Speech and Computer (SPECOM). Patras, Greece*, pp. 583–586, 2005.

[8] Finke M. amd Lapata M., "Clarity:infering discourse structure from speech," *Proc AAAI '98 Spring Symposium on Applying Machine Learning to Discourse Processing*, 1998.

[9] Antonio Quilis and Joseph A. Fernández, *Curso de fonética y fonología española*, CSIC, 1993.

[10] K. Sjölander and J. Beskow, "Wavesurfer - an open source speech tool," *Proc of ICSLP, Beijing*, pp. 464–467, October 2000.

[11] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*, CUED, UK, v3.2 edition, July, 2004.

[12] Elizabeth Shriberg, Andreas Stolcke, Dilek Hakkani-tür, and Gökhan Tür, "Prosody-based automatic segmentation of speech into sentences and topics," *Speech Communication*, vol. 32, pp. 127–154, 2000.

[13] A. Stolcke, N. Coccaro, R. Bates., P. Taylor, C. Van Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer, "Dialogue act modelling for automatic tagging and recognition of conversational speech," *Computational Linguistics*, vol. 26, no. 3, pp. 1–34, 2000.

EVALUACIÓN DE CAMPO DE UN SISTEMA DE DIÁLOGO ORAL EMPLEANDO RELACIONES ESTADÍSTICAS

Zoraida Callejas, Ramón López-Cózar

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada. 18071 Granada
{zcallejas,rlopezc}@ugr.es

RESUMEN

La evaluación de sistemas de diálogo oral se puede realizar mediante medidas "objetivas" calculadas de forma instrumental o mediante expertos y con juicios de opinión de los usuarios que hayan empleado el sistema con anterioridad (medidas "subjetivas"). En la literatura podemos encontrar diversos trabajos que tratan de establecer relaciones entre ambos tipos de medidas. En este artículo describimos los resultados empíricos obtenidos de estudios estadísticos sobre interacciones de usuarios reales con un sistema de diálogo experimental. Estos estudios no han sido suficientemente explorados en la literatura y como demostramos, pueden mostrar relaciones importantes entre criterios de evaluación, que pueden servir de guía para refinar los sistemas que se evalúan, así como para contribuir al conocimiento acerca de cómo los aspectos cuantitativos pueden afectar la percepción del usuario acerca del sistema.

1. INTRODUCCIÓN

Con el fin de minimizar costes y optimizar resultados, existe la necesidad de encontrar métodos, arquitecturas y criterios estándar para evaluar, comparar y predecir el rendimiento y la usabilidad de los sistemas de diálogo. Numerosas investigaciones realizadas durante los años 90 (p.e. [1][2]) sentaron las bases para establecer un conjunto común de criterios cuantitativos de evaluación. Sin embargo, no existe un consenso global acerca de qué criterios deben tenerse en cuenta para optimizar la usabilidad de los sistemas de diálogo. Algunos proyectos han intentado abordar el problema de la predicción de la usabilidad y la satisfacción del usuario a partir de criterios de rendimiento medibles. Este es el caso del modelo PARADISE [3], que se ha convertido en uno de los modelos de referencia para la evaluación de este tipo de sistemas.

Debido a la complejidad y el esfuerzo que demanda la aplicación de este modelo, muchos autores aplican medidas cualitativas y cuantitativas por separado. Por ejemplo, el sistema multimodal de navegación MUMS [4], el sistema VIrtual CO-driver [5], el quiosco multimedia MASK [6] y el sistema de diálogo SAMMIE [7], se han evaluado únicamente de forma subjetiva. Otros autores, p.e. [8], evalúan sus sistemas tanto con criterios medidos instrumentalmente como con opiniones de los usuarios acerca de su calidad, pero sin establecer enlaces entre las distintas medidas de evaluación empleadas.

Por otra parte, los resultados en la literatura están por lo general basados en interacciones restringidas en laboratorio, en las que se solicita a los usuarios que interactúen con el sistema siguiendo unos escenarios previamente establecidos. La principal desventaja de este método es que dichos escenarios pueden diferir de las tareas que un usuario habría seleccionado en una interacción no predefinida. Por el contrario, la evaluación de campo se realiza a partir de interacciones de usuarios reales con el sistema final en sus entornos reales. Los resultados obtenidos mediante evaluaciones de campo son robustos ante la heterogeneidad de usuarios, dispositivos y entornos; por consiguiente, son más relevantes para la predicción del comportamiento real de los sistemas que los estudios de laboratorio.

La contribución del artículo al estado del arte en la evaluación de sistemas de diálogo oral consiste en la obtención de nuevas evidencias empíricas por medio de un estudio de campo llevado a cabo sobre nuestro sistema de diálogo experimental UAH (Universidad al Habla). Para ello se han empleado estudios de correlación, estableciendo relaciones entre criterios tanto cuantitativos como cualitativos.

2. CRITERIOS DE EVALUACIÓN

Este artículo presenta los resultados de evaluación del sistema UAH, que se desarrolló para proveer acceso telefónico automático a información académica de nuestro Departamento [9]. La evaluación del sistema se ha llevado a cabo tanto con parámetros de interacción como con juicios de calidad. Los primeros han sido extraídos de forma semi-automática a partir de los diálogos grabados, mientras que los segundos se han obtenido de cuestionarios que los usuarios podían rellenar de forma voluntaria.

Para calcular los parámetros de interacción, hemos empleado un corpus de diálogos construido a partir de las llamadas telefónicas realizadas al sistema UAH por alum-

Este trabajo ha sido subvencionado por el proyecto HADA TIN2007-64718 (Ministerio de Educación y Ciencia).

nos de nuestra Universidad durante su primer año de utilización. Este corpus consta de 85 diálogos y 422 turnos de usuario, con una media de 5 turnos por diálogo. Los parámetros de evaluación empleados han sido los siguientes: éxito de la tarea, completitud del diálogo, duración del diálogo, número de turnos de usuario, media de palabras por turno, WER, confianza de reconocimiento media, porcentaje de elocuciones correctamente comprendidas, y número de turnos de confirmación.

Las medidas de evaluación subjetivas que hemos empleado han sido las siguientes: percepción de hasta qué punto UAH entiende al usuario, percepción de hasta qué punto el usuario entiende a UAH, velocidad de interacción percibida, presencia percibida de errores cometidos por UAH, facilidad percibida de corregir los errores de UAH, facilidad percibida de conseguir la información requerida, satisfacción del usuario, percepción de hasta qué punto el usuario sabía qué hacer en cada momento de la interacción y percepción de hasta qué punto UAH se comportaba de forma similar a un ser humano. Además, también se ha extraído de los cuestionarios el nivel de conocimiento técnico de los usuarios y su experiencia previa utilizando el sistema.

3. ESTUDIOS ESTADÍSTICOS

Para encontrar relaciones relevantes entre los criterios utilizados, hemos correlacionado todas las variables, obteniendo el valor absoluto del *coeficiente de correlación de Pearson*, así como la significatividad (o *p-value*) de cada coeficiente de correlación.

Dado que la mayoría de las variables estaban intercorrelacionadas, se ha estudiado el efecto que cada criterio ejercía en la significatividad de las relaciones entre los demás criterios. Para estudiar las relaciones aisladamente, eliminando el efecto del resto de los criterios, hemos medido los *coeficientes de correlación parcial* conjuntamente con sus niveles de significación.

El *coeficiente de correlación de Pearson* funciona correctamente para las variables escalables (p.e. la duración del diálogo), pues es apropiado realizar comparaciones de distancia entre valores. Sin embargo, para la investigación descrita también se han empleado variables ordinales (p.e. parámetros de calidad percibida) y dicotómicas (p.e. "éxito de la tarea" o "completitud del diálogo"). Para obtener resultados fiables, se han generado tablas de contingencia para los criterios ordinales así como los coeficientes *Tau-b de Kendall* y *Rho de Spearman*.

Además, hemos llevado a cabo análisis de varianza utilizando el test *one-way ANOVA* junto con el *coeficiente F*. Para obtener más información en la que basar las interpretaciones realizadas, y especialmente para el caso de las variables dicotómicas, también se ha calculado el valor de la *V de Cramer*, que permite contrastar la hipótesis de independencia en las tablas de contingencia.

4. DISCUSIÓN DE LOS RESULTADOS

Los dos valores más altos de correlación con la satisfacción del usuario han sido obtenidos en todos los estudios estadísticos para los criterios siguientes: "facilidad percibida de conseguir la información requerida" y "éxito de la tarea". Según lo esperado, la satisfacción del usuario es alta cuando éste consigue fácilmente la información que requiere. Sin embargo, cabe destacar que el modo en que los usuarios obtienen la información tiene respecto a su satisfacción, la misma significatividad que el hecho de que finalmente consigan dicha información. En [10], la satisfacción del usuario también está correlacionada con que el usuario obtuviera finalmente la información que buscaba. Sin embargo, el indicador de Möller de facilidad de la comunicación no proporcionaba una contribución significativa a la satisfacción total del usuario. Este hecho puede sugerir que la facilidad de la comunicación es más importante para los usuarios que tienen una necesidad verdadera de obtener la información (estudios de campo), que para quienes la interacción se realiza siguiendo escenarios predefinidos (estudios de laboratorio).

Además, Rajman et al. [11] mostraron que, dado que los usuarios en evaluaciones de laboratorio no tienen la posibilidad de contrastar la información proporcionada por el sistema de diálogo, éstos confían ciegamente en las respuestas del sistema. Es decir, no comprueban si la información es correcta o útil, y por tanto, consideran el hecho de obtener una respuesta del sistema equivalente a obtener un resultado correcto. En nuestros experimentos, se ha proporcionado a los usuarios información académica real. Dado que necesitaban realmente esta información, podían contrastarla y saber si era exacta o no. Así, entre los diálogos no exitosos (tanto desde el punto de vista de los parámetros de la interacción como de las valoraciones sobre la calidad) se han dado casos donde a pesar de que el sistema proporcionó al usuario información, ésta no era la que él deseaba, como demuestra el hecho de que algunos diálogos completos no fueron exitosos. La evaluación de campo presenta, de este modo, la gran ventaja de posibilitar una separación entre la calidad de la interacción y la calidad de los resultados obtenidos.

Centrándonos en los parámetros de la interacción, hay una correlación notable entre la completitud del diálogo y el éxito de la tarea. Aunque los usuarios podían finalizar la llamada en cuanto recibían la información deseada, éstos esperaron generalmente hasta el final en los diálogos exitosos. Este hecho difiere de los resultados de otros autores. Por ejemplo, Turunen et al. [12] mostraron que había diferencias significativas entre la forma de llevar a cabo la interacción en las pruebas de laboratorio y en evaluaciones de campo con el sistema Stopman. En su evaluación de campo menos de un 10 % de los usuarios esperaron al final de la llamada antes de colgar. El número de diálogos en los cuales los usuarios esperaron hasta el final de la interacción (es decir, el número de diálogos completos) en nuestro estudio de campo es un 50 % mayor que el mos-

trado en [12], seguramente debido a una actitud "tecnofílica" de nuestros usuarios, en su mayoría estudiantes de la Escuela de Ingeniería Informática y Telecomunicaciones.

Otro criterio que está estrechamente correlacionado con el éxito de la tarea y la satisfacción del usuario es la facilidad percibida para corregir errores. Sin embargo, la presencia percibida de errores no se correla con ninguno de estos criterios. Esto puede deberse a que, aunque en el 48,19 % de los diálogos exitosos los usuarios han detectado errores, en la mayoría de los casos han sabido corregirlos y obtener la información que buscaban. Concretamente, un 69,23 % de los usuarios han considerado "fácil" o "muy fácil" corregir errores en los diálogos exitosos. Sin embargo, en los no exitosos, un 83,33 % de los usuarios ha manifestado que la corrección de errores era "difícil" o "muy difícil".

En [10], la opinión de los usuarios sobre si los malentendidos podrían ser aclarados fácilmente (que se clasificó como un factor que contribuía a la calidad del diálogo), no resultó ser un buen indicador de la satisfacción del usuario. Además, el autor encontró que la satisfacción del usuario no se podía predecir completamente mediante el éxito de la tarea, y sostuvo que este resultado podría ser debido a las condiciones poco realistas de la experimentación de laboratorio empleada en su investigación. Por tanto, se ha corroborado este hecho en nuestro estudio de campo, puesto que los cuestionarios subjetivos no se han podido substituir por los parámetros de la interacción empleados sin que esto supusiera pérdida de información.

Por otra parte, el criterio que ha mostrado un mayor número de correlaciones significativas ha sido la "percepción de hasta qué punto UAH entiende al usuario". Las relaciones más significativas entre esta valoración de la calidad y otros parámetros han sido obtenidos con el éxito de la tarea y la satisfacción del usuario. Cabe también destacar que el grado con el cual el usuario percibe que el sistema UAH le entiende no está correlacionado con los parámetros de la interacción que miden el funcionamiento del reconocedor del habla, como WER o medidas de confianza. Sin embargo, sí está correlacionado con el porcentaje de elocuciones correctamente entendidas, ello indica que desde el punto de vista del usuario, los errores de reconocimiento del habla no son importantes siempre y cuando las interpretaciones semánticas sean correctas y estos errores sean imperceptibles para el usuario.

4.1. Influencia de la iniciativa de gestión del diálogo

Para estudiar la influencia de la iniciativa utilizada para la gestión del diálogo, hemos repetido la experimentación comentada anteriormente, pero distinguiendo entre los diálogos con iniciativa dirigida por el sistema y los diálogos con iniciativa mixta.

El éxito de la tarea es aproximadamente igual para ambas iniciativas de gestión del diálogo. Este resultado difiere de los que se pueden encontrar en la literatura, p.e. [10], donde una iniciativa más flexible conduce a tasas de éxito considerablemente más altas. En nuestros experimentos el éxito es mayor para la iniciativa mixta, pero la diferencia entre ambas es insignificante pues el 77,77 % de los diálogos de iniciativa mixta y el 76,92 % de los diálogos con iniciativa por parte del sistema han concluido con éxito.

Sin embargo, a la luz de los resultados experimentales, el éxito de la tarea parece estar relacionado con distintos factores en cada tipo de iniciativa. De esta manera, en la iniciativa mixta la seguridad del usuario sobre qué hacer en cada momento del diálogo no está correlacionada con el éxito de la tarea, la satisfacción del usuario ni la percepción sobre la facilidad de obtener la información requerida. Por el contrario, el éxito de la tarea tiene una correlación significativa con la seguridad del usuario en los diálogos dirigidos por el sistema. Este hecho sucede probablemente porque el usuario dispone de mayor libertad en las interacciones con iniciativa mixta y, por tanto, el sistema no restringe lo que debe decir en cada momento. Sin embargo, esta situación no conduce a malos resultados de la interacción, pues el éxito de la tarea no se reduce al emplear iniciativa mixta.

Las correlaciones de la facilidad percibida de conseguir la información requerida son también muy diferentes en ambos casos. En el caso de la iniciativa por parte del sistema, está relacionada con la completitud del diálogo, el porcentaje de elocuciones correctamente comprendidas y la opinión que el usuario tiene sobre el comportamiento humano del sistema. Por el contrario, para los diálogos con iniciativa mixta, la facilidad percibida no está correlacionada con estas medidas, sino con indicadores de la duración de la interacción como la "duración del diálogo" o el "número de turnos de usuario". Igualmente sucede con la satisfacción y el éxito de la tarea, que están altamente correlacionadas con medidas de duración en interacciones con iniciativa mixta, pero no en los diálogos dirigidos por el sistema. La duración de estos diálogos está correlacionada de manera perceptible con la satisfacción del usuario, mientras que en sistemas con iniciativas más estrictas de la interacción, no es considerada tan importante por los usuarios. Además, la duración media de los diálogos es menor cuando la interacción es más flexible.

Los estudios basados en pruebas de laboratorio como los de Rajman et al. [11] no han podido percibir variaciones claras en la calidad con respecto al predominio de la iniciativa del sistema o del usuario. Además, algunas pruebas de laboratorio como las llevadas a cabo en [10] para el sistema BoRIS no pudieron encontrar ninguna relación significativa entre la iniciativa y otros parámetros de la interacción. Sin embargo, nuestros resultados demuestran que la significación de las relaciones entre los diversos criterios de evaluación, incluyendo parámetros de la interacción y valoraciones de la calidad, varía dependiendo de la iniciativa utilizada para la gestión del diálogo.

5. CONCLUSIONES

En este artículo se ha presentado un estudio de las relaciones entre varios criterios estándar de-facto para la evaluación de un sistema de diálogo oral con el que se interactúa telefónicamente. Nuestros resultados experimentales se basan en un estudio de campo que utiliza interacciones reales registradas por usuarios no reclutados previamente que han llamado espontáneamente al sistema para obtener información.

Para realizar nuestro estudio se han calculado parámetros de la interacción (o medidas objetivas) y juicios de la calidad (medidas subjetivas) empleando un corpus de las interacciones reales sistema-usuario. Se han llevado a cabo un conjunto de estudios estadísticos a partir de los cuales se han extraído relaciones significativas entre todos los criterios.

Nuestros resultados demuestran que el éxito de la tarea, la facilidad percibida de obtener la información y el punto hasta el cual el usuario percibe que el sistema le entiende están correlacionados con la satisfacción del usuario. Estos resultados sugieren que obtener la información requerida no conlleva necesariamente la satisfacción del usuario, dado que los usuarios valoraron en algunos casos que diálogos exitosos no les habían satisfecho debido a que encontraron dificultades para obtener la información que estaban buscando (a pesar de haber recibido datos que concordaban con su petición). Ésta es una de las implicaciones derivadas del uso de los estudios de campo, en los cuales los usuarios se preocupan no sólo de obtener la información que buscaban, sino también de obtenerla fácilmente y de que ésta sea correcta.

Además, la relación entre la facilidad percibida de obtener la información y otros criterios varía notablemente con la estrategia de gestión de diálogo empleada. Los datos estadísticos sugieren que la predicción de la satisfacción del usuario también depende de la iniciativa del diálogo empleada. En los diálogos con iniciativa mixta parece estar relacionada más directamente con medidas objetivas, como la duración del diálogo. Sin embargo, en diálogos más restringidos, las medidas subjetivas como el grado hasta el cual el usuario percibe que el sistema le entiende, tienen un impacto mayor. Se trata de un resultado importante que podría indicar una necesidad de adaptar los procedimientos de evaluación al tipo de interacciones que se analizan.

6. BIBLIOGRAFÍA

[1] M.A. Walker, A. Rudnicky, R. Prasad, J. Aberdeen, E. Owen Bratt, J. Garofolo, H. Hastie, A. Le, B. Pellom, A. Potamianos, R. Passonneau, S. Roukos, G. Sanders, S. Seneff, y D. Stallard, "DARPA Communicator: Cross-System Results for the 2001 Evaluation," in *Proc. of ICSLP'02*, Denver, USA, 2002, vol. 1, pp. 269–272.

[2] EAGLES, "Evaluation of Natural Language Processing Systems. Final report. Document EAG-EWG-PR2," Tech. Rep., Center for Sprogetknologi, Copenhague, Denmark, 1996.

[3] M. Walker, C. A. Kamm, y D. J. Litman, "Towards developing general models of usability with PARADISE," *Natural Language Engineering*, pp. 363–377, 2000.

[4] T. Hurtig, "Visualization and multimodality: a mobile multimodal dialogue system for public transportation navigation evaluated," in *Proc. of MobileHCI'06*, Helsinki, Finland, 2004, pp. 251–254.

[5] P. Geutner, F. Steffens, y D. Manstetten, "Design of the VICO Spoken Dialogue System: Evaluation of User Expectations by Wizard-of-Oz experiments," in *Proc. of LREC'02*, Las Palmas de Gran Canaria, Spain, 2002.

[6] L. Lamel, S. Bennacef, J.L. Gauvain, H. Dartigues, y J.N. Temem, "User evaluation of the MASK kiosk," *Speech Communication*, vol. 38, no. 1-2, pp. 131–139, 2002.

[7] T. Becker, C. Gerstenberger, I. Kruijff-Korbayova, A. Korthauer, M. Pinkal, M. Pitz, P. Poller, y J. Schehl, "Natural and intuitive multimodal dialogue for In-Car Applications: The SAMMIE System," in *Proc. of PAIS'06*, Riva del Garda, Italy, 2006, pp. 612–616.

[8] S. M. Robinson, A. Roque, As. Vaswani, y D. Traum, "Evaluation of a spoken dialogue system for virtual reality call for fire training," in *Proc. of the 25th Army Science Conference*, Orlando, USA, 2006.

[9] Z. Callejas y R. López-Cózar, "Implementing modular dialogue systems: a case study," in *Proc. of ASIDE'05*, Aalborg, Denmark, 2005.

[10] S. Möller, *Quality of telephone-based spoken dialogue systems*, Springer, 2005.

[11] M. Rajman, T. H. Bui, A. Rajman, F. Seydoux, A. Trutnev, y S. Quarteroni, "Assessing the usability of a dialogue management system designed in the framework of a rapid dialogue prototyping methodology," *Acta acustica united with acustica*, vol. 90, pp. 1906–1111, 2004.

[12] M. Turunen, J. Hakulinen, y A. Kainulainen, "Evaluation of a spoken dialogue system with usability tests and long-term pilot studies: Similarities and differences," in *Proc. of Interspeech/ICSLP 06*, Pittsburgh, USA, 2006, pp. 1057–1060.

EVALUACIÓN SUBJETIVA DE UNA BASE DE DATOS DE HABLA EMOCIONAL PARA EUSKERA

Iñaki Sainz, Ibon Saratxaga, Eva Navas, Inmaculada Hernáez, Jon Sanchez, Iker Luengo, Igor Odriozola, Eneritz de Bilbao

Aholab – Dept Electrónica y telecomunicaciones. Facultad de Ingenieria.

Universidad del País Vasco, Urkijo z/g 48013 Bilbo

Email: inaki, ibon, eva, inma, ion, ikerl, igor, eneritz @aholab.ehu.es

RESUMEN

El presente artículo describe el proceso de evaluación de una base de datos de voz emocional grabada para Euskera normalizado (Euskera *batua*). El propósito de dicha evaluación no es otro que determinar la validez de la base de datos tanto para la caracterización del habla emocional como para su empleo en el desarrollo de un sistema de síntesis de habla expresiva. El corpus está formado por setecientas frases semánticamente neutras que han sido grabadas por dos actores profesionales, para 6 emociones y estilo neutro. Los resultados del test muestran que todas las emociones son correctamente identificadas muy por encima del nivel de azar, y para ambos locutores. Por lo tanto, podemos concluir que la base de datos representa un recurso lingüístico válido para los propósitos de investigación y desarrollo para los que fue originalmente diseñada.

1. INTRODUCCIÓN

Debido al progreso en las técnicas de síntesis de voz durante los últimos años, la inteligibilidad de la mayoría de los CTV (conversor de texto a voz) es prácticamente equivalente a la del habla humana. Sin embargo, la naturalidad y fluidez de las voces sintéticas está lejos de ser indistinguible de la voz natural. Una expresión emocional apropiada representa uno de los aspectos claves de la naturalidad del que aún carecen los sistemas de síntesis del habla. El habla emocional puede ser tanto una forma de reducir la monotonía de la voz sintética, como una manera de mejorar la comunicación hombre-máquina.

A lo largo de los últimos años se han llevado a cabo una serie de intentos para desarrollar un CTV expresivo [1][2][3]. El resultado final no ha alcanzado aún un nivel suficiente como para que las emociones sean percibidas como naturales. Con el fin de transmitir emociones realistas, es necesario una profunda investigación de las características prosódicas del habla emocional: contorno de pitch, duración de los fonemas y curva de energía. Y para que dicho estudio sea posible, resulta imprescindible la grabación de una buena base de datos emocional.

Si bien es cierto que las modificaciones prosódicas tienen una indudable influencia a la hora de expresar emociones [4][5], el habla expresiva resultante con estos métodos no llega a alcanzar la naturalidad deseada [6]. Y esto sucede aún cuando se utiliza directamente prosodia extraída de señales reales [7]. Ello es debido a que las emociones se expresan no solo a través de variaciones de la prosodia, sino mediante cambios en las características espectrales de la voz [8]. Dado que modelar dichas propiedades acústicas de forma explícita es realmente complejo, se pueden emplear técnicas basadas en corpus para hacerlo implícitamente.

En las soluciones basadas en corpus, cada frase está generada a partir de la concatenación de unidades extraídas de una base de datos de voz natural. El algoritmo de selección de unidades está basado en la minimización de una función coste global formada por un coste objetivo y uno de concatenación [9]. Ambos costes se hallan delimitados entre los valores 0 (la mejor elección posible) y 1 (el peor escenario posible).

El coste objetivo mide el parecido entre la unidad deseada (predicha por el módulo lingüístico y prosódico del CTV) y las unidades candidatas disponibles en la base de datos. El coste de concatenación por su parte, es calculado como una medida de la distorsión resultante al unir dos unidades candidatas. Obviamente el coste es igual a cero si ambas unidades aparecen de forma consecutiva en la base de datos.

Los sistemas de selección de unidades proporcionan buenos resultados cuando se dispone de suficientes unidades candidatas similares a la secuencia de unidades objetivo dada. De manera que no sea necesario llevar a cabo modificaciones prosódicas que supondrían la introducción de distorsiones, reduciendo así la calidad de la voz sintética. Es por ello de vital importancia contar con una base de datos de gran tamaño, poblada con una cantidad suficiente de unidades

para cada una de las emociones que se pretenda sintetizar.

A lo largo de este artículo se presenta la evaluación subjetiva de una base de datos de habla con emociones. En la sección 2 se procederá a describir el proceso de diseño y grabación de la base de datos. El protocolo de evaluación se detalla en la sección 3 y los resultados son presentados y argumentados en la sección 4. Finalmente se plasman algunas conclusiones sobre el proceso.

2. DISEÑO Y GRABACIÓN DEL CORPUS

La base de datos que se describe a continuación, fue creada teniendo en mente dos propósitos. Por una parte, se buscaba que formara el núcleo sobre el que desarrollar un sistema CTV emocional y basado en corpus para el Euskera. Por otra parte, se pretendía que fuera un recurso útil para el análisis prosódico y espectral del habla emocional.

2.1. Grabación del habla emocional

Existen diversas técnicas para grabar voz emocional, cada una con sus ventajas e inconvenientes, y entre las que podemos destacar las siguientes:

- *Emociones espontaneas*: Sin lugar a dudas las emociones más auténticas pero plantean dificultades técnicas para su grabación. Por otra parte, es prácticamente imposible controlar el contenido de las grabaciones, por lo que resulta inviable recolectar una base de datos adecuada para síntesis por corpus, debido a las restricciones en la cobertura fonética que este tipo de aplicaciones imponen.
- *Emociones provocadas*: El locutor es puesto en una situación concreta con la intención de despertar en él un estado emocional específico. Sin embargo, dado que cada persona actúa de manera diferente incluso ante un mismo estímulo, la emoción grabada mediante este método no se puede garantizar totalmente. Otra desventaja es la relativa a las consideraciones éticas que saltan a la palestra cuando es necesario evocar situaciones negativas para la recolección de emociones como pueden ser el miedo o la tristeza.
- *Emociones actuadas*: Esta técnica consiste en la lectura de un texto por parte de un actor profesional, intentado emular la emoción deseada. Este tipo de grabación es comúnmente acusada de producir emociones exageradas y faltas de naturalidad.

Finalmente, el tercer método de grabación fue el elegido por las ventajas que proporciona. Por una parte, permite controlar el contenido de la base de datos preservando la variabilidad fonética con la que fue diseñado el corpus. Y por otro lado, facilita el estudio y comparación de las características de cada emoción ya que el contenido textual se mantiene.

Durante la grabación se utilizaron textos semánticamente neutros no relacionados con las emociones. Un único corpus textual fue utilizado para la grabación de todas las emociones. La validez de esta elección fue probada experimentalmente en [10]. En lo que al contenido expresivo se refiere, se consideraron las seis “Big emotions” (Tristeza, alegría, enfado, miedo, sorpresa y asco) [11] ya que representan las más universalmente reconocibles y vinculadas con gestos faciales. Además, se realizó la grabación de estilo neutro típicamente utilizado en los sistemas CTV genéricos.

Para la grabación de aproximadamente una hora de habla para cada emoción, se seleccionaron un total de 702 frases mediante técnicas de análisis textual, garantizando tanto el balanceado fonético como la cobertura de difonemas presentes en el Euskera. El corpus fue grabado por dos actores profesionales: un actor de doblaje y una locutora de radio. Para una descripción detallada de las características del corpus consúltese [12].

3. EVALUACIÓN

Para determinar la validez del contenido emocional de la base de datos, se puso en marcha una campaña de evaluación subjetiva.

.

3.1. Diseño del test

Para descubrir si los evaluadores eran capaces de identificar la emoción expresada en las grabaciones de la base de datos, se llevó a cabo un test de elección forzada. Utilizando una interfaz basada en web se presentaron 30 estímulos de cada uno de los actores. Las señales de test se agruparon aleatoriamente de diez en diez formando formularios. Los sujetos evaluadores debían seleccionar una de las 6 posibles emociones, ya que no existía la opción de respuesta “no identificada”. Todas las frases de test eran enunciativas salvo una interrogativa. La longitud media de las señales era de 8.61 palabras, oscilando entre 4 para la más corta y 14 para la más larga.

3.1. Protocolo de evaluación

Cada uno de los evaluadores realizó el test individualmente. Las señales eran escuchadas a través de auriculares y tarjetas de sonido de gran calidad. Los oyentes no tuvieron un periodo de entrenamiento ni recibieron constancia de si sus respuestas eran correctas o no. Debían identificar las 10 señales presentadas en cada formulario, una vez hecho lo cual no se les permitía volver atrás para modificar sus contestaciones. Sin embargo, sí que podían escuchar las señales tantas veces como les fuera necesario antes de decidir la respuesta final. También se permitían descansos entre formulario y

formulario, si bien la mayoría completó el test sin necesitar ninguno.

Un total de 20 sujetos participaron en la evaluación (14 hombres y 6 mujeres) con edades que oscilaban entre 20 y 53 años. Todos los evaluadores hablaban Euskera si bien es conveniente realizar ciertos apuntes. El Euskera es una lengua minoritaria cuyo número de hablantes ha aumentado en los últimos años gracias a la promoción del idioma llevada a cabo en el sistema educativo. Podemos dividir la comunidad vasco-parlante en dos grupos: Aquellos para los que el Euskera es su primera lengua (Euskaldun Zahar) y los que lo han aprendido como segunda lengua (Euskaldun Berri). De los que completaron el test, solo 11 eran hablantes nativos.

4. RESULTADOS

Los resultados del test subjetivo se presentan en la Tabla 1. Cada una de las filas de la matriz representa la emoción expresada por el actor, mientras que las columnas muestran las emociones identificadas por los oyentes. Los valores son porcentajes y las letras simbolizan las emociones de la siguiente manera: Enfado (E), Miedo (M), Sorpresa (S), Asco (A), Alegría (L) y Tristeza (T).

La tabla también incluye los estadísticos Precisión (P) y Recall o Recuperación (R). La Precisión se calcula como el número de identificaciones correctas entre el número de respuestas asignadas a esa emoción. La Recuperación por su parte, se mide como el número de identificaciones correctas entre el número de estímulos existentes para dicha emoción.

Actores	EVALUADORES							
	E	M	S	A	L	T	P	R
E	81.5	2.5	5.5	9	-	1.5	0.78	0.82
M	0.5	64	3	-	1	31.5	0.68	0.64
S	6	2.5	73	1	17.5	-	0.80	0.73
A	15.5	4	3.5	67	2.5	-	0.86	0.67
L	0.5	0.5	5	-	94	-	0.81	0.94
T	-	20	1	0.5	1	77.5	0.66	0.78

Tabla 1. *Matriz de confusión con los resultados totales*

Puede apreciarse de manera clara que todas las emociones fueron identificadas muy por encima del nivel de azar (17%) a pesar de que el corpus estaba formado por frases semánticamente neutras que podían, a priori, dificultar el proceso de identificación. El nivel medio de reconocimiento es de 76.6%, siendo Alegría (94%) la emoción más fácilmente reconocida y Miedo (64%) la que más dificultades mostraba. Tristeza es la emoción con la menor Precisión (66%) ya que fue seleccionada como respuesta para estímulos relativos a Miedo, Asco o Enfado. La baja Recuperación pero alta Precisión para el Asco se debe al hecho de que raramente ha sido elegida como respuesta durante el test, pero cuando se ha hecho se ha acertado en la mayoría de los casos. Igualmente aunque a la inversa, Alegría ha sido una elección frecuente en el test y de ahí su alta Recuperación pero moderada Precisión.

Las Tablas 2 y 3 Ilustran las matrices de confusión para cada uno de los actores. Igual que para la matriz global, Alegría obtiene los mejores resultados en ambos casos con un 96% de identificaciones correctas para la actriz y 92% para el actor. Sin embargo, la emoción peor identificada difiere en esta ocasión, siendo Miedo (61%) para la locutora y Asco (59%) para él, aunque con una gran precisión en este último caso. La tasa de reconocimiento media es muy similar también: 75.83% para la actriz y ligeramente superior (76.50%) para el locutor masculino.

Actriz	EVALUADORES							
	E	M	S	A	L	T	P	R
E	75	-	6	15	-	3	0.76	0.75
M	1	61	4	-	1	34	0.73	0.61
S	10	2	68	-	20	-	0.82	0.68
A	13	-	2	75	1	9	0.83	0.75
L	1	-	3	-	96	-	0.81	0.96
T	-	19	-	-	1	80	0.63	0.80

Tabla 2. *Matriz de confusión para la actriz*

Actor	EVALUADORES							
	E	M	S	A	L	T	P	R
E	88	4	5	3	-	-	0.81	0.88
M	1	67	2	-	1	29	0.64	0.67
S	2	3	78	2	15	-	0.79	0.78
A	18	8	5	59	4	6	0.91	0.59
L	-	1	7	-	92	-	0.81	0.92
T	-	21	2	1	1	75	0.68	0.75

Tabla 3. *Matriz de confusión para el actor*

Volviendo a los resultados globales, las emociones que más comúnmente se confunden son Miedo y Tristeza. Miedo es identificado como Tristeza el 34% de las ocasiones, y Tristeza es identificada como Miedo el 20%. Ambas representan igualmente, el par de emociones que mayor número de veces se confunden tanto para las señales del actor como de la actriz por separado, en un rango que oscila entre el 19% y 34%. La confusión entre estas dos emociones ya se había observado por ejemplo, en la base de datos Interface para castellano [13].

Teniendo en cuenta que los evaluadores no dispusieron de una sesión de entrenamiento previo, se procedió a analizar si los resultados obtenidos en la segunda mitad del test (tasa de reconocimiento del 79.7%, rango de confianza: entre 76.26% y 83.07%) eran significativamente mejores que los de la primera (72.64%, intervalo: 69.26% - 76.06%). En esta ocasión el test t-student dictamina que la hipótesis acerca de la significancia estadística de las tasas de reconocimiento, sí resulta verdadera. Concretamente se obtiene una $t=2.85 \rightarrow p=0.0044 > 0.05$. Un análisis más detallado de los datos revela que dicha mejora del 7% en la tasa de identificación, se mantiene prácticamente constante en los distintos grupos: hombres, mujeres, actor, actriz, hablantes nativos, etc.

El resultado resulta comprensible porque durante los primeros estímulos y debido al test de respuesta forzada y a la ausencia de entrenamiento previo, es posible que los evaluadores elijan una respuesta sin estar completamente seguros. Según el test va avanzando, los oyentes aprenden la forma en la que el actor expresa determinadas emociones y ello facilita la identificación del resto por descarte.

5. CONCLUSIONES

Los resultados de la evaluación subjetiva han constatado que todas las emociones son fácilmente reconocibles para ambos locutores. Por lo tanto, esta base de datos de voz, representa un recurso lingüístico válido que permitirá tanto la caracterización del habla emocional para el Euskera, como la creación de un CTV con emociones que actualmente se encuentra en pleno desarrollo.

En lo que al diseño del test se refiere, la significativa mejora de las identificaciones durante la segunda mitad del test nos lleva a modificar la estrategia para futuras evaluaciones. Se mantendría así la estructura de elección forzada de una respuesta entre las 6 emociones posibles, añadiendo para cada señal una pregunta binaria para determinar si la identificación ha resultado sencilla o no. Esto permitiría analizar las respuestas finales con mayor eficiencia.

6. AGRADECIMIENTOS

Esta evaluación ha sido posible gracias a la financiación del Gobierno del País Vasco dentro del programa ANHITZ (ETORTEK96/114) y al MEC (TEC2006-13694-C03-02/TCM).

Los autores agradecen la colaboración de todos los sujetos que participaron en la evaluación.

7. BIBLIOGRAFÍA

[1] IIda, A., Campbell, N. Higuchi, F., & Yasumura, M. (2003). A Corpus based speech synthesis system with emotion, In *Speech Communication*, 40, pp. 161--187

[2] Murray, I.R. and Arnott, J.L. Synthesising emotions in speech: is it time to get excited?, In *ICSLP* 1996, pp. 1816--1819

[3] Bulut, Murtaza, Shrikanth S. Narayanan, & Ann K. Syrdal. Expressive speech synthesis using a concatenative synthesizer, In *ICSLP* 2002, pp. 1265--1268

[4] Vroomen, J., Collier, R., & Mozziconacci, S. J. L., Duration and Intonation in Emotional Speech, In *Eurospeech* 1993, Vol. 1, pp. 577--580

[5] Montero, J. M., Gutiérrez-Arriola, J., Colás, J., Enríquez, E., & Pardo, J. M., Analysis and Modeling of Emotional Speech in Spanish, In *ICPhS* 1999, pp. 957--960

[6] Schröder, M. Can emotions be synthesized without controlling voice quality?, In *Phonus 4, Research Report of the Institute of Phonetics* 1999, Saarland University, pp. 37—55

[7]Heuft, B., P ortele, T., & Rauth, M., Emotions in Time Domain Synthesis, In *ICSLP* 1996, pp.1974—1977

[8] Rank, E., & Pirker, H., Generating Emotional Speech with a Concatenative Synthesizer, In *ICSLP* 98, Vol. 3, pp. 671--674

[9] Hunt, A. and Black, A. Unit selection in a concatenative speech synthesis system using a large speech data base, In *ICASSP* 1996, pp. 373-376.Erlbaum Associates, pp. 252--262

[10] Navas, E., Hernáez, I., Luengo, I. An Objective and Subjective Study of the Role of Semantics and Prosodic Features in Building Corpora for Emotional TTS, in *IEEE Transactions on audio, speech and language processing* 2006, vol. 14, n. 4, pp. 1117--1127.

[11] Cowie, R., Cornelius, R.R. Describing the Emotional States that Are Expressed in Speech, In *Speech Communication* 2003, 40(1,2) pp. 5--32

[12] Saratxaga I, Navas E., Hernaez I., Luengo I. Designing and Recording an Emotional Speech Database for Corpus Based Synthesis in Basque, In *Proceedings of the LREC* 2006, pp. 2126--2129

[13] Nogueiras, A., Moreno, A., Bonafonte, A., Mariño, J.B., Speech Emotion Recognition Using Hidden Markov Models, In *Proceedings of Eurospeech* 2001, pp. 2679--2682

GRABACIÓN DE UNA BASE DE DATOS BILINGÜE EUSKERA/CASTELLANO PARA VERIFICACIÓN DE LOCUTOR

Iker Luengo, Eva Navas, Iñaki Sainz, Ibon Saratxaga, Jon Sánchez, Igor Odriozola, Juan José Igarza, Inmaculada Hernáez

AhoLab Signal Processing Group.
Departamento de Electrónica y Telecomunicaciones.
Universidad del País Vasco (UPV/EHU).
Alda. Urquijo s/n, 48013 Bilbao.

{ikerl, eva, inaki, ibon, ion, igor, jigarza, inma}@aholab.ehu.es

RESUMEN

Los grupos de investigación de procesado del habla que trabajan con lenguas minoritarias han de afrontar una serie de dificultades a la hora de grabar nuevas bases de datos orales para esas lenguas, tales como la falta de recursos previos, la escasez de personas que hablan el idioma de forma fluida y la dificultad de encontrar financiación para el proyecto. Algunas veces es posible aprovechar campañas de grabación para otros proyectos y extenderlos de tal forma que se incluyan grabaciones en esa lengua minoritaria para cada donante que lo domine. De esta forma se puede grabar una nueva base de datos con poco esfuerzo, ya que la campaña de grabación a sido preparada y financiada de antemano. Usando esta misma técnica se ha creado una nueva base de datos bilingüe euskera/castellano, gracias a la cual se está llevando a cabo un estudio sobre sistemas de verificación bilingües en estos idiomas. En el presente artículo se describe la base de datos resultante así como las dificultades encontradas durante su grabación.

1. INTRODUCCIÓN

Hoy en día muchos sistemas requieren de algún mecanismo de autenticación de usuario para evitar fraudes o accesos no autorizados. La mayoría de estos sistemas utilizan una autenticación basada en claves, pero estas claves se pueden olvidar o robar. Actualmente los métodos de autenticación biométrica son la mejor alternativa, ya que proporcionan una verificación extremadamente segura y precisa [1]. Además, las características biométricas no se pueden perder ni olvidar, y son muy difíciles de imitar. Este tipo de autenticación se utiliza en la actualidad en sistemas como ordenadores portátiles con control de acceso mediante huella digital o acceso a edificios mediante geometría de la mano. La voz es una característica biométrica no intrusiva, que tiene un alto grado de aceptabilidad y que es apropiada para sistemas de verificación a larga distancia sobre redes de datos y voz. Para el desarrollo de estos sistemas de autenticación basados en voz, es necesario contar con bases de datos orales con grabaciones de diferentes locutores.

Como método de autenticación biométrica, la verificación de locutor ha de decidir si una persona es o no quien dice ser, utilizando para ello una o más señales de voz de esta persona [2]. En un sistema de verificación de locutor general se pueden distinguir dos módulos: El módulo de entrenamiento (que genera un modelo para cada usuario del sistema) y el módulo de pruebas (que decide si una señal de voz ha sido producida por un locutor específico) [3][4]. Generalmente se supone que el idioma de las señales de entrenamiento y prueba es el mismo. Pero en entornos multilingües es deseable que los usuarios del sistema de verificación puedan utilizar cualquiera de los idiomas que conozcan para acceder al sistema, sin notar diferencias apreciables en el funcionamiento del mismo. Por ello, en los últimos años, varios grupos de investigación han centrado su atención en sistemas de reconocimiento de locutor en entornos multilingües, donde los modelos pueden ser entrenados utilizando un idioma y las pruebas ser realizadas en otro [5][6].

Este entorno multilingüe añade algunas dificultades al sistema de verificación. Por un lado, la diferencia entre los idiomas de entrenamiento y prueba provoca una reducción de la precisión del sistema [7]. Por otro, las diferencias entre los idiomas del modelo de locutor y el modelo de locutor universal en un sistema de verificación de locutor GMM provoca también un aumento de los errores [8].

El País Vasco es un ejemplo de este tipo de entornos multilingües, en el que conviven dos idiomas oficiales, el euskera y el castellano. El euskera es un idioma minoritario, y por tanto, existe una falta de recursos lingüísticos en este idioma [9]. Concretamente, no existe ninguna base de datos oral pública disponible para el desarrollo de sistemas de verificación en este idioma.

Este artículo presenta el trabajo y las dificultades de grabar una nueva base de datos oral bilingüe en el País Vasco para el desarrollo de sistemas de verificación de locutores bilingües euskera/castellano. La sección 2 analiza los problemas asociados a la grabación de nuevas bases de datos para lenguas minoritarias. La sección 3 describe la base de datos grabada y sus contenidos, mientras que en la sección 4 se describen las dificultades que surgieron durante esta adquisición. Finalmente se comentan algunas conclusiones en la sección 5.

2. ADQUISICIÓN DE NUEVAS BASES DE DATOS ORALES PARA LENGUAS MINORITARIAS

Aunque se consideran de alto interés social, las lenguas minoritarias no son económicamente interesantes. Puesto que hay pocos hablantes, no compensa invertir una gran cantidad de dinero para la investigación y desarrollo de nuevos recursos orales como bases de datos. Esto significa que generalmente es difícil encontrar fuentes de financiación para estos proyectos y que en los casos en los que se consigue, la financiación lograda es escasa. Además generalmente no hay muchos grupos de investigación trabajando en estos idiomas, por lo que tampoco es sencillo buscar colaboraciones ente grupos para repartir la carga de trabajo y los costes.

En el caso de bases de datos de verificación de locutor, el proceso se complica aun más debido a los requerimientos específicos de estas bases de datos. Por un lado, las grabaciones deben realizarse a lo largo de un período de tiempo suficientemente largo como para recoger la variabilidad natural de la voz [10]. Esto significa que cada locutor debe ser grabado más de una vez, en diferentes sesiones, lo que hace que el proceso de grabación sea más largo y caro. Por otro lado, es interesante que la distribución de sexo y edad de los locutores se aproxime a la verdadera distribución de los usuarios potenciales del sistema. Esta restricción, junto con el hecho de que en una lengua minoritaria hay pocos hablantes, hace que el reclutamiento de los donantes sea más complicado.

Para hacer factible la grabación de bases de datos en una lengua minoritaria puede ser interesante aprovechar las campañas de grabación organizadas para otros proyectos e incluir algunas grabaciones adicionales en esta lengua, aunque éste no sea el objetivo principal de la campaña. De esta forma se pueden obtener nuevas bases de datos con poco esfuerzo, dado que la campaña de grabación ya ha sido preparada de antemano.

En el laboratorio de procesado de señal Aholab de la Universidad del País Vasco se llevan a cabo diferentes investigaciones en tecnologías de la voz para el euskera, principalmente en los campos de conversión de texto a habla (CTH), reconocimiento automático del habla (RAH) y verificación de locutor. Para el desarrollo de estas investigaciones se necesitan bases de datos orales en euskera. Algunas veces, cuando se está llevando a cabo una campaña de grabación para otros proyectos (principalmente en castellano), se pide a los donantes que realicen algunas grabaciones extra en euskera, para completar una base de datos paralela en esta lengua.

3. DESCRIPCIÓN DE LA BASE DE DATOS

La nueva base de datos euskera/castellano se grabó junto con una base de datos biométrica multimodal adquirida en cinco universidades de España, incluyendo la Universidad del País Vasco [11]. En esta base de datos se adquirieron diferentes características biométricas, tales como huella dactilar, firma, escritura manuscrita, iris o habla (en castellano). Aprovechando la oportunidad también se grabó en euskera a aquellos donantes reclutados en la Universidad del País Vasco que eran hablantes fluidos en este idioma. De esta forma se consiguió construir una pequeña base de datos bilingüe para verificación de locutor con poco esfuerzo.

3.1. Diseño de la base de datos

El protocolo de adquisición incluyó cuatro sesiones distribuidas en el tiempo para capturar la variabilidad intra-locutor. Hay una diferencia de dos semanas entre la primera y la segunda sesión, cuatro ente la segunda y la tercera y seis semanas ente la tercera y la cuarta sesión.

En cada sesión se grabaron una serie de frases aisladas y unas secuencias numéricas. El conjunto de frases es el mismo para todos los locutores, aunque cambian de una sesión a otra. La primera sesión consta de cuatro frases para cada idioma, mientas que en las demás sesiones sólo se grabaron dos en cada idioma. Por tanto, el corpus contiene 10 frases en castellano y otras 10 en euskera. Se trata de frases fonéticamente ricas y equilibradas, seleccionadas mediante la herramienta CorpusCRT a partir de dos grandes corpus textuales, uno para cada idioma. Esta herramienta desarrollada por el grupo TALP de la UPC[1] proporciona un conjunto de frases reducido manteniendo, en la medida de lo posible, la frecuencia de aparición de los fonemas del corpus original.

Las secuencias numéricas están formadas por 8 dígitos que el locutor podía leer como prefiriera. Cada locutor tiene una secuencia numérica única que se repite cuatro veces en cada sesión. Además, cada locutor también grababa la secuencia numérica asignada a otros tres locutores, diferentes cada vez, con el objetivo de utilizarlas como pruebas de impostor. En total se graban 7 secuencias numéricas por cada sesión, locutor e idioma.

[1] Universidad Politécnica de Catalunya. www.talp.upc.es

3.2. Datos adicionales

Las grabaciones de la base de datos se procesaron para extraer la información de actividad vocal y las curvas de entonación.

La estimación de la actividad vocal es necesaria para descartar tramas en las que no hay información del habla, de forma que el nivel de ruido existente durante los silencios no corrompa los parámetros calculados para el sistema de verificación. La detección de actividad vocal utilizada se basa en la desviación espectral a largo plazo (LTSD) tal y como se propone en [12].

Para el cálculo de las curvas de entonación se utilizó una herramienta desarrollada por el mismo grupo Aholab [13], que utiliza programación dinámica y coeficientes cepstrales para estimar la curva de pitch.

4. DIFICULTADES ENCONTRADAS

4.1. Escasez de hablantes bilingües

Recoger una base de datos en euskera no es fácil, ya que no hay mucha gente que lo hable de forma fluida, ni siquiera en el propio País Vasco. La Tabla 1 presenta el número de hablantes de euskera en la Comunidad Autónoma del País Vasco en 2001, según edades[2]. En esta tabla se consideran hablantes bilingües tanto activos como pasivos, es decir, incluye a aquellos locutores cuyo primer idioma no es euskera, y por tanto, cuyo dominio del idioma no es siempre bueno (algunos no son hablantes fluidos).

Edad	Total	Porcentaje
16-24	170 453	23.1%
25-34	171 608	23.3%
35-49	175 522	23.8%
50-64	104 055	14.1%
>=65	115 442	15.7%
TOTAL	737 080	100.0%

Tabla 1: *Distribución de hablantes bilingües activos y pasivos por edades en la Comunidad Autónoma del País Vasco en 2001.*

El conocimiento y uso del euskera varía según el rango de edad. La Tabla 2 muestra el porcentaje de personas monolingües y bilingües para cada rango de edad. Como puede verse la proporción de hablantes de euskera es mayor entre las personas jóvenes.

Además de la falta de hablantes, el hecho de que la base de datos bilingüe se grabara como una extensión de otra base de datos biométrica perteneciente a otro proyecto también acarrea problemas. Las especificaciones principales de la base de datos biométrica, como por ejemplo el número de voluntarios, su distribución de edad y los plazos de entrega tenían que ser respetados. Puesto que las grabaciones en euskera no formaban parte de la base de datos principal, el conjunto de especificaciones para la base de datos biométrica no tenía en cuenta los requerimientos especiales necesarios para una base de datos bilingüe. Por ejemplo, no era posible rechazar a un voluntario por el hecho de no hablar euskera, ya que esto hubiera supuesto dificultar el reclutado y extender los plazos de entrega de la base de datos principal. Esta es la razón por la que, aunque 55 voluntarios fueron grabados en castellano, sólo 30 de ellos se grabaron en euskera, al ser los únicos realmente bilingües.

Edad	Monolingües	Bilingües
16-24	31.4%	68.6%
25-34	50.5%	49.5%
35-49	63.3%	36.7%
50-64	72.5%	27.5%
>=65	67.4%	32.6%

Tabla 2: *Porcentaje de hablantes monolingües y bilingües por edades en la Comunidad Autónoma del País Vasco en 2001.*

4.2. Desviación de la distribución de edad

En una base de datos de verificación de locutor la población debe estar correctamente representada. Es importante que la base de datos incluya ejemplos representativos de todos los potenciales usuarios del sistema. Esta es la razón por la que este tipo de bases de datos suelen estar equilibrados en sexo y rangos de edad. Para lograr este equilibrio se propone una distribución objetivo para los locutores, según la distribución real de los usuarios potenciales, y se seleccionan los donantes de acuerdo a este objetivo. La Tabla 3 muestra la distribución objetivo de rangos de edad y la distribución de los locutores grabados en castellano y euskera.

Edad	Objetivo	Castellano	Euskera
18 – 25	30%	32.7%	33.3%
25 – 35	20%	40.0%	53.3%
35 – 45	20%	12.7%	10.0%
45 – 55	20%	7.3%	3.3%
>= 55	10%	7.3%	0.0%

Tabla 3: *Distribución de edad en los locutores de la base de datos en castellano y euskera.*

El reclutamiento de los locutores se realizó principalmente entre los estudiantes y el personal de la Escuela de Ingeniería de la Universidad. La media de edad en este colectivo es relativamente baja, tal y como

[2] Fuente: EAS (Sistema Indicador de Lengua del País Vasco). http://www1.euskadi.net/euskara_adierazleak/zerrenda.apl?hizk=i&gaia=25&sel=64

se refleja en la desviación del objetivo en los rangos de 25 a 35 años y de más de 45, tanto para castellano como para euskera. Además, es muy difícil reclutar donantes bilingües mayores de 35 años, ya que la mayoría de las personas en este rango de edad no hablan euskera, tal y como se refleja en la Tabla 1. Esta es la razón por la que hay tan pocos locutores grabados en euskera en rangos de edad altos.

La desviación de la distribución de edades es mayor para las grabaciones en euskera que para el castellano. Otra vez, la razón principal es que durante el reclutamiento era prioritario mantener la distribución de edad para la base de datos principal, en la que se incluían las grabaciones en castellano. Pero al descartar a los no bilingües, la nueva distribución de edades para el euskera no coincidía con el objetivo.

El equilibrio de sexos fue más sencillo de conseguir. En la Tabla 4 se muestra la distribución objetivo junto con las obtenidas para el castellano y euskera. Como se aprecia, las distribuciones logradas no difieren significativamente entre ambos idiomas.

Sexo	Objetivo	Castellano	Euskera
Hombre	50%	47.3%	43.3%
Mujer	50%	52.7%	56.7%

Tabla 4: *Distribución del sexo de los locutores en la base de datos en castellano y euskera.*

5. CONCLUSIONES

Teniendo en cuenta que el euskera es una lengua minoritaria, el desarrollo de nuevos recursos orales para este idioma es difícil y la financiación escasa. Bajo estas circunstancias, la adquisición de una base de datos en una lengua mayoritaria representa una oportunidad que puede aprovecharse para construir otra base de datos en la lengua minoritaria. Haciendo uso de esta estrategia se ha creado una nueva base de datos para verificación de locutores bilingües en euskera y castellano. Sus características no son ideales, ya que el proceso de adquisición no fue diseñado explícitamente para ella, pero sigue siendo un recurso útil, con el que ya se han realizado algunos experimentos de verificación de locutor [14].

6. AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el Gobierno Vasco bajo la subvención IE06-185 (proyecto ANHITZ, http://www.anhitz.com) y por la Universidad del País Vasco y EJIE S.A. bajo la subvención EJIE07/02 (proyecto MULTILOK).

Los autores también quieren agradecer su participación a todos los voluntarios que tomaron parte en la adquisición de la base de datos biométrica.

7. BIBLIOGRAFÍA

[1] A.K. Jain, A. Ross, S. Pankanti, *Biometrics: a tool for information security*, IEEE Transactions on Information Forensics and Security, vol. 1, pp. 125—143, 2006.

[2] J.P. Campbell, S*peaker Recognition: A tutorial,* In Proceedings of the IEEE, vol. 85, pp. 1437—1462, 1997.

[3] J.M. Naik, *Speaker verification: a tutorial*. IEEE Communications Magazine, vol. 28, pp. 42—48, 1990.

[4] F. Bimbot, J.F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, D.A. Reynolds, *A Tutorial on Text-Independent Speaker Verification* EURASIP Journal on Applied Signal Processing, vol. 4, pp. 430—451, 2004.

[5] T. Nordstrom, H. Melin, J. Lindberg, *Comparative Study of Speaker Verification Systems using the Polycost Database,* In Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP), pp. 1359—1362.

[6] M. Faundez-Zanuy, A. Satue-Villar, *Speaker Recognition Experiments on a Bilingual Database*, In Proceedings of the 14th European Conference on Signal Processing (EUSIPCO), 2006.

[7] B. Ma, H. Meng, *English-Chinese bilingual text-independent speaker verification* In Proceeding of the IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP), vol. 5, pp. 293—296, 2004.

[8] R. Auckenthaler, M.J. Carey, J.S.D. Mason, *Language dependency in text-independent speaker verification* In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) vol. 1, pp. 441—444, 2001.

[9] A. Díaz de Ilarraza, K. Sarasola, A. Gurrutxaga, I. Hernaez, N. Lopez de Gereñu, *HIZKING21: Integrating language engineering resources and tools into systems with linguistic capabilities,* In Proceedings of the Workshop on NLP of Minority Languages and Small Languages, 2003.

[10] P. Kenny, P. Dumouchel, *Disentangling speaker and channel effects in speaker verification*, In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 1, pp. 37—40, 2004.

[11] J. Galbally, J. Fierrez, J. Ortega-Garcia et. al., *BiosecurID: a Multimodal Biometric Database*, In Proceedings of the User-Centric Technologies and Applications Workshop, pp. 68—76, 2007.

[12] J. Ramirez, J. Segura, C. Benitez, A. de la Torre, A. Rubio, *Efficient Voice Activity Detection Algorithms Using Long Term Speech Information*, Speech Communication, vol. 42, pp. 271—287, 2004.

[13] I. Luengo, I. Saratxaga, E. Navas, I. Hernáez, J. Sanchez, I. Sainz, *Evaluation Of Pitch Detection Algorithms Under Real Conditions*. In Proceeding of 32nd IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 1057—1060, 2007.

[14] Luengo, I., Navas, E., Sainz, I., Saratxaga, I., Sanchez, J., Odriozola, I., Hernaez, I. Text independent speaker identification in multilingual environments. Proc. of the Sixth International Language Resources and Evaluation (LREC'08), paper 461, 2008.

INTELLIGIBILITY OF ACCENTED SPEECH: THE PERCEPTION OF WORD-FINAL NASALS BY DUTCH AND BRAZILIANS

Denise C. Kluge[1], Mara S. Reis[1], Denize Nobre-Oliveira[2] and Andréia S. Rauber[3]

[1]Federal University of Santa Catarina, Brazil,
[2]Federal Center of Technological Education of Santa Catarina, Brazil, [3]University of Minho, Portugal

ABSTRACT

In both English and Dutch, the nasal consonants /m/ and /n/ in word-final position have different phonological representations and are phonetically distinctive. In contrast, in Brazilian Portuguese /m/ and /n/ undergo similar phonological processes which result in the deletion of the nasals and regressive vowel nasalization. The present small-scale study aims at investigating whether speakers of English as a foreign language with two dissimilar phonological representations and phonetic realizations of nasals in word-final position differ when recognizing English words produced either accurately or in an accented way. The data collection took place at Universidade Federal de Santa Catarina and University of Amsterdam, with 10 speakers of each language. The results indicate that Dutch speakers tend to recognize the nasal productions more consistently than the Brazilians, a fact that is interpreted as due to the similar phonological and phonetic patterns of the target sounds that Dutch and English share.

1. INTRODUCTION

The nasalization of a vowel that precedes a nasal consonant is considered a widespread coarticulatory process present in the majority of the world's languages [1]. However, the degree of nasalization is different among languages, varying from subtle, as in English [2] and Dutch [3], to strong as in Portuguese [4].

Furthermore, languages may also have different patterns of phonological representations of the same phonemes in different word positions. The object of the present study is the investigation of the English bilabial and the alveolar nasal consonants /m/ and /n/ in monosyllabic word-final position. These consonants vary in the type of phonological representation between the two groups involved in the study, Brazilian Portuguese (BP) speakers, and Dutch speakers. For the Dutch, /m/ and /n/ are phonetically distinctive in word-final position, while in Portuguese they are phonetic realizations of the archiphoneme /N/.

The presence of nasalized vowels or consonants is spread over 99% of the languages [1], and this process of coarticulatory nasalization is extremely common. However, the nasals /m/ and /n/ in English word-final position are fully pronounced [5], with different places of articulation [6]. In fact, /m/ and /n/ in word-final position are phonetically distinctive in English, which leads to the existence of minimal-pairs such as *gym-gin*. What differs among languages is the degree of nasalization—while vowel nasalization is subtle in English ([2, 7, 8]) and in Dutch [3], BP is characterized by its typical vowel nasalization [4]. It is important to note that although vowel nasalization can occur in English, there are no nasal vowels in its inventory [2].

Therefore, due to the representation of the nasals in their native language (L1) in the context of a monosyllabic word, whereas Dutch speakers, as well as English speakers, maintain distinctive realizations between the nasals, Brazilians nasalize the preceding vowel and delete the nasal consonant. In other words, while Dutch and English have similar patterns of representation and realization of the nasals in word-final position, Portuguese differs in both aspects. Previous studies ([9, 10]) show that BP learners of English as a foreign language (EFL) tend to transfer the L1 pattern to both their second language (L2) perception and production.

As regards perception and production studies, it is commonly believed that adults are language-specific perceivers and that speech perception occurs through the filter of the L1 system, at least in initial stages of L2 learning ([11, 12, 13, 14, 15, 16]). Furthermore, current models of L2 phonological perception or of L2 phonological learning ([17, 18, 19]) have highlighted the role that accurate speech perception plays on accurate L2 speech production. A study conducted by Kluge et al. [10] found that, as proposed by Flege and colleagues, there is a tendency for a positive correlation between perception and production of English word-final nasals by Brazilian EFL learners, that is, the sounds which are better perceived are the ones which are better produced. Drawing on this perspective, it can be assumed that Brazilian and Dutch speakers/listeners would perceive the English target nasals according to their specific L1 norms. Table 1 summarizes the main differences among the languages involved in the study as far as the nasals /m/ and /n/ in word-final position are concerned.

Phonological system	realization	phonetic status	vowel nasalization
Brazilian Portuguese	deleted	not distinctive	yes – strong
Dutch	full	distinctive	maybe – subtle
English	full	distinctive	yes – subtle

Table 1. Realization of /m/ and /n/ in word-final position in BP, Dutch, and English.

Therefore, the English interlanguage of BP and Dutch speakers is expected to perform differently if they transfer their L1 phonological representations of /m/ and /n/ in word-final position into their L2: while Dutch would tend to perceive the English nasals in a more target language fashion, since the two systems have similar representations of the target nasals, Brazilians would not consistently distinguish the differences in the English production of nasals, as already shown by Kluge et al. [10].

The influence that a foreign accent exerts on speech intelligibility is a debatable aspect implicated in successful cross-language communication [20]. As Reis [21] points out, intelligibility, as far as English is concerned, is a current issue

in this period of "globalization and the importance of English as the contemporary lingua franca" (p. 138).

The intelligibility of foreign-accented speech has been evaluated through a variety of procedures. In fact, Bent, Bradlow and Smith [22] state that intelligibility depends on testing methods, and that results from different procedures could not be compared. Thus, the type of testing material (i.e., word, sentence, passages), the way of eliciting speech (e.g., reading tasks vs. extemporaneous speech), the listening condition (e.g., in quiet or with noise), and the tasks of the judges (e.g., subjective rating, transcription, comprehension questions, summary of the utterance) interfere in what may be analyzed as intelligible or not. Nonetheless, Weil [23] is assertive about foreign-accented speech studies: "accented speech is less intelligible than non-accented speech" (p. 7).

As regards the procedures of intelligibility tests, studies have applied a variety of them, such as mispronunciation detection [24], sentence verification [25], phonetic and word discrimination ([20, 26]), and transcription accuracy ([25, 27, 28]). Ingram and Nguyen [29] argue that the use of judgment based on rating scales is the most common type of intelligibility and accentedness assessment ([30, 31, 32, 33]). In this kind of test, the listeners are required to evaluate how difficult it is to understand an utterance, or how strong the accent is. The present study provides the listeners with a word intelligibility test with two types of tasks: (i) word recognition, and (ii) judgment on a rating scale of how English-like the pronunciation of a word sounds. Two types of realization of word-final nasals were presented in the test: accurately produced with full distinctive realization of each nasal consonant, and BP-accented speech produced with vowel nasalization/nasal consonant deletion. The tasks and the entire method used in the study will be described in the next section.

2. METHOD AND PROCEDURES

Two instruments were used for data collection: a questionnaire for assessing the participants' background, and a word recognition test. The data gathering took place at the Universidade Federal de Santa Catarina (UFSC), and at the University of Amsterdam (UvA).

2.1. Research question and hypotheses

In order to examine the intelligibility of English monosyllabic words with the nasal consonants in word-final position produced with the typical BP vowel nasalization/nasal deletion accent, the following research question (RQ) and hypotheses (H) are proposed:

RQ 1: How do groups with different L1 patterns of phonological representation of the nasal consonants in word-final position recognize L2 English words produced both accurately and in an accented way?

H1a: Dutch listeners will recognize accurately produced English words more often than will BP listeners;

H1b: Dutch listeners will recognize accented English words more often than will BP listeners.

2.2. Participants

Two groups of EFL speakers participated in the study: 10 Brazilian EFL learners (9 females and 1 male, ages ranging from 18 to 30 years) and 10 Dutch participants (all females, ages ranging from 18 to 26 years).

The questionnaire that assessed the BP participants' profile showed that they had been learning English for an average of 8 years. They used to speak the L2 in an average of 9% of their daily routines (at home/school, with family/friends, at work); however, they listened to the L2 in an average of 26% of the time (at school, on the internet, watching TV etc.).

The Dutch listeners had never been to a Portuguese-speaking country, thus we might assume that they were not used to the typical vowel nasalization that Portuguese speakers transfer when producing English nasals in word-final position. These participants had been studying English for an average of 9.4 years, used the L2 in about 8% of their daily routines, and listened to the L2 an average of 22% of the time.

Although the level of L2 proficiency was not assessed, and we assume a considerable difference between the quantity and quality of authentic input that these two groups receive, the present study upholds Flege's and other scholars' viewpoint that the amount of first language use is one of the determinant factors that interferes in L2 perception ([34, 35, 36, 37]). Therefore, given that the two groups (i) demonstrate similar length of L2 experience in formal settings, (ii) use their L1s more often than the L2, and (iii) use the L2 with similar frequency rates, they could be considered functional monolinguals or naïve non-native listeners [16]. According to Best and Tyler [16], functional monolinguals are those who learn the L2 in formal learning settings, and do not use the L2 in an everyday basis.

2.3. Materials

2.3.1. Stimuli

Six monosyllabic minimal-pair words were used in the perception test, all of them ending with the nasals /m/ or /n/ in word-final position: *cam/can, Tim/tin,* and *gem/gen*. The words were recorded by two female speakers, one American and one Brazilian. Both speakers had phonetic training and were proficient in their L2, i.e., the American in BP, and the Brazilian in American English. The speakers were recorded individually in a silent room, with a Sony MZ-NHF800 Minidisk and a monodirectional Sony microphone (ECM-MS907). Each word was recorded in two different conditions: with and without vowel nasalization/nasal deletion. That is, the word *Tim*, for example, was recorded either as /tɪm/ or as /tĩ/ by the two talkers. It is important to note that the vowel quality was maintained in both productions, according to the American vowel inventory.

Thus, each of the six words had two different conditions (with or without vowel nasalization/nasal deletion) and was produced by the two speakers, so that the six words resulted in twelve realizations. In the test, each realization was repeated four times, two produced by each of the speakers. As a consequence, the entire test consisted of 48 productions (12 realizations x 4 repetitions = 48 samples). The stimuli were digitized and normalized for peak intensity with Sound Forge 7.0, and the 48 words were organized and randomized in Praat [38]. Three extra trials were inserted both in the beginning and in the end of the test, totaling 54 trials. However, these 6 extra trials were not analyzed.

2.3.2. Intelligibility test

The intelligibility test consisted of a word recognition task. The words were presented in isolation, and each of the 48 words was repeated twice in each trial. In the word recognition task, the participants heard the word and had 4 seconds to mark, within a three-alternative forced choice answer, the word they heard. For example, when the participants heard the production of *Tim* or *tin*, they had to choose between *Tim, tin* or *neither of the alternatives.*

The results of this task guided the data analysis as regards the interference of vowel nasalization and nasal consonant deletion on word intelligibility. It was expected that, due to the pattern of phonological representation of the nasals in word-final position of each language—with or without phonetic distinction—, the BP-accented English speech would be consistently perceived as accented by the Dutch, and inconsistently perceived as accented by the BP participants. The analysis of the responses in the word recognition task was considered correct only when the participants chose the appropriate corresponding label for the intended production. For example, if the word produced was /tɪm/, the corresponding label was *Tim*. If the word produced was /t̃ĩ/, which is either a mispronunciation of *Tim* or *tin*, the listeners were expected to choose the label *neither of the alternatives*.

2.3.3. Statistical analysis

The statistical analysis was based on the correct responses of the two groups for the 48 items in the test. Due to the limited number of participants, 10 in each group, the raw data were converted into percentages. Statistical significance (alpha level) was set at .05, and due to the non-consistency between the results of skewness and kurtosis, the entire data were considered not normally distributed. Thus, the following non-parametric tests were used (1) Mann-Whitney for between groups comparison of means; (2) Friedman for within group comparison of means, and (3) Wilcoxon as the post hoc test to verify the relation between the variables that had achieved significance in the Friedman test.

3. RESULTS AND DISCUSSION

The Dutch listeners were hypothesized to recognize more words, either produced accurately or in an accented way, than were BP listeners. Table 2 shows that accurate productions were recognized in an average of 77.9% by the Brazilians, and 97.5% by the Dutch, while the nasalized words were recognized 40.8% by the Brazilians, and 76.7% by the Dutch. A Mann-Whitney U Test confirms that not only had the Dutch significantly recognized more accurate words than the Brazilians (Z= -3.449, p= .001), they also outperformed the Brazilians in the recognition of the nasalized words (Z= -2.612, p= .009). Thus, the overall results demonstrate that Dutch listeners significantly recognized more words, either accurate or accented, than did the Brazilians (Z= -3.080, p= .002), a result which corroborates the hypotheses of the study.

	Accurate N: 240		Nasalized N: 240		Total N: 480	
	Score	mean	score	mean	score	Mean
BP	187	77.9 (14)	95	40.8 (27.7)	282	58.7 (17.9)
Dutch	234	97.5 (4.5)	184	76.7 (17.9)	418	87.1 (8.6)

Table 2. Recognition of accurate words.
N= Total number of occurrences. Score = total number of recognized words. Standard deviation in parentheses.

When analyzing word recognition by type of realization within the same group, a Friedman test confirms that there is a significant difference between the recognition of the accurate and accented words: for the Dutch (X^2 (1, N=10) = 7.000, p=.008), and for the Brazilians (X^2 (1, N=10) = 9.000, p=.003). That is, the statistical test confirmed that nasalized words disfavored word recognition by the two groups.

Therefore, since accurate pronunciation led to more word recognition by the two groups, the suggestion given by some authors ([39, 40]) that foreign-accented speech is more intelligible for L2 speakers is not corroborated by the results of the present study. Alternatively, the results of the word recognition task seem to support Ingram and Nguyen's [29] statement that accented-speech does not necessarily favor intelligibility by non-native listeners, as indicated by some studies ([41, 42, 43]).

4. CONCLUSION

The present small-scale study aimed at investigating whether EFL speakers with two different phonological representations of nasals in word-final position differed in a word intelligibility test of tokens produced accurately and with BP-accented word-final nasals. The hypothesis was that, due to different patterns of phonological representations, Dutch listeners would recognize either accurate or nasalized words more consistently than the Brazilians. The results showed that, in general, not only do the Brazilian participants recognize fewer words, they also vary more in word recognition than the Dutch participants.

It is important to bear in mind that the results of the word intelligibility test do not implicate that BP-accented production of word-final nasals *impede* overall speech intelligibility. However, whereas some studies have shown discrepancy between word comprehensibility and overall intelligibility ([25, 27, 35]), Weil [23] asserts that accented-speech surely is less intelligible than more native-like speech. Our findings lead to the conclusion that awareness of the difficulty in producing the target nasals may help L2 speakers to avoid vowel nasalization, thus enhancing intelligibility.

To conclude, it is important to state that this small-scale study had some limitations: (i) although other Portuguese varieties (e.g., European Portuguese) present vowel nasalization, due to availability of participants only BP speakers were tested; (ii) since in Dutch and English /m/ and /n/ are phonetically distinctive in word-final position, it would have been convenient to have another control group formed by native speakers of another Latin language; (iii) only front vowels preceded the target nasals, future research should investigate whether similar results would be obtained with back vowels; (iv) still regarding the stimuli, the place of articulation of the first consonant of the monosyllabic words should also be controlled.

5. REFERENCES

[1] N.F. Chen, J. Slifka, and K.N. Stevens. "Vowel nasalization in American English: Acoustic variability due to phonetic context". Proceedings of 16th ICPhS. Saarbrücken, Germany, 6-10 August 2007.
[2] H.J. Giegerich. *English phonology: An introduction*. Cambridge: Cambridge University Press, 1992.
[3] J. Stroop. "A Triggered Nasalization" (translation from the original article "Afgedwongen nasalering" in *Tijdschrift voor Nederlandse taal- en letterkunde*, deel 110 (1994), blz.55-67) in: http://www.janstroop.nl/artikelen/trigger.shtml, 1993.
[4] J.C. Oliveira, and T. Cristófaro-Silva. "Aprendizado de língua estrangeira: O caso da nasalização de vogais." Unpublished paper. UFMG, 2005.
[5] J.D. O'Connor. *Better English pronunciation*. Cambridge: Cambridge University Press, 1975.
[6] O. Fujimura, and D. Erickson. Acoustic Phonetics. In W.J. Hardcastle & J. Laver (Eds.), *The handbook of phonetic sciences*. Cambridge: Blackwell Publishers, 1997.

[7] M. Hammond. *The phonology of English: A prosodic optimality theoretic approach*. Oxford: Oxford University Press, 1999.
[8] P. Ladefoged. *A course in Phonetics*, *(4th Ed.).* Boston: Heinle & Heinle, 2001.
[9] D.C. Kluge. "The perception and production of English syllable-final nasals by Brazilian learners." Unpublished Master's thesis. Florianópolis: Universidade Federal de Santa Catarina, 2004
[10] D.C. Kluge, A.S. Rauber, M.S. Reis, and R.A.H. Bion. "The relationship between perception and production of English nasal codas by Brazilian learners of English." Proceedings of Interspeech 2007, pp. 2297-2300, 2007.
[11] D. Ellis. "Theory and explanation in information retrieval research." Journal of Information Science, vol. 8, pp. 25-38, 1984.
[12] A.M. Schmidt. "Cross-language identification of consonants. Part 1. Korean perception of English." JASA, vol. 99, no. 5, pp. 3301-3211, 1996.
[13] J.D. Harnsberger. "On the relationship between identification and discrimination of non-native nasal consonants." JASA, vol. 110, pp. 489-503, 2001.
[14] C.T. Best, G.W. McRoberts, and E. Goodell. "Discrimination of non-native consonant contrasts varying in perceptual assimilation to the listener's native phonological system." JASA, vol. 109, no. 2, pp. 775-794, 2001.
[15] R.P. Wayland. "The relationship between identification and discrimination in cross-language perception: The case of Korean and Thai." In O.-S. Bohn & M.J. Munro (Eds.), *Second language speech learning: The role of language experience in speech perception and production* (pp. 201–218). Amsterdam: John Benjamins, 2007.
[16] C.T. Best, and M.D. Tyler. "Nonnative and second-language speech perception: Commonalities and complementarities." In O.-S. Bohn & M.J. Munro (Eds.), *Second language speech learning: The role of language experience in speech perception and production* (pp. 13–34). Amsterdam: John Benjamins, 2007.
[17] J.E. Flege. "Second language speech learning: Theory, findings, and problems." In W. Strange (Ed.), *Speech perception and linguistic experience*: *Issues in cross-language research* (pp. 233-272). Timonium, MD: York Press, 1995.
[18] P.K. Kuhl, and P. Iverson. "Linguistic experience and the perceptual magnet effect." In W. Strange (Ed.), *Speech perception and linguistic experience*: *Issues in cross-language research* (pp.121-154). Timonium, MD: York Press, 1995.
[19] C.T. Best. "A direct realistic view of cross-language speech perception." In W. Strange (Ed.), *Speech perception and linguistic experience: Issues in cross-language research* (pp. 171-206). Timonium: York Press, 1995.
[20] S. Wijngaarden, H. Steeneken, and T. Houtgast. "Methods and models for quantitative assessment of speech intelligibility in cross-language communication." Proceedings of the Workshop on Multilingual Speech and Language Processing, Aalborg, Denmark, 2001. (Available as NATO Publication RTO-MP-066, April 2003.)
[21] M.S. Reis. "Guia de Pronúncia do Inglês para Brasileiros: Book review and experimentation." In J.M.F. de Luna (Ed.), *Educação e Linguística: Ensino de línguas*, pp. 137-149, 2001.
[22] T. Bent, A.R. Bradlow, and B.L. Smith. "Phonemic errors in different word positions and their effects on intelligibility of non-native speech." In O.-S. Bohn & M.J. Munro (Eds.), *Second language speech learning: The role of language experience in speech perception and production* (pp. 331-347). Amsterdam: John Benjamins, 2007.
[23] S.A. Weil. "The impact of perceptual dissimilarity on the perception of foreign accented speech." Unpublished PhD dissertation. Ohio State University, Psychology, 2003.
[24] P.M. Schmid, and G.H. Yeni-Komshian. "The effects of speaker accent and target predictability on perception of mispronunciations." Journal of Speech, Language, and Hearing Research, vol. 42, pp. 56-64, 1999.
[25] M.J. Munro, and T.M. Derwing. "Foreign accent, comprehensibility and intelligibility in the speech of second language learners." Language Learning, vol. 45, pp. 73–97, 1995.
[26] C.L. Rogers, J.M. Dalby, and K. Nishi. "Effects of noise and proficiency level on intelligibility of Chinese-accented English." Poster presented at the 141st meeting of the Acoustical Society of America, Chicago IL, 2001.
[27] M.J. Munro, and T.M. Derwing. "Processing time, accent and comprehensibility in the perception of native and foreign accented speech." Language and Speech, vol. 38, pp. 289–306, 1995.
[28] S.A. Weil. "Comparing intelligibility of several non-native accent classes in noise." Unpublished poster. Ohio State University, Psychology, 2002.
[29] J. Ingram, and T. Nguyen. "Vietnamese accented English: Foreign accent and intelligibility judgment by listeners of different language backgrounds." Submitted to TESOL in the internationalization of higher education in Vietnam Conference. Hanoi, Vietnam. May 12, 2007.
[30] E. Brennan, and J. Brennan. "Measurements of accent and attitude toward Mexican-American speech." Journal of Psycholinguistic Research, vol. 10, pp. 487–501, 1981.
[31] A. Burda, J. Scherz, C. Hageman, and H. Edwards. "Age and understanding of speakers with Spanish or Taiwanese accents." Perceptual and Motor Skills, vol. 97, pp. 11–20, 2003.
[32] T.M. Derwing, and M.J. Munro. "Accent, intelligibility, and comprehensibility: Evidence from four L1s." Studies in Second Language Acquisition, vol. 19, pp. 1-16, 1997.
[33] I. Thompson. "Foreign accents revisited: The English pronunciation of Russian immigrants." Language Learning, vol. 41, pp. 177–204, 1991.
[34] J.E. Flege, E. Frieda, and T. Nozawa. "Amount of native-language (L1) use affects pronunciation of an L2." Journal of Phonetics, vol. 25, pp. 169-186, 1997.
[35] J.E. Flege, M.J. Munro, and I.R.A. Mackay. "Factors affecting strength of perceived foreign accent in a second language." JASA, vol. 97, pp. 3125-3134, 1995.
[36] T. Piske, and I.R.A. Mackay. "Age and L1 use effects on degree of foreign accent in English." Proceedings of the 14th International Congress of Phonetic Sciences, pp. 1433-1436, 1999.
[37] D. Meador, J.E. Flege, and I.R.A. Mackay. "Factors affecting the recognition of words in a second language." Bilingualism: Language and Cognition, vol. 3, no. 1, pp. 55-67, 2000.
[38] Boersma, P., and Weenink, D. (1992–2008). Praat: doing phonetics by computer (Version 5.0.03) [Computer program]. Retrieved January 11, 2008, from http://www.praat.org/
[39] L. Smith, and J. Bisazza. "The comprehensibility of three varieties of English for college students in seven countries." Language Learning, vol. 32, pp. 259–269, 1982.
[40] S. Gass, and E. Varonis. "The effect of familiarity on the comprehensibility of nonnative speech". Language Learning, vol. 34, pp. 65–89, 1984.
[41] J.E. Flege. "Factors affecting degree of perceived foreign accent in English sentences." JASA, vol. 84, pp. 70–79, 1988.
[42] R. Major, S. Fitzmaurice, F. Bunta, and C. Balasubramanian. "The effects of nonnative accents on listening comprehension: Implications for ESL assessment". TESOL Quarterly, vol. 36, pp. 173–190, 2002.
[43] M.J. Munro, T.M. Derwing, and S.L. Morton. "The mutual intelligibility of foreign accents." Studies in Second Language Acquisition, vol. 28, pp. 111-131, 2006.

NUEVA TÉCNICA DE POST-CORRECCIÓN DE ERRORES DE RAH PARA SISTEMAS DE DIÁLOGO ORAL

Ramón López-Cózar, Zoraida Callejas

Dpto. de Lenguajes y Sistemas Informáticos, E.T.S.I. Informática y de Telecomunicación, Universidad de Granada, 18071 Granada, {rlopezc, zoraida}@ugr.es

RESUMEN

Este artículo propone una técnica para corregir errores de RAH en sistemas de diálogo oral que presenta dos novedades. Por una parte, el uso de varios contextos en los que se puede corregir un error de RAH, y por otra, el uso de valores de confianza asignados a las palabras empleadas para corregir palabras erróneas. Los resultados experimentales obtenidos usando el sistema de diálogo Saplen muestran que la técnica permite mejorar las tasas de exactitud de palabras, comprensión de frases, recuperación implícita y logro de tareas en 8,5%, 16,54%, 4% y 43,81% absoluto, respectivamente.

1. INTRODUCCIÓN

La mayor parte de las técnicas de post-corrección de errores de RAH existentes en la literatura usan información estadística acerca de palabras pronunciadas y palabras reconocidas [1], [2]. No obstante, estas técnicas requieren grandes cantidades de datos de entrenamiento. Además, el éxito de las mismas depende de la calidad de los resultados de RAH, y del tamaño de la base de datos de errores usada en el aprendizaje. Para soslayar estos problemas, diversos autores proponen usar información léxica, sintáctica, semántica o relacionada con la historia del diálogo [3].

La técnica que proponemos, aplicable a sistemas de diálogo oral, sigue esta última aproximación. Además de considerar diversas fuentes de información, tiene en cuenta diversos contextos en los cuales se pueden corregir errores de RAH. Asimismo, propone un método simple para asignar un valor de confianza a cada palabra empleada para corregir otra errónea. La técnica toma cada frase reconocida, y realiza correcciones en dicha frase. Se asume que el reconocedor asigna un valor de confianza a cada palabra reconocida, p.e. "quiero (0,7590) un (0,9268) bocadillo (0,8182) de (0,6532) lomo (0,4598)". No obstante, la técnica es igualmente aplicable si el reconocedor no proporciona dichos valores, descartándose en tal caso el algoritmo para el cálculo de los mismos.

Este trabajo ha sido subvencionado por el proyecto HADA TIN2007-64718 (Ministerio de Educación y Ciencia).

2. LA TÉCNICA PROPUESTA

2.1 Elementos necesarios

2.1.1 Clases de palabras

La técnica usa clases de palabras K_i que han de ser creadas a partir de las transcripciones de un corpus de frases de entrenamiento. Cada clase contiene palabras de un determinado tipo que son significativas para obtener el contenido semántico de las frases. Por ejemplo, clases de palabras relacionadas con el "pedido de comida rápida" son las siguientes: DESEO = {quiero, dame, ponme,…}, CANTIDAD = {un, una, uno, dos,…}, COMIDA = {bocadillo, tarda, ensalada,…}, BEBIDA = {agua, cerveza, refresco,…}. Llamamos Ω al conjunto de clases de palabras usadas para implementar la técnica propuesta en un determinado dominio de aplicación: $\Omega = \{K_1, K_2, \ldots, K_r\}$.

2.1.2 Reglas gramaticales

La técnica emplea un conjunto de reglas gramaticales simples que se usan para corregir errores de RAH que afectan a la semántica de las frases reconocidas. El formato general de una regla gramatical r_i es el siguiente: r_i: $pss_i \rightarrow restricción_i$, donde pss_i es un patrón sintáctico-semántico (descrito en la Sección 2.1.3) y $restricción_i$ es una condición que debe ser satisfecha por todas las clases de palabras en pss_i. Por ejemplo, una regla gramatical usada en nuestros experimentos es la siguiente:

r_1 : pss_1 → número(CANTIDAD) = número(BEBIDA)
and número(BEBIDA) = número(TAMAÑO)
and número(CANTIDAD) = número(TAMAÑO)

donde número es una función que devuelve 'singular' o 'plural' para cada palabra en la clase de palabras que recibe como entrada, y pss_1 es el patrón: CANTIDAD BEBIDA TAMAÑO.

2.1.3 Modelos sintáctico-semánticos

Un modelo sintáctico-semántico es una representación de la estructura conceptual de un tipo de frases

pronunciadas por locutores en un determinado dominio de aplicación. Por ejemplo, en nuestros experimentos hemos creado modelos sintáctico-semánticos para pedidos de comida y/o bebida, números de teléfono, códigos postales, direcciones, confirmaciones, etc. Para crear un modelo, se toma la trascripción de cada frase de un determinado tipo y se transforma en lo que llamamos un *patrón sintáctico-semántico* (*pss*), que es una secuencia de las clases de palabras en la trascripción. Por ejemplo, el *pss* correspondiente a la trascripción: *"por favor, quiero un bocadillo de jamón y una ensalada verde"* es el siguiente:

pss = DESEO CANTIDAD COMIDA INGREDIENTE CANTIDAD COMIDA INGREDIENTE

A partir del análisis de todas las transcripciones se crea un conjunto de *pss*'s. Dicho conjunto debe ser procesado para eliminar *pss*'s repetidos, y asociar a cada *pss* su frecuencia de aparición en el conjunto. Al resultado de este proceso lo denominamos *modelo sintáctico-semántico* asociado con el prompt T (MSS_T). Si el sistema de diálogo en que se pretende aplicar la técnica propuesta genera u tipos de prompts distintos, debemos crear u modelos sintáctico-semánticos. Llamamos α al conjunto formado por todos los modelos sintáctico-semánticos creados para el sistema de diálogo: $\alpha = \{MSS_{Ti}\}$, i = 1 ... u.

2.1.4 Modelos léxicos

Los modelos léxicos contienen información acerca del funcionamiento del reconocedor de habla del sistema de diálogo para cada estado del diálogo. Se asume que existe un estado del diálogo por cada tipo de prompt generado por el sistema. La técnica propuesta requiere que se cree un modelo léxico para cada tipo de prompt T (ML_T), cuyo formato es el siguiente: $ML_T = \{w_i, w_j, p_{ij}\}$, donde w_i es una palabra pronunciada por un locutor, w_j es el resultado de reconocimiento correspondiente a w_i, y p_{ij} es la probabilidad a posteriori de obtener w_j dada w_i. Para crear un ML_T se debe alinear cada trascripción de frase pronunciada con su correspondiente frase reconocida, y calcular la probabilidad p_{ij} para cada par de palabras (w_i, w_j). Si el sistema de diálogo en que pretendemos aplicar la técnica genera u tipos de prompts distintos, debemos crear u modelos léxicos. Llamamos β al conjunto formado por todos los modelos léxicos creados para el sistema: $\beta = \{ML_{Ti}\}$, i = 1 ... u. Para crear dichos modelos, hemos usado el algoritmo de alineación descrito en [4].

2.2 Algoritmos para implementar la técnica

2.2.1 Cálculo de valores de confianza y corrección a nivel estadístico

El objetivo de la corrección a nivel estadístico es encontrar palabras *w*'s en la frase reconocida que pertenezcan a conceptos incorrectos *K*'s. Si se encuentra una de estas palabras, la técnica debe determinar el concepto correcto *K*' y seleccionar la palabra más adecuada $w' \in K'$ para sustituir a la palabra w en la frase reconocida. Asimismo, la técnica debe calcular un valor de confianza para *w*', denotado C(*w*'), en caso de que la palabra w tuviera asociado un valor de confianza. Para calcular dicho valor, se tiene en cuenta el número de palabras existentes en el modelo léxico empleado para realizar la corrección, es decir, palabras u_i con las cuales se confunde la palabra w. Supongamos que dichas palabras forman el conjunto $U = \{u_1, u_2, \ldots, u_p\}$. Si U sólo contiene una palabra, u_1, entonces $w' = u_1$ y C(w') = 1,0. Si U contiene varias palabras, entonces w' es la palabra con mayor probabilidad de confusión con w. Si denotamos como p dicha probabilidad, entonces, C(w') = p. El algoritmo para realizar la corrección a nivel estadístico emplea los dos pasos siguientes:

Paso 1. ***Comparación de patrones***. Este paso opera sobre un patrón sintáctico-semántico *enriquecido* obtenido a partir de la frase reconocida, al que llamamos $psse_{INPUT}$. Dicho patrón es una secuencia de *contenedores* que almacenan información acerca de las palabras y valores de confianza en la frase reconocida, así como de las clases a las que pertenecen las palabras. La finalidad de este paso es transformar $psse_{INPUT}$ en otro patrón llamado $psse_{BEST}$. Para ello, se crea inicialmente un patrón sintáctico-semántico llamado pss_{INPUT}, que contiene únicamente las clases de palabras en $psse_{INPUT}$, por ejemplo:

pss_{INPUT} = DESEO CANTIDAD COMIDA INGREDIENTE

Seguidamente, se determina si pss_{INPUT} coincide con alguno de los patrones en MSS_T. En caso afirmativo, se asigna $psse_{BEST} = psse_{INPUT}$ y se prosigue con la corrección a nivel lingüístico (Sección 2.2.2). En caso contrario, se buscan patrones similares a pss_{INPUT} en MSS_T. Para ello, se compara pss_{INPUT} con cada patrón p en MSS_T, calculando un valor de similitud entre ambos patrones como sigue: similitud(pss_{INPUT}, p) = (n – m_{ed}) / n, donde n es el número de clases de palabras en pss_{INPUT}, y m_{ed} es la distancia mínima de edición entre ambos patrones [5]. La técnica selecciona como similares aquellos patrones con un valor de similitud mayor que un umbral $t \in [0{,}0–1{,}0]$, cuyo valor óptimo debe ser calculado empíricamente. Llamamos $pss_{SIMILAR}$ a cualquier patrón p en MSS_T tal que similitud($pss_{SIMILAR}$, p) > t. Consideramos 3 casos:

Caso 1. Sólo hay un $pss_{SIMILAR}$ en MSS_T. En este caso, el algoritmo crea un nuevo patrón llamado pss_{BEST}, realiza la asignación $pss_{BEST} = pss_{SIMILAR}$, y prosigue con el Paso 2 (Alineamiento de patrones).

Caso 2. No hay ningún $pss_{SIMILAR}$ en MSS_T. En este caso, se intenta encontrar algún $pss_{SIMILAR}$ en el conjunto α (descrito en la Sección 2.1.3). Si no se encuentra ninguno, no se realiza corrección a nivel estadístico. Si sólo se

encuentra uno, el procesamiento es como en el Caso 1. Si se encuentra más de uno, el procesamiento es como en el Caso 3.

Caso 3. Existen varios pss_{SIMILAR}'s en MSS_T (o en α). La cuestión entonces es determinar el mejor pss_{SIMILAR}. Para ello se selecciona el pss_{SIMILAR} que tenga mayor similitud con pss_{INPUT}. Si sólo existe un pss_{SIMILAR}, se asigna $pss_{\text{BEST}} = pss_{\text{SIMILAR}}$ y se prosigue en el Paso 2. Si existen varios pss_{SIMILAR}'s, se seleccionan aquéllos que tengan mayor frecuencia de aparición en MSS_T (o en α). Si sólo existe uno, se asigna $pss_{\text{BEST}} = pss_{\text{SIMILAR}}$ y se prosigue en el Paso 2. En caso de existir más de uno, no se realiza corrección a nivel estadístico.

Paso 2. ***Alineamiento de patrones***. Llegados a este punto, se ha obtenido pss_{BEST} a partir de pss_{INPUT}, pero no se ha creado $psse_{\text{BEST}}$. El objetivo de este paso es crear este patrón. Para ello, se alinea pss_{INPUT} con $psse_{\text{BEST}}$, y considerando cada contenedor C_i en pss_{INPUT}, se analizan 3 casos:

Caso A. La palabra w_i en C_i no afecta al contenido semántico de la frase. En este caso, se crea un nuevo contenedor D_i, se asigna $D_i = C_i$ y se añade D_i a $psse_{\text{BEST}}$.

Caso B. La palabra w_i en C_i afecta al contenido semántico de la frase. En este caso, se estudia si dicha palabra debe ser corregida, teniendo en cuenta los *pss*'s observados en el entrenamiento. Para ello, se intenta alinear el concepto C_i con algún concepto C_j en pss_{BEST}, y se consideran 3 casos:

Caso B.1. $C_i \neq C_j$. Este caso representa la situación en que se encuentra un concepto erróneo en la frase reconocida. Por tanto, se debe encontrar una palabra $w' \in C_j$, determinar su valor de confianza, almacenar dicha información en un nuevo contendor D_i, y añadir este contendor a $psse_{\text{BEST}}$. Para encontrar w' se usa ML_T y se crea el conjunto U de palabras $u \in C_j$ con las cuales se confunde la palabra w. Si sólo existe una palabra u_1 en U, se crea un nuevo contendor D_i cuyo nombre es el del contenedor C_j, y que contiene a la palabra u_1 junto con su valor de confianza, $C(u_1) = 1.0$. Finalmente, se añade D_i a $psse_{\text{BEST}}$. Si U está vacío, se actúa de forma análoga, pero considerando el conjunto β en lugar de ML_T. Si existe más de una palabra en U, se actúa de forma análoga, seleccionando la palabras que tenga mayor probabilidad de confusión con la palabra w.

Caso B.2. $C_i = C_j$. En este caso, se asume que el concepto de la frase reconocida es correcto, y que por tanto, no se debe realizar ninguna corrección. Por consiguiente, se hace $D_i = C_i$ y se añade D_i a $psse_{\text{BEST}}$.

Caso B.3. No es posible alinear C_i. Esto ocurre cuando el concepto C_i proviene de una palabra insertada en la frase reconocida a causa de un error de RAH. El algoritmo descarta C_i, es decir, no lo añade a $psse_{\text{BEST}}$.

2.2.2 Corrección a nivel lingüístico

La finalidad de la corrección a nivel lingüístico es corregir errores no detectados a nivel estadístico que afectan al contenido semántico de las frases. Por ejemplo, en nuestros experimentos, la frase "*una cerveza grande*" a veces es reconocida como "*dos cerveza grande*". Este tipo de error no puede ser reconocido a nivel estadístico, pues la secuencia de conceptos en la frase es correcta. Para realizar la corrección, se usa el conjunto de reglas gramaticales descrito en la Sección 2.1.2. Para cada regla se realiza el siguiente procesamiento. El patrón sintáctico-semántico de la misma se introduce en una *ventana* que se *desliza* de izquierda a derecha sobre $psse_{\text{BEST}}$. Si la secuencia de conceptos en la ventana se encuentra en $psse_{\text{BEST}}$, entonces se aplica la restricción $restricción_i$ a las palabras existentes en los contadores de $psse_{\text{BEST}}$. Si las palabras cumplen la restricción, no se realiza ninguna corrección. En caso contrario, se intenta determinar la causa de la incongruencia, buscando una palabra incorrecta. Este es el caso del ejemplo, pues número(CANTIDAD) ≠ número(BEBIDA). Para determinar la palabra w' con que se debe sustituir la palabra considerada errónea w, se examina el modelo léxico LM_T y se define el conjunto $U = \{u_1, u_2, ..., u_p\}$, constituido por palabras pertenecientes a la misma clase de palabras que w. Seguidamente, se actúa de forma análoga a como se ha explicado en el Caso B.1, con la salvedad de que ahora el objetivo no es reemplazar un concepto por otro, sino una palabra de un concepto por otra palabra del mismo concepto.

3. EXPERIMENTOS

El objetivo de los experimentos ha sido comprobar la efectividad de la técnica propuesta usando el sistema de diálogo Saplen [6], [7]. Las medidas de evaluación han sido las siguientes: exactitud de palabras (WA), comprensión de frases (SU), recuperación implícita (IR) y logro de tareas [8]. Para realizar comparaciones, los resultados experimentales han sido obtenidos usando dos sistemas de RAH distintos:

i) Sistema de RAH *base*, formado únicamente por el reconocedor de habla basado en HTK usado inicialmente por el sistema Saplen.
ii) Sistema de RAH *mejorado*, compuesto por el mismo reconocedor basado en HTK, seguido de un módulo adicional que implementa la técnica propuesta.

Se ha usado un corpus de frases construido a partir de diálogos reales entre estudiantes de nuestra Universidad y el sistema Saplen. Dicho corpus consta de unas 5.500 frases y unas 2.000 palabras distintas. El corpus ha sido dividido en dos corpus disjuntos, uno para entrenamiento (2.750 frases) y el otro para evaluación (2.750 frases). Usando el corpus de entrenamiento, se ha compilado una bigramática de palabras que permite reconocer frases de los 18 tipos existentes en el corpus.

Los experimentos han sido realizados empleando un simulador de usuarios desarrollado en un trabajo previo [6]. La interacción entre el sistema Saplen y el simulador se lleva a cabo empleando escenarios que representan objetivos que el simulador debe intentar conseguir durante la interacción. Hemos creados dos corpora de escenarios: *EscenariosA* (300 escenarios) y *EscenariosB* (100 escenarios). Cada diálogo generado durante la interacción simulador-Saplen se ha almacenado en un fichero log que se ha analizado para obtener los resultados experimentales.

Dado que la creación de los modelos sintáctico-semánticos y léxicos descritos en las Secciones 2.1.3 y 2.1.4 se ha realizado empleando los diálogos obtenidos mediante el simulador de usuarios, hemos realizado estudios preliminares para determinar la cantidad de diálogos que permite obtener la mayor cantidad posible de información sintáctico-semántica y léxica. Los resultados obtenidos muestran que a partir de 900 diálogos no aumenta la cantidad de información aprendida.

3.1 Experimentos con el sistema de RAH base

Usando el simulador de usuarios y empleando *EscenariosA* hemos generado otro corpus que contiene 900 diálogos. La Tabla 1 muestra los resultados medios (en %) obtenidos a partir de la evaluación de dicho corpus.

WA	SU	IR	TC
76,12	54,71	9,19	24,51

Tabla 1. Resultados de evaluación usando el sistema de RAH *base*

3.2 Experimentos con el sistema de RAH mejorado

De acuerdo con lo expuesto en la Sección 2.1.1, se ha creado un conjunto de clases de palabras $\Omega = \{K_1, K_2,\ldots, K_{21}\}$, que contiene las mismas clases usadas en un estudio previo [7]. Teniendo en cuenta lo expuesto en la Sección 2.1.2, se ha creado un conjunto de reglas gramaticales para determinar la concordancia en cuanto a número en las frases de tipo "pedido de productos." Para crear los modelos sintáctico-semánticos y léxicos, comentados en las Secciones 2.1.3 y 2.1.4, hemos usados el corpus de 900 diálogos inicial, obteniendo los conjuntos $\alpha = \{MSSL_{Ti}\}$ y $\beta = \{ML_{Ti}\}$, i = 1 ... 43, pues el sistema Saplen genera 43 tipos distintos de prompts.

Para determinar el valor óptimo del umbral de similitud *t* (discutido en la Sección 2.2.1) hemos realizado experimentos considerando diversos valores de dicho umbral. Usando el simulador de usuarios, y empleado *EscenariosB*, hemos generado un corpus de 300 diálogos por cada valor de *t*, usando en todos los casos la técnica propuesta. El análisis de estos corpora de diálogos muestra que los mejores resultados se obtienen cuando *t* = 0,5.

Empleando el valor óptimo de *t*, se ha vuelto a usar el simulador de usuarios y *EscenariosA* para generar otro corpus de 900 diálogos. La Tabla 2 muestra los resultados medios (en %) obtenidos a partir de la evaluación de dicho corpus.

WA	SU	IR	TC
84,62	71,25	13,20	68,32

Tabla 2. Resultados de evaluación usando el sistema de RAH *mejorado*

4. CONCLUSIONES Y TRABAJO FUTURO

Comparando las Tablas 1 y 2 se observa que la técnica propuesta permite mejorar las tasas de exactitud de palabras (WA), comprensión de habla (SU), recuperación implícita (IR) y logro de tareas (TC) en 8,5%, 16,54%, 4% y 43,81% absoluto, respectivamente. Una línea de trabajo futuro es aplicar la técnica propuesta en otros sistemas de diálogo que empleen otros tipos de frases. Otra línea consiste en estudiar métodos alternativos para determinar el valor de confianza de las palabras empleadas en las correcciones. Por ejemplo, un método alternativo podría consistir en considerar un valor proporcional al número de fonemas en común entre la palabra errónea y la palabra que se usa para realizar la corrección.

5. BIBLIOGRAFÍA

[1] E. K. Ringger, J. F. Allen, "A fertility model for post correction of continuous speech recognition", Proc. ANLP-NAACL Satellite Workshop, pp. 1-6, 2000.

[2] Z. Zhou, H. Meng, "A two-level schemata for detecting recognition errors", Proc. ICSLP, pp. 449-452, 2004.

[3] M. Jeong, B. Kim, G. G. Lee, "Semantic-oriented correction for spoken query processing", Proc. ICSLP, pp. 897-900, 1996

[4] W. M. Fisher, J. G. Fiscus, "Better alignment procedures for speech recognition evaluation", Proc. ICASSP, pp. 59-62, 1993

[5] F. Crestani, "Word recognition errors and relevance feedback in spoken query processing", Proc. Conf. on Flexible Query Answering Systems, pp. 267-281, 2000

[6] R. López-Cózar, Z. Callejas, M. McTear, "Testing the performance of a spoken dialogue system by means of a new artificially simulated user", Artificial Intelligence Review, 26, pp. 291-323, 2006

[7] R. López-Cózar, Z. Callejas, "Combining language models in the input interface of a spoken dialogue system", Computer Speech and Language, 20, pp. 420-440

[8] M. Danieli, E. Gerbino, "Metrics for evaluating dialogue strategies in a spoken dialogue system", Proc. AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation, pp. 34-39, 1995

TRANSCRIPCIÓN FONÉTICA EN UN ENTORNO PLURILINGÜE

Tatyana Polyákova, Antonio Bonafonte

Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

RESUMEN

España es un país plurilingüe, lo cual es sin duda una gran riqueza, pero a veces también es una dificultad añadida para las tecnologías del habla. Cada vez más las aplicaciones de voz tienen que ser adaptadas al ámbito multilingüe, ya que se pretende que tengan la mayor difusión y utilidad posibles.
La UPC está participando en el proyecto AVIVAVOZ cuyo objetivo es crear un sistema completo de traducción de voz a voz capaz de realizar traducciones entre las lenguas oficiales del Estado español. Nuestro grupo de síntesis tiene como objetivo conseguir que el sistema lea correctamente textos con muchas palabras de otras lenguas. Con ese fin hemos desarrollado sistemas de identificación de la lengua y de conversión fonética específicos para cada lengua en cuestión, junto a una serie de reglas de nativización. Los resultados obtenidos para la identificación de la lengua son buenos tanto para párrafos como para palabras aisladas (como nombres propios). La calidad de síntesis fue mejorada, utilizando sistemas de conversión fonética específicos para cada lengua y aplicando reglas de nativización.

1. INTRODUCCIÓN

En la era de la globalización y del multilingüismo nos encontramos con un abanico de nombres propios de diversos orígenes en todo tipo de ámbitos, como por ejemplo conversaciones, noticiarios, prensa, etc. La globalización de los medios de comunicación, la interacción económica a nivel mundial, el emergente desarrollo de tecnologías de alta gama, sin mencionar la movilidad internacional de recursos humanos, aportan una gran variedad lingüística a nuestro día a día. Los textos multilingües ya no sorprenden a nadie, pero, sin embargo, presentan una dificultad para muchos lectores. En los países de habla no sajona, el uso de anglicismos es creciente. En España cada vez más podemos oír nombres y apellidos procedentes del mundo entero.
El problema de la pronunciación de textos multilingües se puede descomponer en dos partes: (i) mejora de pronunciación de nombres propios, y (ii) mejora de pronunciación de palabras y frases extranjeras dentro del texto en otro idioma. Los nombres propios, sin embargo, son difíciles de pronunciar incluso para los humanos, dado que su pronunciación depende de una serie de factores tales como nivel de asimilación fonética y ortográfica, la popularidad de los mismos en un contexto dado, y también de las preferencias individuales de la persona portadora del nombre [7].
En [1], Font Llitjós probó que el hecho de saber el origen del nombre propio, su pertenencia a una de las familias lingüísticas o ambos, ayuda a mejorar su transcripción fonética.
El identificador de la lengua presentado por los autores consistía en modelos n-grama entrenados para cada lengua a partir de una base de datos que incluía los marcadores de principio y final de palabra. Para cada palabra de entrada, se buscaban todos los trigramas y se estimaba su probabilidad de pertenecer a cada una de las lenguas, de modo que la lengua con mayor probabilidad se elegía como lengua de origen de la palabra. La información de la lengua de origen se podía incorporar de manera directa o indirecta. La manera directa consistía en entrenar un sistema de conversión fonética independiente para cada lengua, mientras la indirecta permitía más flexibilidad incluyéndola sólo como característica adicional en un clasificador, usándola en los casos cuando era relevante. El objetivo en [1] era pronunciar todos los nombres propios correctamente, desde el punto de vista de la gramática americana, o, en otras palabras, americanizarlos. Los n-gramas también se usaron en [3] y [4] para identificar la lengua. En [3], aparte de los n-gramas, se propuso usar clústeres de letras basados en sílabas, siendo éstas unidades estables que contienen más información lingüística que las letras.
Los objetivos de este trabajo consisten en mejorar la transcripción fonética de palabras extranjeras y de nombres propios de origen extranjero. Con el fin de alcanzar un alto nivel de inteligibilidad del habla sintetizada proponemos adaptar la pronunciación de la palabra extranjera a la lengua del texto, sabiendo que la voz nativizada es más fácil de entender que la voz sintetizada a partir de fonemas extranjeros [5]. Para España este es un problema de gran importancia, debido a la existencia de al menos tres grandes regiones bilingües, donde las lenguas específicas de la región se usan con la misma naturalidad (frecuencia) y en los mismos ámbitos que el castellano. Nos vamos a centrar en lo relacionado con Cataluña.
En catalán, igual que en castellano, las palabras extranjeras tienden a adquirir una pronunciación "nativizada" basándose en los correspondientes conjuntos de fonemas. No obstante, los nombres españoles en catalán, la mayoría de las veces, se pronuncian con fonemas castellanos. Por ejemplo, el nombre Jorge en catalán se lee /x 'o r x e/ y no /dZ 'o r dZ e/ como sugerirían las reglas de fonética catalana, a pesar de que el fonema /x/ no existe en catalán.
Sirva como ejemplo ilustrativo el fenómeno del "Spanglish", cuando un hablante bilingüe de español e inglés en EE.UU. cambia de un idioma a otro de forma espontánea, en mitad de una frase, con una pronunciación impecable en cada lengua. Asimismo, en Cataluña se pueden oír frases bilingües y existen textos escritos en castellano con un gran porcentaje de nombres propios o palabras catalanas y viceversa. Tampoco podemos olvidar las palabras inglesas cuyo uso es elevado en todos los ámbitos. El fenómeno de multilingüismo, tan frecuente en la prensa, foros, programas de televisión, correos electrónicos, SMS, etc., necesita una atención especial. Se ha de precisar que en las sociedades bilingües españolas, se da más importancia en los medios de comunicación a la lengua "autonómica", de modo que existen más textos en castellano con muchas palabras catalanas que al revés.
Los siguientes ejemplos ilustran diferentes ocurrencias del multilingüismo. En la Figura 1 tenemos un fragmento de un foro de discusión de noticias del diario "AVUI" editado en

catalán. El primer comentario está en catalán e inglés, mientras el segundo está en castellano.

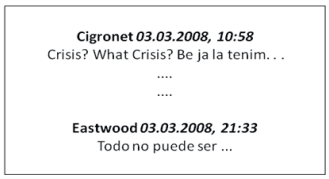


Figura 1. *Comentarios multilingües en el diario "AVUI".*

A continuación tenemos otro ejemplo de un texto en castellano con muchos nombres propios ingleses.

Figura 2. *Fragmento de un reportaje sobre las elecciones primarias en EE.UU.*

Existen muchos más ejemplos donde la transcripción fonética depende de si se conoce o no la lengua de la palabra.

2. SISTEMA DE TRASCRIPCIÓN FONÉTICA MULTILINGÜE

Tener una herramienta capaz de determinar la lengua del párrafo es muy importante a la hora de tratar información multilingüe procedente de diarios, foros, correos electrónicos, artículos científicos, manuales técnicos, páginas web, y otras fuentes donde la lengua del párrafo puede no saberse a priori o puede cambiar de forma repentina. En casos como esos, sabiendo la lengua podemos mejorar la calidad de síntesis considerablemente.

Una vez la lengua del párrafo quede determinada, es necesario identificar la lengua de cada una de las palabras de forma aislada. Esto es importante por dos razones: (i) para mejorar la pronunciación de las palabras extranjeras, y (ii) para la adaptación de esa pronunciación a la lengua del párrafo. Noticias internacionales y deportivas abundan en nombres propios de diversos orígenes mientras en los foros, comentarios en las páginas web, correos electrónicos, posters de publicidad de líneas aéreas (p.ej. Vueling), SMS, etc., hay mucha mezcla lingüística y la probabilidad de encontrar una palabra española o catalana en un texto en inglés, o al revés, es muy alta.

El diagrama abajo muestra nuestro sistema de identificación de la lengua y de nativización de la pronunciación. El primer paso del método consiste en determinar la lengua de cada párrafo. Pongamos que sea la lengua por defecto "LANG" o llamémosla "lengua destino".

La búsqueda de la transcripción fonética se hace para cada palabra de forma aislada. Primero se averigua si la palabra, incluyendo POS, está en el diccionario de la lengua destino. Si es así, la correspondiente transcripción fonética se considera válida; si no, el siguiente paso consiste en averiguar si la lengua de la palabra es distinta de la lengua destino del texto. Para ello, se efectúa una llamada al módulo identificador de la lengua.

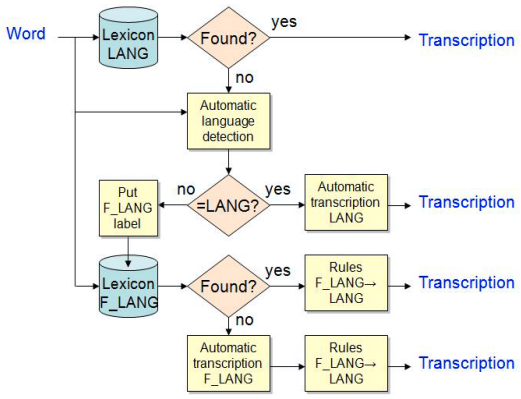


Figura 3. *Sistema de transcripción fonética multilingüe.*

Si efectivamente resulta ser distinta, la palabra se etiqueta con la lengua origen proporcionada por el identificador, F_LANG. En caso contrario, la pronunciación se deriva usando el sistema automático de transcripción fonética para la lengua destino. A continuación, la transcripción fonética de las palabras con la etiqueta de una cierta lengua origen se buscan en el diccionario de esa lengua. Antes de validar la pronunciación, se aplican las reglas de "nativización" entre la lengua origen y la lengua destino. Si la palabra no se encuentra en el diccionario de la lengua origen, se procesa con el sistema automático de transcripción fonética de esa lengua y luego, igual que antes, se aplican las reglas de "nativización". El proceso de nativización se explica en la sección 3.3. Es importante enfatizar que si la palabra es de origen extranjero pero se encuentra en el diccionario de la lengua destino, se considera nativizada. Por eso no hay necesidad de identificar la lengua antes del primer paso.

2.1. Identificación de la lengua

Para identificar la lengua hemos implementado el modelo n-grama estándar, previamente utilizado para la misma tarea en [1, 3, 4]. Se estima un n-grama para cada lengua en cuestión. Los n-gramas incluyen los marcadores del principio <s> y final <s/> de la palabra: <s>mi<s/><s>casa<s/>.

$$Lng^{*} = \arg\max_{Lng} p\left(l_1 \ldots l_n \middle| Lng\right)$$

La lengua del párrafo se determina teniendo en cuenta la lengua de cada una de las palabras aisladas.

2.2. Trascripción fonética

En nuestro anterior trabajo [6] obtuvimos una tasa de palabras correctas igual a 79.63%, utilizando traductores de estados finitos [8] en combinación con el método de aprendizaje a partir de errores [9], para el diccionario LC-STAR de inglés americano [10]. Los traductores de estados finitos constituyen un método poco costoso en tiempo pero eficaz en cuanto a resultados, lo que justifica su utilización para obtener la pronunciación de palabras desconocidas en inglés. La mayoría de los métodos necesitan un diccionario alineado para entrenamiento. En este caso el alineamiento es similar al de [11]. Para trascripción fonética de castellano y catalán

utilizamos transcriptores basados en reglas, ambos desarrollados en la UPC.

2.3. Nativización

La nativización es un proceso de adaptación de la pronunciación de la lengua origen a la pronunciación más cercana, y al mismo tiempo correcta, en la lengua destino.
Es importante destacar que hay una gran diferencia entre el habla nativizada y el habla no nativa (o con acento extranjero) [5]. En muchos aspectos el habla nativizada tiene ventajas sobre la no nativa. El habla no nativa se diferencia del habla nativa en los puntos articulación, distribución de la pausas, elección de las palabras, conducta en las fronteras entre palabras, errores de pronunciación, existencia de fonemas suspendidos entre pronunciación correcta e incorrecta, etc., mientras que el habla nativizada conserva el punto de articulación de la lengua origen, no implica errores de pronunciación, ya que sólo pretende moldear de la mejor manera la pronunciación de la palabra para que encaje suavemente en las oraciones en la lengua destino. La transcripción nativizada está basada en reglas concretas y no da lugar a fonemas medio-nativos o mal pronunciados.
Se ha trabajado en la síntesis tanto en catalán como en castellano, dado que se disponía de sintetizadores en estas dos lenguas. Antes de llevar a cabo los experimentos, se definieron los conjuntos de fonemas (*phonesets*), para cada lengua. En el caso del catalán, se definió un nuevo conjunto enriquecido al que se llamó "CAT+", que además de los fonemas propios del catalán contiene 5 fonemas propios del castellano: /x/, /T/, y las tres vocales átonas /a/, /o/ y /e/, que no existen en catalán. Esto fue posible gracias a que los locutores cuyas voces se emplearon en el sintetizador eran perfectamente bilingües. De hecho, es un fenómeno propio de las sociedades bilingües el uso de un *phoneset* que combine los fonemas de las dos lenguas coexistentes, como ocurriría también en el caso del Spanglish. El *phoneset* "CAT+" es ventajoso para la adaptación al catalán de palabras tanto inglesas como procedentes de otras lenguas oficiales del estado español, a que tiene más fonemas en común con cada una de ellas que el *phoneset* básico tanto de catalán como de castellano. En el caso del castellano, en cambio, los locutores cuyas voces se usaron para la construcción del sintetizador solamente dominaban este idioma, de modo que el *phoneset* utilizado fue el estándar.
Se desarrollaron manualmente tablas de nativización para adaptar el inglés americano (EN-US), euskera (EU), castellano (ES), y gallego (GA) al catalán enriquecido (CAT+). Del mismo modo, se desarrollaron tablas para nativizar al castellano los idiomas EN-US, EU, CA (catalán con *phoneset* estándar), y GA. En nuestros experimentos hemos utilizado sólo las tablas correspondientes a EN-US → CAT+, ES → CAT+, CAT → ES, EN-US → ES. Se prevé trabajar con otros pares de lenguas en el futuro.
La nativización se realiza fonema a fonema, utilizando las tablas correspondientes. En los casos ambiguos, donde utilizando la tabla se obtiene un error evidente de manera reiterada, se usa como ayuda la transcripción fonética de la palabra extranjera obtenida para la lengua destino. Por ejemplo, si la palabra en la lengua origen es *talent* con transcripción / t ‘ae l @ n t`/ y la tabla dice que la shwa /@/en inglés pasa a ser una /a/ en castellano, pero la transcripción por reglas en castellano dice que en esa posición hay una /e/, la transcripción nativizada va a tener una /e/.

3. EXPERIMENTOS Y RESULTADOS

Los experimentos se hicieron para la identificación de la lengua de párrafos y de nombres propios. Para la evaluación de la transcripción nativizada se utilizó el sintetizador Ogmios [12] basado en la selección de unidades, entrenado con 10 horas de voz. Se sintetizaron varias frases antes y después de aplicar las reglas de nativización. La calidad de las mismas fue puntuada por oyentes sin experiencia en la síntesis del habla, para obtener una valoración lo menos influenciada posible.

3.1. Descripción de la base de datos.

Para el experimento de la identificación de la lengua del párrafo se consideraron las siguientes lenguas: catalán, castellano, euskera, gallego e inglés.
En el marco del proyecto AVIVAVOZ se desarrollaron varios corpus bilingües para la traducción estadística, disponible para catalán, euskera, gallego y castellano. Las frases y sus traducciones se extrajeron de la revista de consumidores Eroski. La presencia de nombres propios es insignificante en comparación con la de nombres comunes. El modelo de lenguaje para inglés se entrenó a partir de los nombres comunes del diccionario LC-STAR.
Para los experimentos de identificación de la lengua de nombres propios (palabras aisladas), creamos una base de datos de nombres y apellidos gallegos de tamaño de ~3600 palabras, y otra base de datos de nombres y apellidos vascos (alrededor de 11200 palabras). Para castellano y catalán sólo consideramos los nombres propios de personas, marcados como tales en los diccionarios de nombres propios creados en el marco del proyecto LC-STAR para estas lenguas [10] (alrededor de 20-27 mil palabras). Hay que tener en cuenta que cada diccionario fue filtrado para eliminar los nombres presentes en cualquier otro diccionario, de modo que todas las entradas son únicas. Eso fue necesario para “pulir” los modelos de los nombres que se escriben igual en varias lenguas, p.ej. David /d a B ‘i D/ en español y /d ‘ae v I d/ en inglés. Los modelos de lenguaje se entrenaron a partir del 90% del corpus Eroski, y el 10% restante fue reservado para la evaluación. Se emplearon los mismos porcentajes para entrenar los modelos de nombres propios.

3.2. Resultados de identificación de la lengua.

La Tabla 1 es una tabla de confusión para las cuatro lenguas oficiales del Estado español (las comunidades autónomas) y además el inglés.

Lang. es	ca	es	eu	ga	en
ca	95.2	1.2	2.1	1.5	-
es	1.9	90.9	3	0.1	-
eu	0.6	0.8	92.6	8	-
ga	1.9	0.8	2.1	89.2	-
en	0.6	0.3	0.3	1.2	-

Tabla 1. *Resultados de identificación de la lengua del párrafo para el corpus Eroski.*

Los mejores resultados fueron obtenidos para catalán y euskera, y los peores para el gallego, lo que puede deberse a su alto grado de parecido con el castellano y catalán, que a su vez tienen algunas palabras en común.

La mayoría de los errores se deben a los siguientes factores:
- Presencia de palabras extranjeras como Kodak, Kellogg's, Fuji, etc.
- A las palabras y fragmentos de frases que pueden pertenecer a varias lenguas al mismo tiempo como "agua mineral natural".
- Dígitos y abreviaciones: 10g, rayos UVA, etc.

En la Tabla 2 se muestran los resultados de identificación de la lengua de nombres propios para 200 palabras elegidas aleatoriamente dentro del corpus de evaluación.

Languages	ca	es	eu	ga	en
ca	77	33	5	25	9
es	46	98	12	29	11
eu	6	12	170	6	3
ga	37	24	6	121	9
en	31	33	7	19	168

Tabla 2. *Resultados de identificación de la lengua usando modelos de lenguaje "pulidos".*

Aquí los mejores resultados se obtuvieron para el euskera, seguidos por el inglés y el gallego.

3.3. Test perceptual

Para validar la metodología se ha realizado una primera evaluación de las reglas de nativización mediante un test perceptual. Primero se crearon dos corpus multilingües, cada uno de los cuales constaba de 1000 palabras, uno en castellano con ~50 palabras inglesas y ~50 catalanas y otro en catalán con ~ 50 palabras españolas y ~50 inglesas.

Dado que los oyentes que participaron en la evaluación del sistema tenían conocimientos de castellano, catalán e inglés pero no de euskera ni gallego, se diseñó un test para palabras procedentes de los tres primeros idiomas. Para evaluar la inteligibilidad de la síntesis en catalán y castellano siguiendo el esquema presentado en la Figura 3, se eligieron aleatoriamente y se sintetizaron 10 frases, 5 de cada corpus, Las etiquetas F_LANG fueron añadidas a mano. Se evaluaron la inteligibilidad y la naturalidad de las frases sintetizadas. Entre 10 oyentes, para las 5 frases en castellano, el 70% consideró las frases nativizadas más naturales el 62%, más inteligibles, el 16 % no notó diferencia significativa en la naturalidad y el 28% en la inteligibilidad. El 14% de los oyentes opinó que las frases nativizadas eran menos naturales y el 12% que eran menos inteligibles que la síntesis básica.

Para 5 frases en catalán de los mismos diez oyentes, el 44% opinó que la síntesis nativizada sonaba más natural que la básica, el 36% que sonaba igual de natural que la otra y el 20% que sonaba peor. En cuanto a la inteligibilidad el 26% de oyentes decidió que se entendía mejor, el 48% que se entendía igual, y el 28% que se entendía peor. Los resultados para castellano se pueden considerar bastante buenos. El hecho de que muchos oyentes prefirieron la síntesis básica en catalán se debe principalmente a la ausencia de algunos difonemas en la voz en catalán, que aunque la transcripción fonética fuera la adecuada, hicieron que aparecieran algunos artefactos muy molestos al oído, a su vez afectando a la opinión de los oyentes. Otras fuentes de errores son el alineamiento de los diccionarios de entrenamiento, que a veces introduce ambigüedad, y las reglas de nativización, que en el caso de palabras con comportamiento excepcional no siempre dan los resultados esperados.

4. CONCLUSIONES

De los resultados descritos en el apartado anterior se puede concluir que el sistema de identificación de lengua funciona bien para párrafos y ligeramente peor para palabras aisladas. Una evaluación preliminar indica que sabiendo la lengua y usando el módulo de transcripción fonética multilingüe se consigue mejorar la inteligibilidad y la naturalidad del habla sintetizada en castellano y en algunos casos en catalán. En el futuro se trabajará en la integración del módulo de identificación de la lengua con el módulo de nativización de la pronunciación.

5. AGRADECIMIENTOS

Este trabajo ha sido financiado con el proyecto AVIVAVOZ (TEC2006-13694-C03) y la beca FPU (AP2005-4526).

6. BIBLIOGRAFÍA

[1] Llitjós, A., and Black, A., "Knowledge of language origin improves pronunciation of proper names", Proceedings of EuroSpeech-01, 1919-1922, 2001.

[2] Lewis, S., and McGrath, K., "Language identification and language specific letter-to-sound rules", Colorado Research in linguistics, 17 (1), 1-8, 2004.

[3]. Chen, Y., You, J. Chu. M., Zhao, Y., Wang, J., "Identifying language origin of person names with n-grams of different units", ICASSP 2006, Toulouse, France

[4] Litjós, A., "Improving pronunciation accuracy of proper names with language origin classes", Master Thesis, CMU-LTI-01-169, Carnegie Mellon University, Aug 2001.

[5] Schultz, T. and Kirchhoff, K., "Multilingual speech synthesis", Elseiver, USA 2006

[6] Polyákova, T., Bonafonte, A., "Learning from errors in grapheme-to-phoneme conversions", International Conference on Spoken Language Processing, Pittsburg, USA, 2006

[7] Lindström, A., "English and other foreign linguistics elements in spoken Swedish", Linköping University Linköping, 2004

[8] Galescu L., J. Allen, "Bi-directional Conversion Between Graphemes and Phonemes Using a Joint N-gram Model", In Proc. of the 4th ISCA Tutorial and Research Workshop on Speech Synthesis, Perthshire, Scotland, 2001

[9] Brill E., "Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging", Computational linguistics 21(4), pp. 543-565, 1995

[10] http://www.lcstar.org

[11] Damper R. I., Marchand Y., Marsterns J.-D. and Bazin A., "Aligning letters and phonemes for speech synthesis" in Proceedings of the 5thISCA Speech Syntesis Workshop, Pittsburgh, 209-214., 2004

[12] Bonafonte A., Adell J., Agüero, Erro D., Esquerra I., Moreno A., Pérez J., Polyákova T., "The UPC TTS System Description for the 2007 Blizzard Challenge", 6th ISCA Workshop on Speech Synthesis, Bonn, Germany, August '07.

VOICE PLEASANTNESS: ON THE IMPROVEMENT OF TTS VOICE QUALITY

Luis Coelho[1], *Daniela Braga*[2], *Carmen Garcia-Mateo*[3]

[1] Instituto Politécnico do Porto, ESEIG, Porto, Portugal
[2] MLDC - Microsoft Language Development Center, Lisbon, Portugal
[3] Universidad de Vigo, Dpto. Teoria de la Señal e Telecomunicaciones, Vigo, Spain

RESUMEN

The aim of this paper is to validate the objective description of the voice pleasantness concept. This concept has been objectively defined on an exhaustive study based on subjective voice analysis, comparison and scoring[1, 2]. For increasing text-to-speech (TTS) voice pleasantness while maintaining speaker's identity a set of procedures, that operate on time and frequency domains and encompass phonetic and prosodic levels, are proposed. These enhancement procedures have been integrated in widely know speech engines and the obtained results showed an effective gain on the evaluated parameters providing support to both methodology and objectives. The Mean Opinion Score (MOS) performance evaluations indicated pleasantness improvements of 7% for female voices and 3% for male voices when compared with systems that do not consider these quality aspects.

1. INTRODUCTION

The main goal on speech synthesis systems is the production of high quality speech with the maximum naturalness. The latest developments in acoustic engines are slowly fulfilling these requirements and TTS technology is increasingly being included in our daily lives. Front-end design has also improved and can now deliver rich prosodic information to the acoustic engine which efficiently produces human like speech. Our concerns regarding voice enhancement and quality assessment are often associated, in specialized literature, with voice impairments or disorders [3, 4]. However for a continuous use of TTS systems on a daily basis additional quality requirements arise. On another perspective, considering that no voice pathologies are present, the interaction with a machine's voice must be everything but boring and if possible it should be pleasant. It is our belief that one of the next frontiers on TTS technology resides on improving voice quality in order to help to increase interaction pleasantness.

The rest of the paper is organised as follows: In section 2 we describe the used methodology for processing the speech signal. In section 3, a quick presentation is made on what are parameters can be used to define pleasantness and how the quality goal is obtained. In section 4 we show several performance evaluation tests and an extensive discussion of the results. Finally, in section 5, main conclusions are pointed out and future work is foreseen.

2. METHODOLOGY

To validate the objective pleasantness definition we propose a set of adaptive phone level signal operations for voice enhancement according to the pleasantness parameters. These procedures can be easily integrated with an existent system as an extension, independently of the technology, or can be directly included as a part of the processing operations. Traditional voice conversion techniques [5, 6] involve intense signal manipulation which often lead to quality degradation. In our case quality is a main concern and speaker's identity must be preserved so we decided to support our signal operations with mature proven models.

2.1. Integration with TTS as an output module

Signal processing operations will be performed on the TTS' output speech. The required information consists of phoneme sequence, timing, voiced/unvoiced and prosodic tags with pitch, all given by the front-end, and additionally an enhancement goal in percentage, for each parameter.

The time domain signal decomposition/composition is based on the Time Domain Pitch Synchronous Overlap Add (TD-PSOLA) algorithm [7] since it introduces minimal distortions and is a very effective way to perform small pitch changes. Segment durations are changed accordind to a context and phoneme dependent ratio:

$$d_{new}(p) = \frac{1}{2C+1} \sum_{i=p-C}^{p+C} \frac{d_{target}(i)}{d_{cur}(i)} \qquad (1)$$

where d_{new} is the new duration for segment p, d_{ref} and d_{cur} are the target and current durations respectively and C is a neighbourhod limiter for analysis. In our case most considerer segments are phonemes and a neighborhood of 2 segments, before and after, was used ($C = 2$).

The frequency domain operations are performed by a filter that is calculated for each window considering the

current phoneme. The filter is defined according to the following procedure:

$$s(k) = \sum_i a_i s(k-i) + G.u(k) \quad (2)$$

is adjusted to a source-filter model based on an all-pole filter

$$H(z) = \frac{G}{1 - \sum_{i=1}^{N} a_i z^{-i}} = \frac{G'}{\sum_j (z^{-1} - p_j)} \quad (3)$$

with $U(z)$ as the excitation, G as the model gain and p_j as filter poles. For a sufficient model order N the signal can be successfully reconstructed since both filter and excitation information are known. In our case the modified covariance method [8] was used for Auto-Regressive (AR) estimation, because it leads to stable models, it can efficiently deal with wrong model orders and because it introduces reduced spectral line splitting effects. 2. Extract formant frequencies and bandwidths. For each filter pole p_j with magnitude A_j and frequency w_j, the formant frequency F_j, for a sampling frequency f_s comes as:

$$F_j = f_s \frac{w_j}{2\pi} \quad (4)$$

The bandwidth B_j are approximated by:

$$B_j = f_s \frac{A_j}{\pi} \quad (5)$$

3. Adjust formant frequencies by relocating filter poles and perform formant sharpening. Each (F_j, B_j) pair is then manipulated to fit the given context dependent criteria.

With all this it is possible to build the correction filter that will be applied to each specific window. A discrete Itakura-Saito based metric is used to evaluate the spectral distance between original and enhanced signal:

$$d_{IS} = \frac{1}{N} \sum_{m=1}^{N} \left(\frac{P(\omega_m)}{\hat{P}(\omega_m)} - \log \frac{P(\omega_m)}{\hat{P}(\omega_m)} \right) \quad (6)$$

A threshold limits the extent of spectral changes in the signal.

4. Spectral weighting. Finally, for reducing the effects of noise amplification introduced by the correction filter, an extra spectral weighting filter is used. The filter is based on Linear Prediction (LP) coefficients using a zero/pole transfer function with coefficient weighting:

$$W(z) = \frac{A(z/\alpha)}{A(z/\beta)} = \frac{1 + \sum_k a^k z^{-k}}{1 + \sum_l a^l z^{-l}} \quad (7)$$

The values for α and β are computed for each window according to the previously extracted spectral envelope and, to keep a low distortion between sudden changes on the parameters, an interpolation is performed considering the previous values. Initial values are found by first adjusting α and then β, in successive iteration, with spectral distance restrictions. Tipical values are $\alpha \in [0.85, 1.00]$ and $\beta \in [0.82, 0.95]$.

2.2. Integration with HMM based TTS

We also applied the voice enhancement criteria to an HMM based synthesis system [9]. Since these systems are centered on time-frequency statistical sound models, the voice changes can be performed by small arrangements on the model's statistical parameters. In our case we trained a voice font with what we considered to be an optimal voice (our optimal voice resulted from a set of several subjective tests from which the enhancement parameters were derived) and using a weighted re-estimation process, based on the Baum-Welch algorithm, we adapted the desired voice font. The binary trees that help to determine the synthesis parameters were also rebuilt. All the process was conducted during the training and no changes were made to the speech production engine.

3. VOICE QUALITY

The studies performed on [1] and [2] correlate objective parameters such as formant frequencies, fundamental frequency, duration, etc. with with subjective scores. In an unpublished work of the authors the study is extended to European Spanish, French and English with some preliminary results presented here. The subjective classification was performed on a five points scale (1 for the worst opinion and 5 for the best opinion) and the correlation values were obtained using the standard correlation equation:

$$correl(X,Y) = \frac{\sum_i^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i^N (x_i - \bar{x})^2 \sum_i^N (y_i - \bar{y})^2}} \quad (8)$$

where $X = \{x_1, \ldots, x_N\}$ and $Y = \{y_1, \ldots, y_N\}$ are the set of objective results and the subjective scores respectively.

The objective evaluation for each voice and for each group show different results but share some common indicators. The results for the most representative parameters, in all the five languages, are presented averaged in table 1. The fundamental frequency is analysed considering the average, maximum and minimum, standard deviation and the difference between maximum and minimum values. Speaking rate (SPR) is also considered as the number of words per time unit (in this case words per second) and in percentage as the relation between the speaking time and non-speaking/pause time (a pause was defined as any non-speaking segment with duration greater than 20ms). The energy and the intensity were normalized before comparison.

The obtained values can be biased if a group of speakers share some similar voice characteristic. To avoid this

Objective Feature	Score Correlation Average	St. Dev.
F0 Average (Hz)	- 0.56	0.052
F0 Min (Hz)	- 0.45	0.042
F0 Max (Hz)	- 0.67	0.023
F0 Std. Dev (Hz)	0.29	0.015
F0 range (Hz)	0.10	0.012
SPR (words/sec)	0.17	0.011
SPR (%)	- 0.28	0.018
Pause Rate (%)	0.30	0.011
Total Energy (dB)	- 0.54	0.034
Energy St. Dev.	- 0.52	0.030
Avg. Intensity	- 0.59	0.038

Table 1. Correlation results between objective features and voice classification scores

Phon.	Formant	Frequency EP	Frequency BP	Bandwidth EP	Bandwidth BP
A	1	- 0.33	- 0.36	- 0.20	- 0.18
	2	- 0.31	- 0.30	- 0.18	- 0.19
6	1	- 0.41	- 0.57	0.30	0.36
	2	- 0.32	- 0.52	- 0.21	- 0.27
i	1	- 0.51	- 0.32	- 0.19	- 0.19
	2	- 0.27	- 0.28	- 0.12	- 0.11
o	1	- 0.01	0.04	0.27	0.29
	2	- 0.05	0.02	0.17	0.17
E	1	0.15	- 0.43	- 0.18	- 0.14
	2	- 0.09	- 0.11	- 0.21	- 0.19
U	1	- 0.47	- 0.45	- 0.26	- 0.26
	2	- 0.14	0.21	- 0.22	- 0.21

Table 2. Correlation results between formant frequencies and formant bandwidth in some vowels and voice classification scores for European Portuguese (EP) and Brazilian Portuguese (BP)

the selected speakers in our case formed a very heterogeneous group with very distinct voices and speaking styles. From the results the following pleasantness indicators can be pointed: *Low fundamental frequency*, we can see that F0 average showed a high inverse correlation with the obtained voice score which indicates a preference for voices with a lower pitch; *Dynamic prosody*, the voices with greater variation on the prosodic parameters (not all are shown) also seemed to be preferred. Variations on pitch, durations and intensity are valued characteristics from the pleasantness point of view; *High speaking rate*, the fast speakers showed to be prefered nevertheless it is desirable to have well defined pauses between utterances. We suppose that this shows assertiveness and transmits confidence to the human listener.

The pleasantness reference is found by following the methodology described in [1]. It requires a group of candidate speakers for voice-font recording, a set of listeners for evaluations and an application context. The initial set of voices should be heterogeneous but compatible with generic voice pleasantness values (this can vary according to culture, language, etc.). For EP and BP female speakers the objective pleasantness reference is defined as $f_0 = 208 \pm 9Hz$ (95% confidence interval), normalized energy $18.8dB/sec$, 3.8 words/sec for the speaking rate and 15.3% of the time should be used for pauses. The values were obtained by maximizing a quadratic polynomial used to model the relation between objective parameter and score. The model is obtained by minimizing the error for a quadratic curve fitting using each of the analysed parameters:

$$k = \sum_{i=1}^{N} \left[y_i - (b_0 + b_1 x_i + b_2 x_i^2) \right]^2 \qquad (9)$$

where y_i represents the obtained score and x_i represent the parameters in evaluation. The maximum of the obtained curve will point the best values for the parameter.

Signal manipulation is performed on time and spectral domains. In the time domain phoneme level durations are changed by a segment dependent fixed ratio which is obtained from the average segment durations on original and reference voice. If the original speaker has a natural speaking style the introduced changes on duration will not introduce ambiguity on speaker identification. In spectral domain similar ratios are previously calculated for formants and related bandwidths in voiced sounds. These ratios have been limited to $[0.9, 1.1]$ to limit spectral changes. In this domain no changes are made to unvoiced sounds.

4. RESULTS AND DISCUSSION

We have used two EP voice databases, one with a female voice (87 min.) and another with a male voice (100 min.). In spite of all the careful taken with recording scripts and recordings no special attention has been given to speaker selection (there was only one person only good articulation and good dynamics at a subjective level were considered). The voices were objectively analysed and a pleasant voice model was built. The several considered parameters were then used to define a voice goal as described. A set of 10 long sentences (expected to give at least 10 seconds of speech) were chosen and the related speech was produced by a concatenative system based on Festival (CF) and by HTS (HTS) basic configuration, the same sentences were the produced using our proposed voice enhancement procedure (CFVE and HTSVE). A group of 11 listeners with no special speech technology knowledge evaluated the sentences and voted for voice pleasantness according to their preference. In figure 1 we show the results obtained in our subjective evaluation tests. It can be observed that the voice enhancement procedures introduce some improvements on the listener's opinion which means that the signal manipulation doesn't introduce any adverse effects. However the changes are not highly sig-

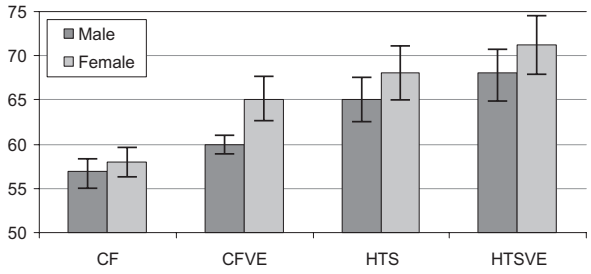


Figure 1. *Results of the subjective tests according to voice pleasantness (90% confidence interval is show in the top of the bars).*

nificant because the speaker's identity was preserved. We expect that the results could show greater improvements if the original voice was a randomly selected voice.

For male voices only an average of 3% of improvements could be measured for both systems. For female speakers there is an increment of around 7% on voice pleasantness preference. The obtained results for HTSVE are not so expressive due to an insufficient amount of trainning material and due to the experienced issues on phoneme adaptation without losing speaker's identity. Nevertheless this suggests that the identified voice quality parameters are important for voice pleasantness and that it is possible to perform voice enhancement by the use of signal manipulation operations.

In figure 2, we can observe an example of the enhancement for a word pronounced by a female speaker. Looking at the spectrum represented above it is possible to see that the formants are better defined while the energy in the spectrum valleys remmains at the same levels, spectral peaks are also sharper. It is possible to see small changes in the durations of the vowels (largest segments).

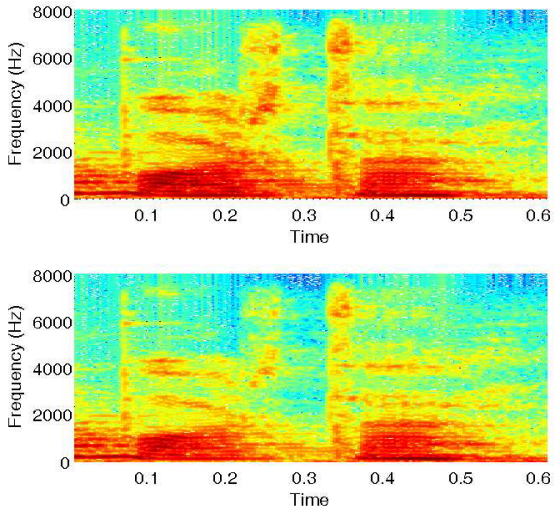


Figure 2. *Spectrum for the word "porta"(door) after (top) and before (bottom) voice enhancement for voice pleasantness.*

5. CONCLUSIONS

In this paper, we presented a set of objective parameters that play an important role on the evaluation of voice quality, particularly concerning pleasantness. It is our belief that these parameters will be of major importance on voice selection and TTS systems' performance evaluations. After defining how to increase the voice quality we have proposed a set of procedures, considering time and frequency domain parameters, which can help to follow this path. We have performed some evaluation experiments and the obtained results show that our proposal can lead to an effective voice enhancement.

6. BIBLIOGRAPHY

[1] Daniela Braga, Luis Coelho, Fernando Gil Resende, y Miguel Dias, "Subjective and objective evaluation of brazilian portuguese tts voice font quality," in *Proc. of Advances in Speec Technology*, 2007.

[2] Daniela Braga, Luis Coelho, y Fernando Gil Resende, "Subjective and objective assessment of tts voice font quality," in *Proc. of SPECOM*, 2007.

[3] J. Kreiman, B.R. Gerratt, G.B. Kempster, y A. Erman, "Perceptual evaluation of voice quality. review, tutorial and a framework for future research," *Journal of Speech and Hearing Research*, pp. 21–40, 1993.

[4] J. Kreiman y B. R. Gerratt, "Sources of listener disagreement in voice quality assessment," *Journal of Acoustical Society of America*, vol. 108, no. 4, pp. 1867–1876, 2000.

[5] A. Kain y M. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proc. of ICASSP*, 1998.

[6] Y. Stylianou, O. Capp'e, y E. Moulines, "Statistical methods for voice quality transformation," in *Proc. of Eurospeech*, 1995.

[7] E. Moulines y F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Communication*, vol. 9, pp. 453–467, 1990.

[8] S. M. Kay, *Modern Spectral Estimation: Theory and Application*, Prentice-Hall, Englewood Cliffs, 1988.

[9] K. Tokuda, T. Yoshimura, T. Kobayashi, y T. Kitamura, "Speech parameter generation algorithms for hmm speech synthesis," in *Proc. of ICASSP*, 2000.

SESIÓN PROYECTOS/DEMOS

A SYSTEM ARCHITECTURE FOR MULTILINGUAL SPOKEN DOCUMENT RETRIEVAL

German Bordel, Arantza Casillas, Mikel Penagarikano, Luis Javier Rodríguez, Amparo Varona

Grupo de Trabajo en Tecnologías Software
Departmento de Electricidad y Electrónica
Universidad del País Vasco

RESUMEN

Finding audio and video resources in internet is becoming an increasingly demanded application. However, search engines are usually limited to adjacent texts (hand supplied transcripts or close captions) to index and classify multimedia documents. Clearly, a key advantage can be taken from using automatic speech recognition and natural language processing technologies, since they allow to transcribe and enrich spoken documents, thus leading to more accurate indexes and more focused search results. In this paper, the architecture of a multilingual (Basque, Spanish, English) spoken document retrieval system is presented. The system, organized around a collection of XML resource descriptors, consists of four main elements: (1) a crawler/downloader that fetches audio and video resources; (2) an audio processing module, which enriches the XML resource descriptors with information extracted from the audio signals; (3) an information retrieval module, which processes the collection of resource descriptors to create an index database, and search this latter to find those resources matching any given query; and (4) a user interface, which allows to formulate queries and access to search results.

Index Terms: Spoken Document Retrieval, Speech Recognition, Natural Language Processing.

1. INTRODUCTION

Search engines are essential tools to find the desired or the most relevant information in internet. Nowadays, finding multimedia (audio and video) resources is becoming as important as finding text resources. However, search engines are limited to adjacent texts (hand supplied transcripts or close captions) to index and classify multimedia documents. These texts are just short descriptions, shallow categorizations or partial transcriptions of the contents, so the resulting index is very coarse and the search cannot focus on specific items.

Clearly, search engines can take a key advantage from using Automatic Speech Recognition (ASR) and Natural Language Processing (NLP) technologies, since they allow to transcribe and enrich spoken documents, thus leading to more accurate indexes and more focused search results. To accomplish this objective, multimedia files must be processed off-line and their contents indexed to allow efficient search [1].

Some systems have been already developed in this way, such as SpeechBot and SpeechFind. SpeechBot [2] was an experimental web-based tool from HP Labs that used speech recognition to create seachable keyword transcripts from thousands of hours of audio content, and then allowed to listen to the material online and read the computer generated transcript. By June 2003, SpeechBot had catalogued more than 17000 hours of multimedia content. SpeechBot was shut down in November 2005, after HP closed their Cambridge Research Lab. SpeechFind [3] is a spoken document retrieval system developed by the Center for Robust Speech Systems at the University of Texas at Dallas. It segments audio input, applies a large-vocabulary continuous speech recognition engine to decode speech segments into text, and generates metadata as a by-product; then, audio, transcripts and metadata are entered into an online repository, which allows the search engine to find those resources matching any given query. SpeechFind is currently used to transcribe the National Gallery of Spoken Words, which covers up to 60000 hours of USA historic recordings from the last 110 years.

SpeechBot and SpeechFind deal with spoken documents in English. In the last years, more systems have been developed to index and search spoken documents in other languages. For instance, the system developed at NTT [4] indexes multimedia contents in Japanese, using audio, speech and visual information. Another interesting proposal was recently presented for indexing spoken documents in Chinese: the ASEKS system [5], which uses keyword spotting technology to create a distributed database of indices in the peer-to-peer network, avoiding the bottleneck of network load common in centralized architectures.

This paper presents the architecture of a Spoken Document Retrieval (SDR) system that works with audio and video resources in Basque, Spanish or English. The system consists of four key elements (see Figure 1): (1) the crawler/downloader; (2) the audio processing module; (3) the information retrieval module; and (4) the user interface. The crawler/downloader fetches audio and video resources from internet or from local repositories. In the case of video resources, only the audio signal is processed. For the speech recognizer to work properly, the audio input is segmented and classified as speech or non-speech and the language in speech segments is identified. The information about segment boundaries, language, word transcription, morphosyntactic analysis, etc. is stored in an XML resource descriptor. The collection of XML resource descriptors is taken as input by the indexer (which is part of the information retrieval module) to build an index database. The search engine traverses this structure and returns a list of audio and video resources related to any given query. A web interface allows the user to formulate queries and process the answers of the SDR system.

The rest of the paper is organized as follows. Section 2 describes the elements involved in collecting and processing audio/video resources, including the XML resource descriptors. Section 3 describes the information retrieval module, which in-

This work has been partially supported by the Government of the Basque Country, under program SAIOTEK, projects S-PE05UN32 and S-PE06UN48.

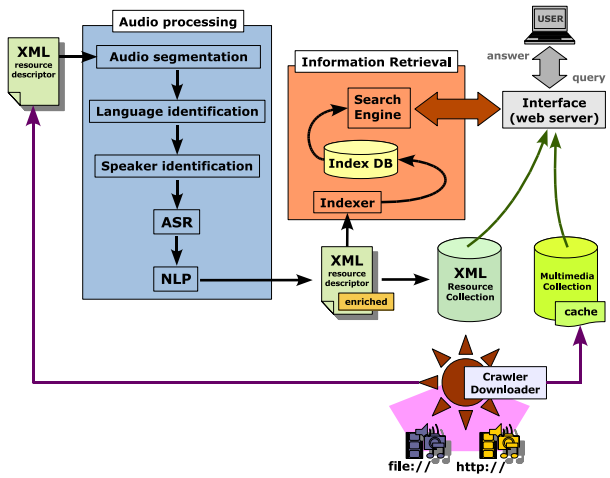


Figura 1. The architecture of the SDR system.

cludes the representation of information, the indexer and the search engine; Section 4 briefly explains the funcionality of the user interface; finally, the main features of the SDR system are summarized in Section 5.

2. COLLECTING AND PROCESSING RESOURCES

Audio and video resources are fetched and processed offline. Copies of the original resources are kept locally, and the audio streams are converted into PCM format for further processing. Audio signals are segmented and classified, the speech segments are transcribed and the resulting sequences of words are morpho-syntactically analysed and disambiguated. The information obtained at each step is incrementally stored in an XML resource descriptor. The elements involved in this process are described in the following paragraphs.

2.1. The crawler/downloader

Two different kinds of resources are considered:
– *Multimedia files obtained from internet.* In this case, a robot explores the web, fetches video and audio files, generates URL lists and creates cached copies for audio processing and indexing. The robot implementation uses the *Nutch* package [6] from the *Lucene* project [7]. Speech signals taken from internet show a high variability in all dimensions: file format, audio coding, noise level, speaker, topic, modality (planned vs. spontaneous speech), etc.
– *A repository of multimedia files.* Under this category we consider audio or video resources acquired in specific controlled situations (for instance, broadcast news or meeting recordings), which are downloaded in a straightforward way from the local filesystem. This kind of resources shows a high variability along some dimensions such as speaker, environmental noise, topic or speech modality, whereas other features such as file format, audio coding or channel conditions are usually fixed.

Each resource is identified by its SHA1 hash. This way we can identify copies of the same resource at different locations and avoid redundant processing and indexing. Audio signals are extracted from multimedia resources by means of a multimedia player. Currently the free *Mplayer* [8] is being used to obtain audio files in PCM format. For presentation purposes, cached copies of the original multimedia resources (audio or video) are saved in Flash video format. Flash video is the more suitable format for resource storage because it provides an easy way to serve multimedia contents over the web to a broad audience. To create Flash videos the free *ffmpeg* software [9] is used.

2.2. Audio processing

For each multimedia file, an XML resource descriptor is generated describing its audio contents. Audio is processed in several steps, all of them (except for NLP) accomplished through the Sautrela system [10]. At each step, the XML document is enriched with information specific to a knowledge level.

2.2.1. Audio Segmentation

This task consists of dividing a continuous audio stream into acoustically homogeneous regions called *segments*. There are robust and unsupervised techniques for doing it. In particular, the identification of speech and non-speech segments is a key step. If non-speech segments are excluded from recognition, not only computation time is saved, but also better transcriptions are obtained. Small interruptions like coughs and other noises produced by the speaker are admitted inside of a speech segment.

2.2.2. Language Identification

Identifying the target language is indispensable for the speech recognizer to use adequate acoustic and syntactic models, and for the NLP tools to apply the linguistic knowledge specific to that language. Language classification is made by means of extremely simple acoustic models that identify the most representative sounds and the phonotactic constraints of each language (Basque, Spanish and English). If no language is reliably identified, the tag *Other* is assigned to the segment, which is discarded for further processing.

2.2.3. Speaker Classification

This task consists of classifying the speech segments in terms of speakers, which has always a positive impact on the accuracy of the speech recognizer, e.g. by applying model adaptation techniques (unsupervised clustering of similar voices, bayesian adaptation, etc.). Moreover, if speaker profiles were available beforehand, then speaker turns could be identified, which is interesting from the point of view of indexing, since users might be interested in finding the opinions of a given speaker (for instance, in meeting recordings).

2.2.4. Automatic Speech Recognition (ASR)

Speech recognition is the process of converting an acoustic signal to a sequence of words. ASR systems are invariably based on the well-known Bayes rule [11], i.e the recognizer looks for the most likely word sequence according to previously estimated acoustic models (typically, hidden Markov models) and language models (typically, n-grams). Acoustic models estimate the frequency distributions of sounds over time, and language models estimate the frequency of word sequences. Specific acoustic and language models are trained for each language. Additional difficulty is posed by spontaneous speech, which is full of mispronunciations, cut-off words, filled pauses, speech repairs, etc. Much work has to be done to address these phenomena. Our ASR module handles some of the acoustic events related to spontaneous speech by defining specific acoustic models and pseudo-words [12].

2.2.5. Natural Language Processing (NLP)

Lemmatization and morpho-syntactic analysis of each segment allow to know the lemma, number, gender and case of each word. In the case of Spanish and English, linguistic information

is extracted by means of the well-known FreeLing package [13]. In the case of Basque, the parsing process starts with the outcome of the morpho-syntactic analyzer MORFEUS [14], which deals with both simple words and multiword units. Morpho-syntactic analysis is an important step, due to the agglutinative character of Basque. Then, grammatical categories and lemmas are disambiguated, by combining linguistic and stochastic rules [15], which reduces the set of parsing tags for each word by taking into account its context.

2.3. Resource Description

The information generated by the audio processing modules is stored in an XML resource descriptor file, called *Ehiztari Resource Descriptor* (ERD). The ERD file structure, designed by means of XML Schema [16], takes into account the kind of data to be indexed and retrieved and the various modules operating on them [17]. It is based on the concept of *segment* and provides generic but powerful mechanisms to: (a) *characterize segments*, and (b) *group segments into sections*. Each segment is characterized by a set of features and consists of a sequence of words, multi-words and acoustic events, in any order. Words and multiwords may also include phonetic, lexical and morpho-syntactic information. Additionally, the ERD files include metadata describing where the audio/video resources were taken from, how they were processed and key features that allow to play their contents.

3. INFORMATION RETRIEVAL

The Information Retrieval (IR) module deals with the representation, organization and retrieval of information [18]. An IR model governs how documents and queries are represented and how the relevance of documents with regard to user queries is defined.

3.1. Spoken Document Representation

In the SDR system architecture presented in this paper, the input to the IR module is the output of the audio processing module, which includes not only the recognized word transcriptions, but also their morpho-syntactic analysis. As noted in Section 2, the audio/video resources are automatically divided into segments. Preliminary experimentation shows that many of them are too short and meaningless. So, our SDR application is defined as *passage retrieval* instead of *segment retrieval*. A passage is defined as a suitable concatenation of segments. Passage boundaries are set taking into account the number and the length of segments. The main advantage of passage retrieval is that it provides meaningful portions of spoken documents [19].

To represent spoken passages the Vector Space Model (VSM) [20] is used. In this model, each passage j is represented by means of a vector of weights $W_j = (w_{j1}, w_{j2}, \ldots, w_{jN})$, with $w_{jk} \geq 0$, and N: number of features. The weight w_{jk} tells how well the feature k characterizes the passage j. Currently, only the *lemmas* of words and multiwords are considered as features. Not all the words participate in featuring passages. Function words like articles, pronouns, prepositions, conjunctions, etc. are excluded, since they are supposed to be useless to represent document contents. To compute feature weights, various well-known functions are usually applied [21]:

– *tf*: the *term's frequency*, defined as the number of times a feature appears in a passage.
– *idf*: the *inverse document frequency*, defined so that it gives more weight to features occurring in few passages. It is the logarithm of the total number of passages divided by the number of passages containing the feature.
– *tf-idf*: it can be considered as a scalar product of *tf* and *idf*. A high *tf-idf* is reached when the feature shows a high frequency in the given passage and a low frequency in the whole collection of passages; so, the *tf-idf* weights tend to filter out common features.

3.2. The indexer

To index spoken documents, Apache Lucene [7], a high-performance full-featured text search engine library written entirely in Java, is used. The collection of feature vectors is taken as input by the indexer to create a hierarchized structure of feature references. The index structure, which includes location information for each feature, is dynamically updated each time a new XML resource descriptor is added to the SDR system.

3.3. Spoken Document Retrieval

The indexer operates offline, processing resources just once, and updating the index database each time the collection of XML resource descriptors is changed. So, it can be seen as the *back-end* of the SDR system. The information retrieval module works online, accepting and processing user queries, and searching for the matching resources. So, it can be seen as the *front-end* of the SDR system. From this point of view, the IR process begins when the user formulates a query. Two steps are carried out to retrieve the more relevant passages:

1. *Query representation*. NLP tools are applied to represent each query as a feature vector, in the same way as the passages (excluding function words as featuring items).
2. *Passage Retrieval and Ranking*. The system retrieves those passages that match the query items in the index database, and ranks them according to a predefined *matching measure*. In this work, the so called *cosine measure* (implemented by Apache Lucene) is used to estimate the similarity between a query q and a passage j:

$$S(q,j) = \frac{\sum_{k=1}^{T} w_{qk} w_{jk}}{\sqrt{\sum_{k=1}^{T} w_{qk} \sum_{k=1}^{T} w_{jk}}} \quad , \qquad (1)$$

where T is the number of content words (i.e. those used as features). Other measures have been implemented too, as the well-known *Dice and the Jaccard coefficient* [18].

4. THE USER INTERFACE

Users can interact with the SDR system by using a standard web browser. A web application based on Java Server Pages (JSP) acts as user interface, accepting queries, sending them to the search engine, and serving the list of matching items. The web application and the search engine communicate through sockets: a simple custom protocol allows sending the query and receiving the results. Results are arranged as a ranked list of references to segments (or sequence of segments) in the ERD files. The web application composes an HTML page presenting the first 10 entries of the list, and allows to request successive blocks of 10 entries.

The information regarding each entry is extracted from the corresponding ERD file and presented in HTML format. It includes the resource name, location and size, passage boundaries (time stamps), links to the original multimedia resources, transcription excerpts and a very interesting feature: a link to a new

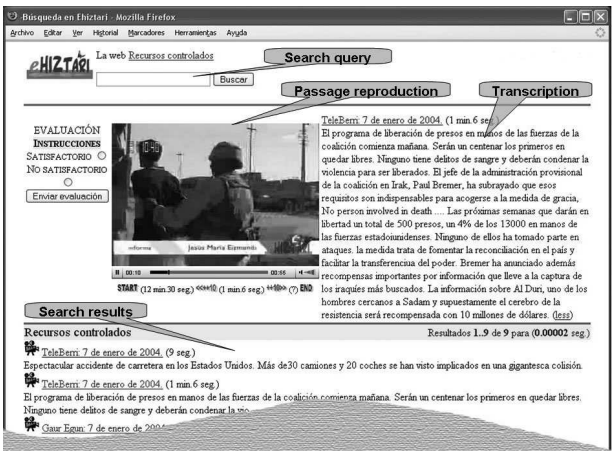


Figura 2. The user interface is based on a JSP web application. This snapshot shows search results over a controlled broadcast news database.

version of the same HTML page showing the selected passage into a customized Flash application, with its full available transcription on the right side (see Figure 2).

5. CONCLUSIONS

In this paper, the architecture of a multilingual (Basque, Spanish, English) Spoken Document Retrieval system is presented. The system consists of four main elements: (1) a crawler/downloader that fetches multimedia resources either from a local database or from internet; (2) an audio processing module, which enriches an XML resource descriptor with information extracted from the audio signal: segmentation, language, speaker, transcription and morpho-syntactic analysis; (3) an information retrieval module, consisting of an indexer (which processes the collection of resource descriptors to create an index database) and a search engine; and (4) a user interface, which allows to formulate queries, present the ranking of resources matching any given query, and access to local copies of such resources.

6. ACKNOWLEDGEMENTS

This work has been partially supported by the Government of the Basque Country, under program SAIOTEK, projects S-PE05UN32 and S-PE06UN48.

7. BIBLIOGRAFÍA

[1] J. Makhoul, F. Kubala, T. Leek, D. Liu, L. Nguyen, R. Schwartz, and A. Srivastava, "Speech and Language Technologies for Audio Indexing and Retrieval," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1338–1353, 2000.

[2] J. V. Thong, P. Moreno, B. Logan, B. Fidler, K. Maffey, and M. Moores, "SpeechBot: An Experimental Speech-Based Search Engine for Multimedia Content in the Web," *IEEE Transactions on Multimedia*, vol. 4, no. 1, pp. 88–96, 2002.

[3] J. H. Hansen, R. Huang, B. Zhou, M. Seadle, J. R. Deller, A. R. Gurijala, M. Kurimo, and P. Angkititrakul, "SpeechFind: Advances in Spoken Document Retrieval for a National Gallery of the Spoken Word," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 712–730, 2005.

[4] K. Ohtsuki, K. Bessho, Y. Matsuo, S. Matsunaga, and Y. Hayashi, "Automatic Multimedia Indexing," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 69–78, March 2006.

[5] R. Ye, Y. Yang, Z. Shan, Y. Liu, and S. Zhou, "ASEKS: A P2P Audio Search Engine Based on Keyword Spotting," in *ISM'06: Proceedings of the Eighth IEEE International Symposium on Multimedia*, San Diego, CA, USA, 2006, pp. 615–620.

[6] http://lucene.apache.org/nutch.

[7] http://lucene.apache.org.

[8] http://www.mplayerhq.hu/.

[9] http://ffmpeg.mplayerhq.hu/.

[10] M. Penagarikano and G. Bordel, "Sautrela: A Highly Modular Open Source Speech Recognition Framework," in *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, San Juan, Puerto Rico, 2005.

[11] F. Jelinek, *Statistical Methods for Speech Recognition (Second Edition)*, ser. Language, Speech and Communication Series. Cambridge, Massachusetts, USA: The MIT Press, 1999.

[12] L. J. Rodriguez, I. Torres, and A. Varona, "Evaluation of sublexical and lexical models of acoustic disfluencies for spontaneous speech recognition in Spanish," in *Proceedings of the European Conference on Speech Communications and Technology (EUROSPEECH)*, Aalborg, Denmark, 2001.

[13] http://garraf.epsevg.upc.es/freeling.

[14] I. Aduriz, E. Agirre, I. Aldezabal, I. Alegria, O. Ansa, X. Arregi, J. Arriola, X. Artola, A. D. de Ilarraza, N. Ezeiza, K. Gojenola, A. Maritxalar, M. Maritxalar, M. Oronoz, K. Sarasola, A. Soroa, R. Urizar, and M. Urkia, "A Framework for the Automatic Processing of Basque," in *Proceedings of the Workshop on Lexical Resources for Minority Languages (First International Conference on Language Resources and Evaluation)*, Granada, Spain, 1998.

[15] N. Ezeiza, I. Aduriz, I. Alegria, J. Arriola, and R. Urizar, "Combining Stochastic and Rule-Based Methods for Disambiguation in Agglutinative Languages," in *Proceedings of the 17th International Conference on Computational Linguistics*, Montreal, Quebec, Canada, 1998, pp. 380–384.

[16] http://www.w3.org/XML/Schema.

[17] http://gtts.ehu.es/Ehiztari/erd.xsd.

[18] W. Frakes and R. Baeza-Yates, *Information Retrieval*. Prentice Hall, 1992.

[19] A. Trotman and S. Geva, "Passage Retrieval and Other XML-Retrieval Tasks," in *Proceedings of the SIGIR 2006 Workshop on XML Element Retrieval Methodology*, 2006, pp. 43–50.

[20] G. Salton and M. McGill, *Introduction to Modern information Retrieval*. McGraw Hill, 1983.

[21] T. J. Mills, D. Pye, N. J. Hollinghurst, and K. R. Wood, "AT&TV: Broadcast Television and Radio Retrieval," in *Proceedings of RIAO'2000*, Paris, France, 2000, pp. 1135–1144.

AnHitz, development and integration of language, speech and visual technologies for Basque

Kutz Arrieta[1], Arantza Diaz de Ilarraza[2], Inma Hernáez[3], Urtza Iturraspe[4], Igor Leturia[5], Eva Navas[3], Kepa Sarasola[2]

[1] VICOMTech
[2] IXA Group - University of the Basque Country
[3] Aholab Group - University of the Basque Country
[4] Robotiker
[5] Elhuyar Foundation

ABSTRACT

AnHitz is a project promoted by the Basque Government to develop language technologies for the Basque language. The participants in AnHitz are research groups with very different backgrounds: text processing, speech processing and multimedia. The project aims to further develop existing language, speech and visual technologies for Basque: up to now its fruit is a set of 7 different language resources, 9 NLP tools, and 5 applications. But also, in the last year of this project we are integrating, for the first time, such resources and tools (both existing and generated in the project) into a content management application with a natural language communication interface.

This application consists of a Question Answering and a Cross Lingual Information Retrieval system on the area of Science and Technology. The interaction between the system and the user will be in Basque (the results of the CLIR module that are not in Basque will be translated through Machine Translation), using Speech Synthesis, Automatic Speech Recognition and a Visual Interface.

The various resources, technologies and tools that we are developing are already in a very advanced stage, and the implementation of the content management application to integrate them all is in work and is due to be completed by October 2008.

1. INTRODUCTION

AnHitz is a project promoted by the Basque Government in its Science and Technology Plans for 2002-2005 and 2006-2008 to develop language technologies for Basque. "Linguistic Info-engineering" has been selected as one of the 25 strategic research lines within this national program.

AnHitz is a collaborative project between five participants, each of them with expertise in a different area:

- VICOMTech (www.vicomtech.org): an applied research center working in the area of interactive computer graphics and digital multimedia.
- Elhuyar Foundation (www.elhuyar.org): a non-profit organization aimed to promote the normalization and standardization of Basque, with activities in the fields of lexicography and terminology, dictionary publishing, language planning, science and technology communication, textbooks and multimedia products and services, alongside with R&D in language technologies for Basque.
- Robotiker (www.robotiker.com): a technology center specialized in information and telecommunication technologies, part of the Tecnalia Technology Corporation.
- The IXA Group of the University of the Basque Country (ixa.si.ehu.es): specialized in the processing of written texts at different levels (morphology, syntax, semantics; corpora, MT, IE-IR…).
- The Aholab Signal Processing Laboratory of the University of the Basque Country (aholab.ehu.es): specialized in speech technologies (speech synthesis and recognition, speaker identification…).

AnHitz is a three-year project that started in 2006 and will finish in 2008. Thanks to this project seven resources, nine language tools and five applications for Basque are being developed or improved. Besides, this project will be the first in joining together the various tools for Basque in a single application that will show the potential of the integration of these technologies.

2. SOME WORDS ABOUT BASQUE AND LANGUAGE TECHNOLOGIES

Basque is an agglutinative language with a very rich morphology. There are around 700,000 Basque speakers, around 25% of the total population of the Basque Country, but they are not evenly distributed. There are six dialects, but since 1968 the Academy of the Basque Language (Euskaltzaindia) has been involved in a standardization process. At present, the

morphology is completely standardized, but the lexical standardization process is still under way.

Language technology development for Basque differs in several aspects from the development of similar technologies for widely used and standardized languages (French [1], German (verbomovil.dfki.de), Swedish (www.speech.kth.se/ctt), Norwegian [2], Dutch-Flemish [3]). This is mainly due to two reasons:

- The size of the speakers' community is small. As a result, there are not enough specialized human resources, they lack financial support, and commercial profitability is, in almost all cases, a very difficult goal to reach.
- Due to its rich inflectional morphology, Basque requires specific procedures for language analysis and generation. Thus, it is not always possible to reuse language technologies developed for other languages. This is relevant in both rule-based and corpus-based approaches. This applicability (or portability) depends largely on language similarity.

For these reasons, we believe that research and development for Basque should be (and, in the case of the members of AnHitz, usually is) approached following these guidelines:

- High standardization of resources to be useful in different lines of research, tools and applications.
- Reuse of language resources and tools.
- Incremental design and development of language resources, tools, and applications in a parallel and coordinated way in order to get the best benefit from them
- Use of open source tools.

3. RESOURCES, TOOLS AND APPLICATIONS

Some of the organizations that are part of AnHitz have been working in Natural Language Processing and Language Engineering for Basque since 1990. The most basic tools and resources (lemmatizers, POS taggers, lexical databases, speech databases, electronic dictionaries, etc.) had been developed before AnHitz, but most of them have been further improved (and are still being so) within it. And, as mentioned above, many others are being created in this project. In the following subsections we will present some of them.

3.1. Resources

- Textual resources:
  - ZT Corpusa (www.ztcorpusa.net): a 8.5-million-word tagged collection of specialized texts in Basque, which aims to be a major resource in research and development with respect to written technical Basque [1].
  - EPEC: a 300,000-word corpus tagged and disambiguated at the morphological, syntactic (syntactic functions and deep dependencies) and semantic level (word senses).
- Speech resources:
  - SpeechDat FDB1060-EU: a SpeechDat-like database for Basque that contains the recordings of 1,060 speakers of Basque obtained over the fixed telephone network.
  - SpeechDat MDB600-EU: another SpeechDat-like database for Basque that contains the recordings of 660 speakers of Basque recorded over the mobile telephone network.
  - EMODB: emotional speech database recorded by a female speaker in the six MPEG4 emotions and neutral style [2].
  - Amaia and Aitor: emotional speech database containing 702 phonetically balanced sentences repeated for the six MPEG4 emotions and neutral style, for female and male voices [3].
  - BIZKAIFON (bizkaifon.ehu.es): multimodal (speech and video) database for the Western dialects of the Basque language containing thousands of recordings of the many different variants of the western dialect of Basque [4].

3.2. Tools

- Textual tools:
  - Erauzterm: tool for automatic term extraction from Basque texts and corpora [5].
  - ElexBI: tool for the extraction of pairs of equivalent terms from Spanish-Basque translation memories [6].
  - Corpusgile and Eulia: advanced tools to create, linguistically annotate and query corpora [1].
  - CorpEus (www.corpeus.org): a web-as-corpus tool that allows querying of the internet as if it were a Basque Corpus, showing KWICs and counts of the search terms; it uses morphological query expansion and language-filtering words to optimize searching for Basque [7].
  - Dokusare: system to identify science news of similar content in a multilingual environment by using cross-lingual document similarity techniques [8].
  - Co3: a system to automatically build multilingual comparable corpora (Spanish-English-Basque) using the Internet as a source [9].
  - AzerHitz: a system to automatically extract pairs of equivalent terms from Spanish-Basque comparable corpora [10].
  - Elezkari: a cross-lingual information retrieval system focused in Basque, Spanish and English.
  - Eulibeltz: tool to create and linguistically annotate bilingual aligned corpora [11].
- Speech tools:
  - AhoT2P: a letter to allophone transcriber for standard Basque.
  - AhoTTS_Mod1: a linguistic processor for speech synthesis.

3.3. Applications

- Text applications:
  - Xuxen: spell-checker suited to the agglutinative nature of Basque that combines dictionaries with morphology, with versions for many programs and operating systems [12].
  - Elebila (www.elebila.eu): a public search engine for content in Basque that obtains a lemma-based search by means of morphological query expansion (improving recall in 89%) and results only in Basque by using language-filtering words (improving precision in 70%) [13].
  - Opentrad-Matxin (www.opentrad.org): open-source machine translation system for Spanish-Basque [14].
  - English-Basque MT: a statistical machine-translation system from English to Basque.
- Speech applications:
  - AhoTTS: a modular Text-To-Speech conversion system for Basque and Spanish [15], which has also been adapted for PDA systems [16].

4. INTEGRATION OF SYSTEMS INTO A DEMO SCENARIO

Apart from developing and/or improving the aforementioned technologies and resources, another main objective in AnHitz is to integrate as many as possible of them in a demo scenario that will show the potential of the different language technologies working together. This has never been done before with language technologies for Basque.

4.1. Features of the system

These are the features of the system we are aiming to build:

- The system will simulate an expert on Science and Technology. It will be able to answer questions or retrieve documents containing some search terms using a multilingual knowledge base.
- It will automatically translate the results to Basque if they are in English or Spanish.
- The interaction with it will be via speech. We will talk to it in Basque, and it will answer speaking in Basque too.
- The system will have a 3D human avatar that will show emotions depending on the success obtained in accomplishing the task.

This system is due to be finished by October 2008.

4.2. Modules used in the system

The system will use the following modules:

- A 3D Human Avatar expressing emotions, developed by VICOMTech.
- A Basque Text-To-Speech synthesizer (TTS), developed by Aholab.
- A Basque Automatic Speech Recognition system (ASR), integrated by Robotiker.
- A Basque Question Answering system (QA), developed by IXA, over a Science and Technology knowledge base, compiled by Elhuyar.
- A Basque-Spanish-English Cross-Lingual Information Retrieval system (CLIR), developed by Elhuyar, over a Basque-Spanish-English comparable corpus on Science and Technology, compiled by Elhuyar.
- Two Spanish-Basque and English-Basque Machine Translation systems (MT), developed by IXA.

4.3. System architecture

Fig. 1 illustrates how the different modules interact within the system and with the user.

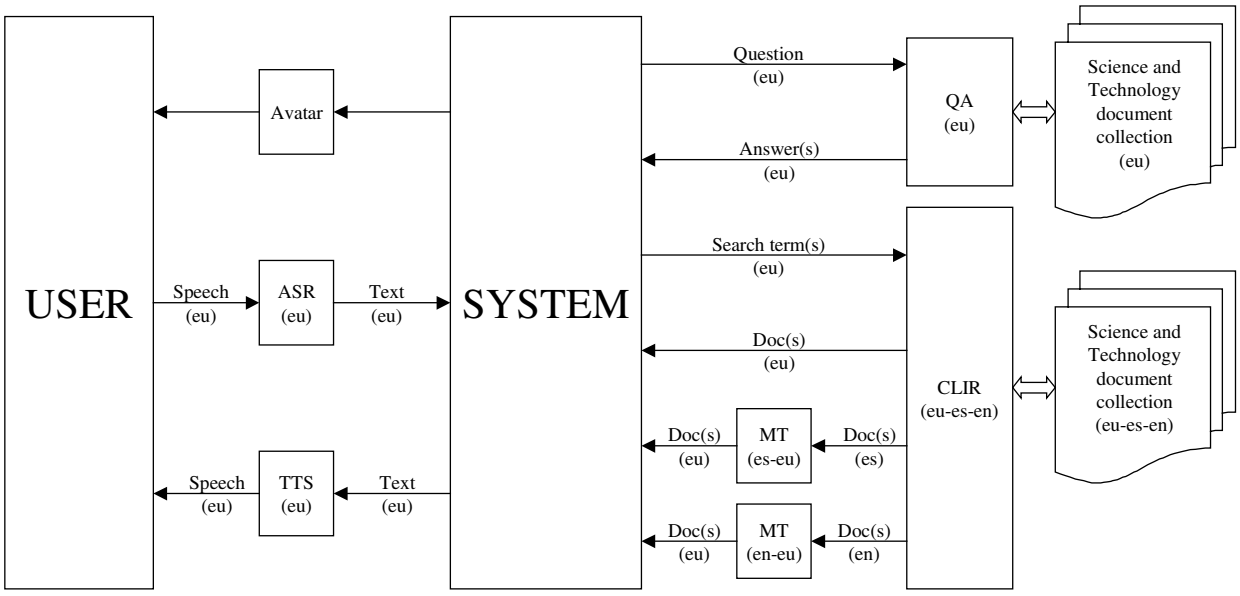


Figure 1. *Diagram showing the system architecture.*

5. CONCLUSIONS

The AnHitz project has proved to be very effective for improving the already existing language and speech resources for Basque and for creating new ones. The system that is now being developed to integrate tools and resources from different areas (an expert in Science and Technology with a human natural language interface) shows that collaboration between agents working in different areas is crucial to really exploit the potential of language technologies and build applications for the end user.

6. ACKNOWLEDGMENTS

This work has been partially funded by the Local Government of the Basque Country (AnHitz 2006: Language Technologies for Multingual Interaction in Intelligent Environments., IE06-185).

7. REFERENCES

[1] S. Chaudiron, J. Mariani, "Techno-langue: The French National Initiative for Human Language Technologies (HLT)", *Proc. of fifth international conference on Language Resources and Evaluation (LREC)*, Genoa, 2006.

[2] B. Maegaard, J. Fenstad, L. Ahrenberg, K. Kvale, K. Mühlenbock, B. Heid, "KUNSTI - Knowledge Generation for Norwegian Language", *Proc. of fifth international conference on Language Resources and Evaluation (LREC)*, Genoa, 2006.

[3] E. D'hallewey, J. Odijk, L. Teunissen, C. Cucchiarini, "The Dutch-Flemish HLT Programme STEVIN: Essential Speech and Language Technology Resources", *Proc. of fifth international conference on Language Resources and Evaluation (LREC)*, Genoa, 2006.

[4] N. Areta, A. Gurrutxaga, I. Leturia, I. Alegria, X. Artola, A. Diaz de Ilarraza, N. Ezeiza, A. Sologaistoa, "ZT Corpus: Annotation and tools for Basque corpora", *Corpus Linguistics 2007 Proceedings*, Birmingham, 2007.

[5] E. Navas, I. Hernáez, A. Castelruiz, I. Luengo, "Obtaining and Evaluating an Emotional Database for Prosody Modelling in Standard Basque", *Lecture Notes on Computer Science 3206*, 2004, pp. 393-400.

[6] I. Saratxaga, E. Navas, I. Hernáez, I. Luengo, "Designing and Recording an Emotional Speech Database for Corpus Based Synthesis in Basque", *Proc. of fifth international conference on Language Resources and Evaluation (LREC)*, Genoa, 2006, pp. 2126-2129.

[7] A. Castelruiz, J. Sánchez, X. Zalbide, E. Navas, I. Gaminde, "Description and Design of a WEB Accessible Multimedia Archive", *Proc. of 12th IEEE Mediterranean Electrotechnical Conference (MELECON)*, Dubrovnik, 2004, pp. 681-684.

[8] A. Gurrutxaga, X. Saralegi, S. Ugartetxea, P. Lizaso, I. Alegria, R. Urizar, "A XML-Based Term Extraction Tool for Basque", *LREC 2004 Proceedings*, 2004.

[9] I. Alegria, A. Gurrutxaga, X. Saralegi, S. Ugartetxea, "Elexbi, A Basic Tool For Bilingual Term Extraction From Spanish-Basque Parallel Corpora", *Euralex 2006 Proceedings*, Torino, 2006.

[10] I. Leturia, A. Gurrutxaga, I. Alegria, A. Ezeiza, "CorpEus, a 'web as corpus' tool designed for the agglutinative nature of Basque", *Web as Corpus 3 workshop Proceedings*, Louvain-la-Neuve, 2007, pp. 69-81.

[11] X. Saralegi, I. Alegria, Similitud entre documentos multilingües de carácter científico-técnico en un entorno web. *Procesamiento del Lenguaje Natural 39*, Sevilla, 2007, pp. 71-78.

[12] I. Leturia, I. San Vicente, X. Saralegi, M. Lopez de Lacalle, "Collecting Basque specialized corpora from the web: language-specific performance tweaks and improving topic precision", *Web as Corpus 4 workshop Proceedings*, Marrakech, 2008, pp. 40-46.

[13] X. Saralegi, I. San Vicente, A. Gurrutxaga, "Automatic Extraction of Bilingual Terms from Comparable Corpora in a Popular Science Domain", *Building and Using Comparable Corpora*, Marrakech, 2008, pp. 27-32.

[14] A. Díaz de Ilarraza, J. Igartua, K. Sarasola, A. Sologaistoa, A. Casillas, R. Martinez, "Spanish-Basque Parallel Corpus Structure: Linguistic Annotations and Translation Units", *Proceedings of TSD 2007 Conference*, Plzen, 2007.

[15] I. Aduriz, I. Alegria, X. Artola, N. Ezeiza, K. Sarasola, "A spelling corrector for Basque based on morphology", *Literary & Linguistic Computing, Vol. 12, No. 1*, Oxford University Press, Oxford, 1997, pp. 31-38.

[16] I. Leturia, A. Gurrutxaga, N. Areta, I. Alegria, A. Ezeiza, "EusBila, a search service designed for the agglutinative nature of Basque", *Proceedings of iNEWS'07 workshop in SIGIR*, Amsterdam, 2007, pp. 47-54.

[17] I. Alegria, A. Díaz de Ilarraza, G. Labaka, M. Lersundi, A. Mayor, K. Sarasola, "Transfer-based MT from Spanish into Basque: reusability, standardization and open source", *LNCS 4394*, Cicling, 2007, pp. 374-384.

[18] I. Hernáez, E. Navas, J.L. Murugarren, B. Etxebarria, "Description of the AhoTTS Conversion System for the Basque Language", *Proceedings of 4th ISCA Tutorial and Research Workshop on Speech Synthesis*, Edinburgh, 2001.

[19] J. Sanchez, I. Luengo, E. Navas, I. Hernáez, "Adaptation of the AhoTTS Text to Speech System to PDA Platforms", *Proceedings of the SPECOM 2006*, San Petersburg, 2006, pp 292-296.

ARQUITECTURA DISTRIBUIDA PARA UN SISTEMA DE TRADUCCIÓN DEL HABLA SOBRE LA PLATAFORMA UIMA

Marc Poch[†]*, David Cuestas*[†]*, José B. Mariño*[†]*, Francisco Méndez*[‡]*, Iñaki Sainz*[††]

[†]Centro de Investigacion TALP (UPC), *(mpoch, dcuestas, canton)@gps.tsc.upc.edu*
[‡]Grupo de Teoría de la Señal (Univ. Vigo), *fmendez@gts.tsc.uvigo.es*
[††]Grupo de Investigación Aholab (EHU), *inaki@bips.bi.ehu.es*

RESUMEN

En el presente trabajo se presenta una solución para la integración de las distintas tecnologías que intervienen en un sistema de traducción del habla. En primer lugar se introduce el software en el que se basa la solución adoptada. A continuación se describe la arquitectura diseñada en su versión local y distribuida entre sedes. Más adelante se explica como se han resuelto los problemas de intercambio de información y sincronismo entre la voz original y la sintetizada.

1. INTRODUCCIÓN

AVIVAVOZ[1] es un proyecto coordinado dirigido a la investigación en todas las tecnologías clave que intervienen en un sistema de traducción de voz (reconocimiento, traducción y síntesis de voz). El objetivo del proyecto es lograr avances en todos los componentes de un sistema de traducción de voz para alcanzar sistemas de intermediación oral entre personas en las lenguas oficiales del estado español (castellano, catalán, euskera y gallego) entre sí y entre el castellano y el inglés. El proyecto AVIVAVOZ ha sido propuesto por un consorcio formado por tres equipos investigadores: el grupo de Procesado de Voz del "Centro de Investigación de Tecnologías y Aplicaciones de la Tecnología del Lenguaje y el Habla" (TALP) de la Universidad Politécnica de Cataluña, el Grupo de Teoría de la Señal (GTS) del Departamento TSC de la Universidad de Vigo y el Grupo de Investigación Aholab (AHOLab) de la Universidad del País Vasco. Una de las actividades del proyecto está dedicada a la integración de los sistemas de reconocimiento, traducción y síntesis a fin de construir el sistema completo de traducción del habla. Uno de los intereses del proyecto es construir un sistema completo de traducción del habla utilizando tecnologías residentes en las distintas sedes, de modo que se pudiesen compartir módulos y se permitiese configurar diversas arquitecturas por cada sede según los intereses de la investigación. Por ello se ha optado por la realización de una arquitectura web para la comunicación entre los diversos subsistemas y sedes. Basándose en la experiencia del proyecto TC-STAR[2], se ha elegido la plataforma UIMA de IBM como solución práctica. Esta comunicación está organizada como sigue. En primer lugar se proporciona una breve descripción de la plataforma UIMA. A continuación se describe la arquitectura diseñada y su realización, que incluye una arquitectura local y otra distribuida, se explica el formato empleado para el flujo de información y se trata brevemente el problema del sincronismo entre el habla original y el habla producida por el sistema de traducción. El trabajo se cierra con las conclusiones.

Este trabajo ha sido subvencionado por el Gobierno Español mediante el proyecto coordinado AVIVAVOZ (TEC2006-13694-C03) y el Govern de la Generalitat de Catalunya mediante el proyecto TecnoParla.

[1] AVIVAVOZ. Tecnologías para la traducción de voz: reconocimiento, traducción estadística basada en corpus y síntesis (TEC2006-13694-C03-01/TCM) *http://www.avivavoz.es*

2. UIMA

UIMA (Unstructured Information Management Architecture) [1] es una plataforma de software extensible y escalable que permite ordenar y procesar datos no estructurados como texto, sonido, imagen, etc. El sistema fue inicialmente desarrollado por Alpha Works IBM aunque ahora es Apache Software Foundation quien se encarga del mantenimiento, de las licencias y de las nuevas versiones. UIMA es un software abierto programado en lenguaje Java que permite al desarrollador trabajar en este mismo lenguaje o en C++. La arquitectura UIMA se basa en el desarrollo de módulos independientes de proceso de datos. Estos se pueden enlazar de forma que un módulo recibe información del módulo anterior y transmite nueva información al módulo siguiente. Gracias a las herramientas de que dispone la plataforma UIMA se pueden utilizar arquitecturas distribuidas. Esto significa que cada uno de los módulos de proceso puede estar localizado en una máquina distinta y ser utilizado de forma remota. La plataforma UIMA cuenta con un conjunto de herramientas [2] para facilitar el desarrollo de módulos, componentes y arquitecturas completas. El software cuenta con distintos plugins para el entorno de programación libre Eclipse. Gracias a estos plugins se pueden editar ficheros fundamentales en UIMA de forma fácil y cómoda. A parte de los plugins, UIMA cuenta con algunas aplicaciones Java

[2] TC-STAR: Technology and Corpora for Speech to Speech Translation (IST-FP6-506738) *http://www.tc-star.org*

de ayuda que, por ejemplo, permiten ejecutar una cadena de módulos de forma simple.

2.1. Arquitectura básica

La arquitectura básica UIMA consiste en enlazar de forma consecutiva un conjunto de módulos. Estos módulos de proceso de datos reciben el nombre de "Analysis Engines" (AE) [3] y son la unidad básica de desarrollo en UIMA. Todos los módulos y componentes UIMA tienen asociados un fichero xml llamado descriptor. En él se definen distintas características del AE y se usa para hacer la llamada al modulo cuando queremos ejecutarlo. De esta forma, un AE consta de dos partes diferenciadas: su descriptor xml y el codigo Java (o C++) que el desarrollador ha implementado. Este codigo o programa recibe el nombre de Anotador. Para poder ejecutar una cadena completa hace falta un programa que sea capaz de llamar de forma ordenada a los distintos AE. Este programa está incluido en UIMA y se llama CPE (Collection Processing Engine). Como todos los componentes UIMA tiene un descriptor xml asociado que nos servirá para definir qué módulos queremos llamar y en qué orden. Para que todo el sistema funcione es fundamental el flujo de información de un AE al siguiente. UIMA utiliza una estructura de datos llamada CAS (Common Analysis structure) para pasar información de un módulo al siguiente. El CAS transporta información de UIMA (parámetros, etc) y los datos que se van a procesar llamados Sofa (Subject of Analysis).

2.2. Common analysis structure (CAS)

El CAS es una estructura de datos basada en objetos que permite la representación de objetos, propiedades y valores. Además puede albergar distintos análisis de unos mismos datos de entrada. Cada análisis corresponde a un Sofa. Para garantizar el flujo de información cada AE debe conocer la estructura del CAS. El Type System define la estructura del CAS mediante un fichero xml. De esta forma el AE puede conocer el CAS al detalle interpretando el Type System. El Type System está formado por un conjunto de Types que son representaciones de objetos y sus propiedades. La representación concreta de un objeto en el CAS recibe el nombre de Annotation.

3. ARQUITECTURA DE INTEGRACIÓN

Los sistemas de traducción del habla (voz a voz) pueden dividirse en tres tareas fundamentales: reconocimiento del habla (ASR), traducción de texto (SMT) y síntesis del habla (TTS). Estas tres tareas son desarrolladas habitualmente de forma independiente e incluso por grupos de investigación diferentes. Entonces, para poder utilizar todo el sistema, traduciendo de voz a voz, es necesario diseñar una arquitectura que sea capaz de lanzar el reconocimiento, la traducción y la síntesis de forma ordenada y de manera que las tres partes se entiendan. A tal efecto, se ha diseñado una arquitectura basada en la plataforma UIMA para poder realizar traducciones del habla. Esta arquitectura permite ejecutar traducciones de voz a voz a pesar de tener sistemas de reconocimiento, traducción y síntesis totalmente independientes. La arquitectura desarrollada permite la comunicación entre reconocedor, traductor y sintetizador aunque estos utilicen tipos de datos y lenguajes distintos. Para la implementación del sistema se han creado todos los descriptores necesarios para definir los componentes y el TypeSystem. A su vez, se han desarrollado todos los anotadores en Java. A continuación se describe la arquitectura diseñada suponiendo que todos los componentes están en una misma máquina. A esta arquitectura la llamaremos arquitectura local. En cambio, cuando algunos o todos los componentes (ASR, SMT, o TTS) están en distintas máquinas se utiliza una arquitectura distribuida.

3.1. Arquitectura local

La arquitectura local supone que tenemos los sistemas implicados (ASR, SMT, TTS) en una misma máquina que también dispone del software UIMA instalado. En caso de que alguno de los sistemas ASR, SMT o TTS estuviese implementado de forma cliente servidor lo que supondremos es que el programa cliente es el que está en la máquina local (con el UIMA y el resto de sistemas) pudiendo estar el servidor en otra máquina. Con tal de simplificar, supondremos que tanto el programa cliente como el programa servidor están en la misma máquina. Tal y como muestra la figura 1, todos los programas y ficheros se encuentran en la máquina local. El "Collection Reader" es un elemento común en todas las cadenas de ejecución UIMA y permite iniciar el sistema. Se encarga de leer el fichero "Audio file URI" (Uniform Resource Identifier) que contiene una lista de las localizaciones de los ficheros que queremos procesar. Se ha diseñado esta arquitectura asignando un AE para cada tarea de reconocimiento, traducción y síntesis. Cada AE cuenta con un anotador y un descriptor xml. El anotador ejecuta el código necesario para preparar y realizar la llamada al programa cliente ASR, SMT o TTS. Además recoge los datos de salida y los incorpora al CAS según corresponda para que el siguiente AE pueda procesarlos. El "CPE Descriptor" es el fichero xml que define toda la cadena y que se puede ejecutar utilizando un script de UIMA.

3.2. Arquitectura distribuida

Para solucionar el problema de las distintas localizaciones y las distintas máquinas para el ASR, SMT y TTS se ha diseñado una arquitectura distribuida basada en las posibilidades de la plataforma UIMA. Esta arquitectura permite ejecutar traducciones de voz a voz a pesar de tener cada componente en una máquina diferente. Para describir la arquitectura supondremos que cada sistema ASR, SMT y TTS está localizado en una máquina distinta. Además, supondremos que la máquina desde la que queremos

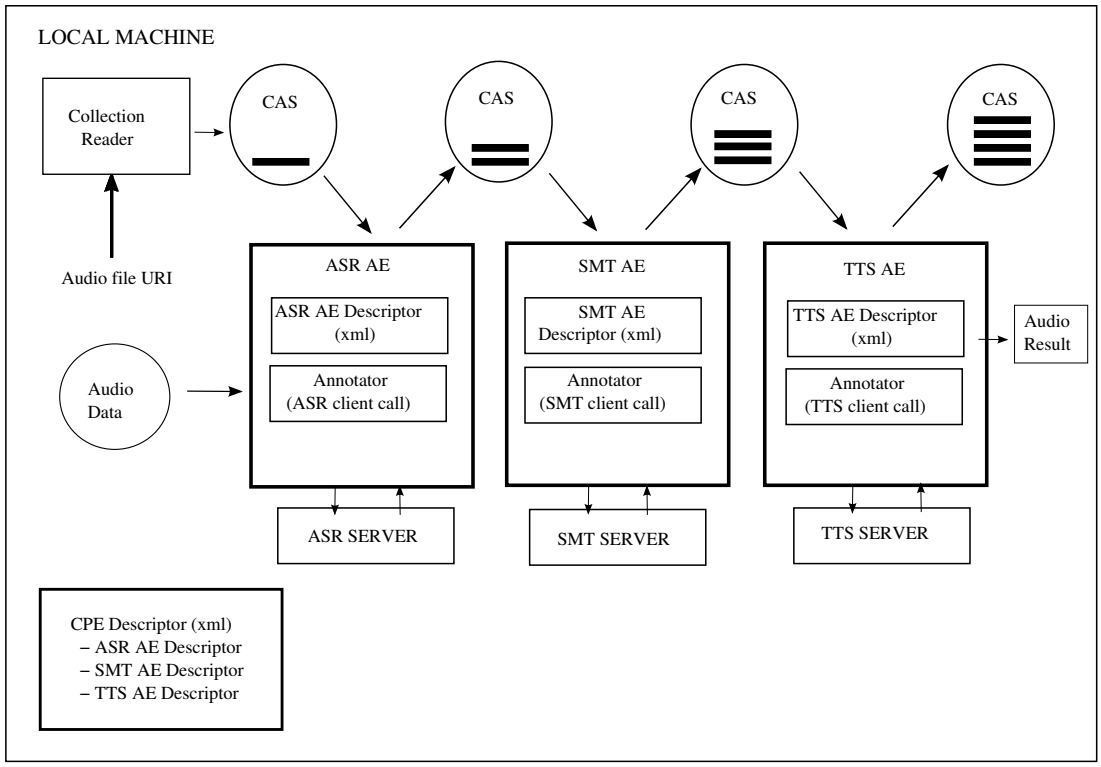


Figura 1. *Esquema de la arquitectura local.*

ejecutar la traducción es otra máquina adicional. La arquitectura distribuida requiere la utilización de un servidor central que en la suposición más general será otra máquina. Tal y como se ha descrito, y como muestra la figura 2, el caso más general de implementación de la arquitectura distribuida consta de 5 máquinas distintas. Para que el sistema funcione todas las máquinas deben tener el software UIMA funcionando. La idea de la arquitectura distribuida consiste en que se puedan ejecutar cadenas completas de forma transparente desde la máquina local que no dispone de ninguno de los servidores ASR, SMT o TTS. Para la arquitectura distribuida no es suficiente con tener los AE listos en cada máquina, hay que dar de alta cada AE como servicio en el servidor central. Éste utiliza un servidor VNS (Vinci Name Service) para gestionar una lista de los servicios. Cada servicio define de forma unívoca a cada AE y contiene información de la máquina en la que se encuentra y los puertos que utiliza. De esta forma con el nombre de un servicio disponemos de la información necesaria para acceder al AE que lo ejecuta. Gracias al sistema VNS no es necesario tener los descriptores de cada AE en la máquina local como pasaba en la versión no distribuida. En vez de eso, se utilizan unos descriptores simples (Vinci Client Descriptor) que sólo contienen el nombre del servicio al que queremos llamar. En la figura 2 aparece un componente llamado "CopyIn". Éste servicio permite copiar el fichero de entrada, situado el la carpeta "in" del servidor central, a la máquina ASR. En cambio, el servicio "CopyOut" accede a la máquina TTS para copiar el fichero de salida a la carpeta "out" del servidor central. De esta manera el usuario puede acceder al fichero de entrada y salida de la cadena completa sin tener que acceder a las máquinas que prestan los servicios.

3.3. Flujo de información

A la hora de construir plataformas de integración de software surge el problema de la comunicación entre componentes. En los apartados anteriores se explica como se ha resuelto el problema de como intercambiar información entre ASR, SMT y TTS. Una vez establecido el flujo de información hay que resolver el problemas de que datos hay que transmitir y de que manera para que todos los componentes los interpreten. Se ha decidido adoptar el estándar HTK [3] de grafos como formato de datos entre los distintos módulos. El sistema HTK cuenta con numerosas librerías y software ya desarrollado para crear e interpretar grafos. Estos grafos serán incluidos en el CAS para ser transportados al siguiente módulo. Los grafos HTK permiten representar múltiples hipótesis, incluir tiempos asociados a las hipótesis, incorporar información sobre la confianza en las hipótesis e incluir meta información.

3.4. Sincronismo

Para poder hacer una traducción de voz sincronizada con la voz original el intercambio de información entre

[3]The Hidden Markov Model Toolkit *http://htk.eng.cam.ac.uk*

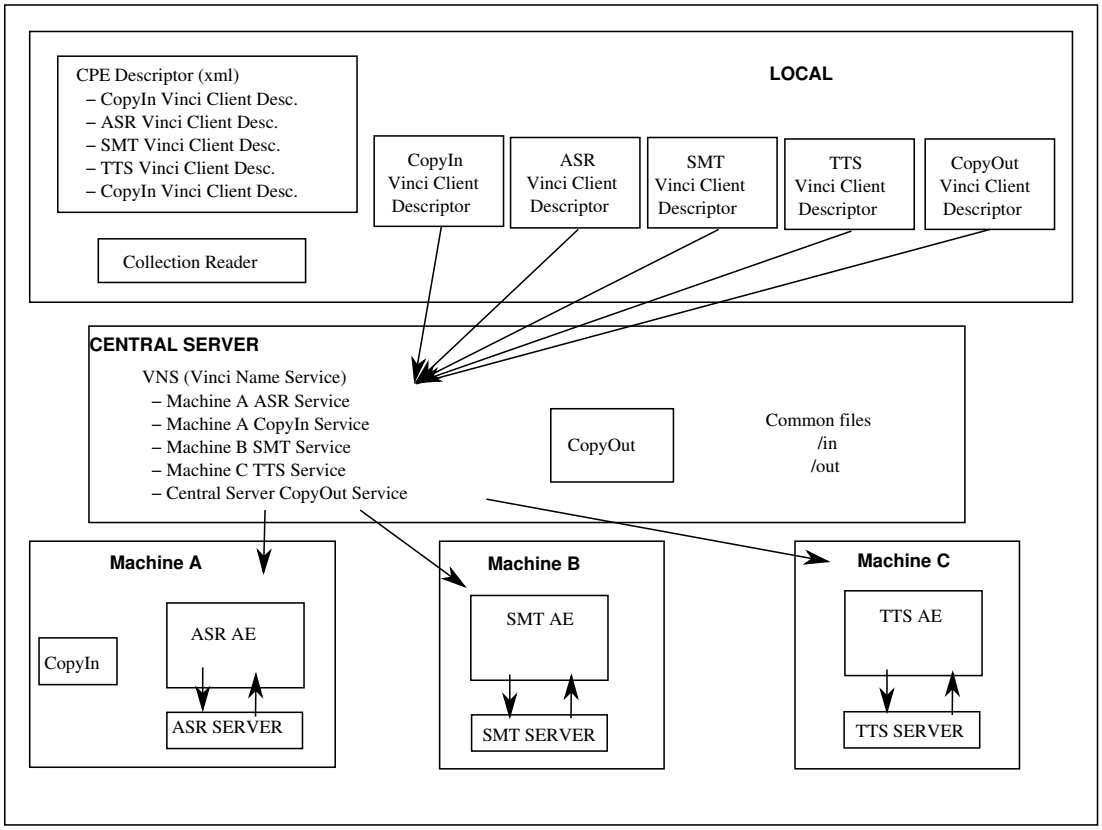


Figura 2. *Esquema de la arquitectura distribuida.*

los módulos no puede ser solo texto. El reconocedor debe ser capaz de dar información temporal asociada al texto. El traductor por su parte deberá ser capaz de establecer una relación entre el texto traducido y el texto original. Y para acabar, el sintetizador deberá ser capaz de interpretar los datos del ASR y el SMT de manera que pueda establecer una relación entre el texto traducido que debe sintetizar y los tiempos del habla original. Aparece otra vez un problema de comunicación entre programas totalmente independientes que hay que resolver. A tal efecto, la arquitectura diseñada cuenta con que el módulo de reconocimiento se encarga de incluir los datos temporales asociados a cada palabra en su grafo de salida. Por su parte, el traductor establece una relación basada en índices de palabras para relacionar una o varias palabras del texto traducido con una o varias palabras del texto original. Toda esta información será incluida en el grafo de salida del módulo SMT. Para acabar, el módulo TTS interpreta los grafos de salida del ASR y SMT y recupera la información que necesita para sincronizar la síntesis de voz.

4. CONCLUSIÓN

Las arquitecturas desarrolladas ofrecen soluciones a los principales problemas de la integración de sistemas de traducción de voz a voz independientes. Ambas arquitecturas, local y distribuida, han sido probadas con éxito. La arquitectura distribuida ha sido montada de forma que tres sedes (Universitat Politècnica de Catalunya, Universidade de Vigo y Universidad del País Vasco) prestaban alguno o todos los servicios de reconocimento, traducción y síntesis al sistema. Hay que destacar que estas arquitecturas no influyen en las prestaciones del reconocedor, del traductor y del sintetizador en cuestión pero permiten avanzar en el desarrollo de sistemas integrados de traducción de voz por su escalabilidad y su flexibilidad. Gracias a ello se puede plantear la concatenación de distintos traductores, la inclusión de nuevos módulos, etc. Para finalizar podemos decir que esta arquitectura es una valiosa herramienta para el desarrollo de las tecnologias del habla y la colaboración entre universidades.

5. BIBLIOGRAFÍA

[1] The Apache UIMA Development Community, *UIMA Overview and Setup*, The Apache Software Foundation, February 2007.

[2] The Apache UIMA Development Community, *UIMA Tools Guide and Reference*, The Apache Software Foundation, February 2007.

[3] The Apache UIMA Development Community, *UIMA Tutorial and Developers Guides*, The Apache Software Foundation, February 2007.

COMPUTER-ASSISTED HANDWRITTEN TEXT TRANSCRIPTION USING SPEECH RECOGNITION

Antonio-L. Lagarda, Vicent Alabau, Carlos-D. Martínez-Hinarejos, Alejandro-H. Toselli, Verónica Romero, José-R. Navarro and Enrique Vidal

Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
Camino de Vera, s/n, 46022 Valencia, Spain
{alagarda, valabau, cmartine, ahector, vromero, jonacer, evidal}@iti.upv.es

ABSTRACT

Handwritten Text Recognition (HTR) has gained a lot of attention during the last few years. On the one hand, there exists a large number of applications that can benefit from it. On the other hand, HTR is similar to Automatic Speech Recognition (ASR). Thus, modelling techniques can be easily adapted and most of the already available ASR tools can be used to train and evaluate HTR systems quite straightforwardly. However, HTR performance is still far from perfect and post-editing is needed. This post-editing can be made efficiently by means of Computer Assisted Transcription of Handwritten Text Images (CATTI) tools, which iteratively interact with the user to achieve the desired transcription. Typically, this interaction has been made via mouse and keyboard. In this paper, we present the development of a new CATTI system which uses speech as an additional mean of interaction. The system is build upon a generic recogniser both for speech and handwriting recognition.

1. INTRODUCTION

Handwriting recognition has become an interesting task in the last years due to its multiple applications. These applications cover from the use of mobile devices where handwritten input is used [1] to the transcription of handwritten documents, specially of ancient documents with high historic value [2].

The recent success of Handwritten Text Recognition (HTR) is based on the use of models that were previously used in other tasks such as speech recognition. Nowadays, handwritting recognition systems make use of Hidden Markov Models (HMM) to define the morphological features of the handwritten text [3]. They also use language models (usually N-grams) to define the relations between the words to be recognised.

Work supported by VIDI-UPV under project 20070315, by the EC (FEDER) and the Spanish MEC under grant TIN2006-15694-CO2-01, by the Spanish research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018), and by the i3media Cenit project (CDTI 2007-1012).

However, recognition accuracy is far to be perfect, and post-editing is usually needed. Apart from the classical post-editing techniques, based on keyboard and mouse input, a novel approach is the use of speech to validate the correctly recognised text. This approach has been previously used in other computer-assisted task, such as Computer-Assisted Translation (CAT) [4], with promising results. Moreover, the validation of the correct recognition by the user entails the possibility of using this information as feedback to improve the recognition models.

In this paper, we show the development of a Computer-Assisted Transcription of Handwritten Text Images (CATTI) [2] system which uses validation based on speech input, i.e., a multimodal system. The system is build using a generic recogniser, the iATROS toolkit [5], which allows an easy construction of multimodal applications. The iATROS toolkit can perform handwriting and speech recognition. Both types of recognitions use models of the same nature and the same search process.

The paper is organised as follows: Section 2 presents the theoretical foundations of the CATTI process and the integration of the speech input in the CATTI framework. Section 3 describes the use of our CATTI system. Section 4 specifies the architecture and relations between the different modules that have been used in the construction of our system. Section 5 presents some conclusions and future improvements for the system.

2. FRAMEWORK

This section introduces the theoretical framework of the CATTI process and how speech input can be incorporated.

2.1. Handwritten text recognition

The traditional handwritten text recognition problem can be formulated as the problem of finding the most likely word sequence, $\hat{w}$, for a given handwritten sentence image represented by a feature vector sequence x, i. e., $\hat{w} = \arg\max_w \Pr(w|x)$. Using the Bayes' rule we can decompose the probability $\Pr(w|x)$ into two probabilities,

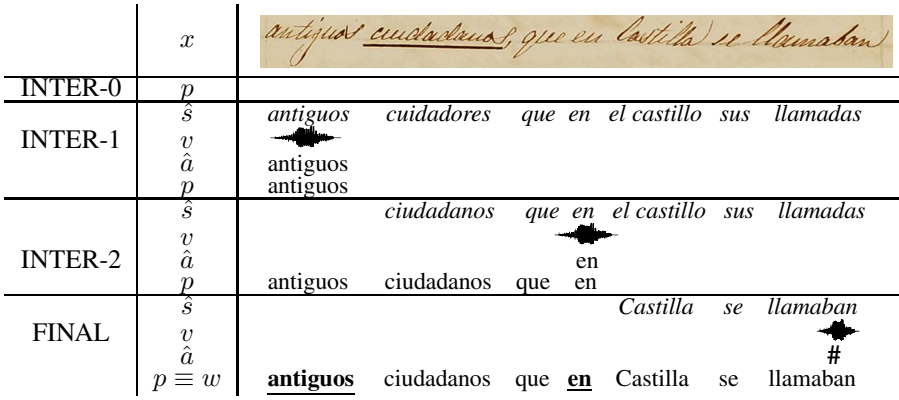


Figure 1. Example of speech interaction with a computer-assisted transcription of handwritten text images. Given a handwritten text image x, the system finds its most likely transcription $\hat{s}$. Then, the user validates its longest error-free prefix p by uttering (v) the last correct word ($\hat{a}$). Next, the new prefix p is taken by the system to suggest a new improved hypothesis $\hat{s}$. This loop is repeated until a transcription is deemed satisfactory by the user, indicated by "#" in the figure. System suggestions are printed in italics, text decoded from user speech in boldface. In the final translation w, words uttered by the user are those that appear in underlined boldface.

$\Pr(x|w)$ and $\Pr(w)$, representing morphological-lexical knowledge and syntactic knowledge, respectively:

$$\hat{w} = \arg\max_{w} \Pr(w|x) = \arg\max_{w} \Pr(x|w) \cdot \Pr(w) \quad (1)$$

$\Pr(x|w)$ is typically approximated by concatenated character models (usually hidden Markov models [6, 7]) and $\Pr(w)$ is approximated by a word language model (usually n-grams [6]).

In the CATTI framework [2, 8], in addition to the given feature sequence, x, a prefix p of the transcription (validated and/or corrected by the user) is available and the HTR should try to complete this prefix by searching for a most likely suffix $\hat{s}$ as:

$$\begin{aligned} \hat{s} &= \arg\max_{s} \Pr(s|x,p) \\ &= \arg\max_{s} \Pr(x|p,s) \cdot \Pr(s|p) \ . \end{aligned} \quad (2)$$

Equation (2) is very similar to (1), being w the concatenation of p and s. The main difference is that now p is given. Therefore, the search must be performed over all possible suffixes s following p and the language model probability $\Pr(s|p)$ must account for the words that can be written after the prefix p.

2.2. Interaction with speech

In the scenario described in the previous section, the user interacts with the system to achieve a better transcription of the original text. Typically, the user validates and/or corrects the prefix with the keyboard and the mouse. However, this interaction would be more natural and efficient if the user could interact with the speech as well. For that reason, we present an alternative way for the user to interact with the system by means of a speech recogniser in a similar fashion of what was proposed in [4].

The problem of speech recognition can be formulated in a similar way to the problem of handwritten text recognition in Equation (1). Let be v the vector of features representing the speech signal, a a sequence of words indicating the action to be performed and $p(v|a,\hat{w})$ the phonological-lexical knowledge. $p(a,\hat{w})$ is a language model of the possible actions that are understood by the system (usually a finite state machine). This model is constrained to the current iteration hypothesis and to the limited set of actions that are allowed at this iteration. As a result, the problem of obtaining the most likely word sequence of the action, $\hat{a}$, given the speech utterance v from the user and the current most likely transcription $\hat{w}$ can be written as:

$$\begin{aligned} \hat{a} &= \arg\max_{a} \Pr(a|v,\hat{w}) \\ &= \arg\max_{a} \Pr(v|a,\hat{w}) \cdot \Pr(a,\hat{w}) \ . \end{aligned} \quad (3)$$

3. SPEECH INTERACTION WITH A COMPUTER-ASSISTED TRANSCRIPTION SYSTEM

This section details the use of a CATTI system with speech interaction, illustrated in Figure 1. The described CATTI system is similar to that explained in [9]. Given a handwritten text image, the iterative process starts when the HTR module proposes a full transcription $\hat{s}$ of the feature vectors sequence x, extracted from the image. Then, the human transcriptor (namely the user of our system) checks this hypothesis until he or she finds a mistake. In this way, he or she validates a prefix p of the transcription which is error-free. In our case, this validation can be carried out by means of the keyboard or speech: in the former, by moving the cursor until the last correct word; in the latter, just by uttering that word ($\hat{a}$). In this way, the user is not only validating an error-free prefix from

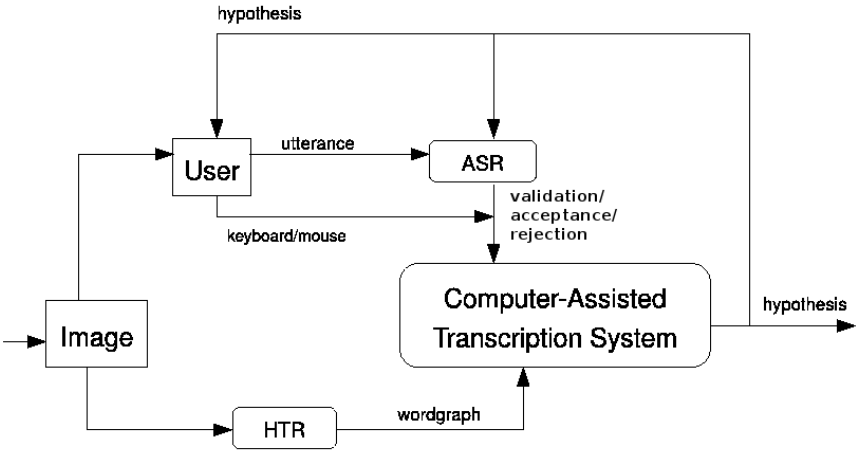


Figure 2. System architecture. From a given handwritten text image, a HTR system (which involves preprocessing, feature extraction, and recognition) produces a word graph of possible transcriptions. This word graph is taken by the Computer-Assisted Transcription System, which starts an iterative process where the user interacts with the system to achieve an error-free transcription. That interaction can be performed by means of speech, the keyboard or the mouse. If speech is employed, the ASR module is in charge of its acquisition, preprocessing and recognition. For further details on this process, see section 3.

the transcription, but also indicating to the system that the following word in the hypothesis is wrong. With that information, the system can suggest an new appropriate suffix $\hat{s}$ to continue that prefix. This cycle is repeated until a correct transcription w of x is accepted by the user by typing an acceptance key sequence or by uttering a reserved acceptance word (in our case, "accept"). There is another special situation in which the first word of the suffix is wrong. In this case, the user just needs to utter a reserved rejection word, and the system will propose a new suffix beginning from a different word.

4. SYSTEM ARCHITECTURE

Figure 2 shows a diagram of the system architecture, while Figure 3 is a screen capture of the graphical interface of the application which implements this architecture. Our system is composed of three main components: a HTR system, a CATTI system, and an Automatic Speech Recognition (ASR) system. Given a handwritten text image, the HTR system obtains a word graph with the most likely transcriptions. This word graph is employed by the CATTI system to perform the decoding process, looking for the most probable transcription taking into account the current validated prefix and applying error-correcting techniques when the current prefix is not in the word graph [10]. The CATTI module interacts with the user until the final transcription is achieved (see Section 3 for more details). That interaction can be carried out by means of speech, which is recognized by the ASR system.

Both the HTR and the ASR systems have been implemented based on the iATROS toolkit [5]. Their structure is similar: given an input (respectively, a handwritten text image and an audio signal) the iATROS toolkit is in charge of its preprocessing, the extraction of the feature vectors, and a search based on Viterbi. In both cases, models and search are of the same nature. HTR preprocesses the image to filter out noise, recover handwritten strokes from degraded images and reduce variability of text styles [11]; then, it extracts the image features, where a feature vector sequence is obtained as the representation of the handwritten text image; and finally, the iATROS recognition module obtains the most likely word sequences for the feature vectors in form of a word graph. ASR acquires an utterance from the user, which is preprocessed, and a final recognition is obtained. As seen in section 3, this speech interaction is used for validation purposes, so the language model must be built from the current suffix and the validation/acceptance/rejection reserved words. The small size of this language model ensures a nearly perfect speech recognition.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented the development of a Computer-Assisted Transcription of Handwritten Text Images system in which speech was employed to perform the user-machine interaction.

In this way, the decoding of a human transcriptor utterance is used to validate a prefix of the final transcription. This prefix is taken into account by the CATTI system to suggest a new suffix. The human transcriptor can

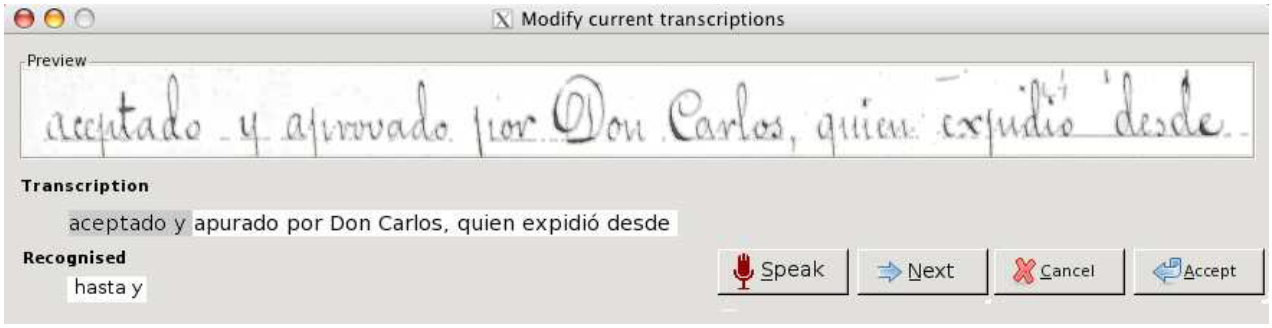


Figure 3. Interface. The original handwritten image appears in the *Preview* box. After an initial transcription is proposed by the system, the user can select a correct prefix and ask for the suffix that best completes it. In the example, once read the suggested transcription, the user decides that its longest error-free prefix is *aceptado y*. Then, he or she pushes the *Speak* button to begin to speak, and utters *hasta y* to select the last correct word. The system will take the new prefix *aceptado y* and try to complete it by suggesting a new suffix.

then accept it or partially validate it by means of speech or by typing in an iterative way until a satisfactory, correct transcription is finally produced.

The iATROS toolkit has been succesfully employed to implement the Handwritten Text Recognition and the Automatic Speech Recognition modules.

In future versions of our system, speaker adaptation techniques could be easily introduced [12], which could improve the speech recognition quality.

In addition, our system could take profit of the user amendments to improve the models quality. In this way, each new iteration, the necessary effort by the user to achieve the correct transcription would be lower, which would enhance both the ergonomics of the system and the user's productivity.

6. REFERENCES

[1] J. Hannuksela, P. Sangi, y J. Heikkila, "Motion-based handwriting recognition for mobile interaction," in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, Washington, DC, USA, 2006, pp. 397–400, IEEE Computer Society.

[2] V. Romero, A. H. Toselli, L. Rodríguez, y E. Vidal, "Computer Assisted Transcription for Ancient Text Images," in *International Conference on Image Analysis and Recognition (ICIAR 2007)*, vol. 4633 of *LNCS*, pp. 1182–1193. Springer-Verlag, Montreal (Canada), August 2007.

[3] M. Gilloux, *Hidden markov models in handwriting recognition*, vol. 126 of *NATO ASI Series*, Springer Verlag, France, 1994.

[4] E. Vidal, F. Casacuberta, L. Rodríguez, J. Civera, y C. Martínez, "Computer-assisted translation using speech recognition," *IEEE Transaction on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 941–951, 2006.

[5] M. Luján, V. Tamarit, V. Alabau, C.-D. Martínez-Hinarejos, M. Pastor, A. Sanchis, y A. Toselli, "iATROS: A speech and handwriting recognition system.," in *V Jornadas en Tecnología del Habla*, p. Accepted. Bilbao, 2008.

[6] F. Jelinek, *Statistical Methods for Speech Recognition*, MIT Press, 1998.

[7] L. Rabiner, "A Tutorial of Hidden Markov Models and Selected Application in Speech Recognition," *Proc. IEEE*, vol. 77, pp. 257–286, 1989.

[8] A. H. Toselli, V. Romero, L. Rodríguez, y E. Vidal, "Computer Assisted Transcription of Handwritten Text," in *9th International Conference on Document Analysis and Recognition (ICDAR 2007)*, pp. 944–948. IEEE Computer Society, Curitiba, Paraná (Brazil), September 2007.

[9] V. Romero, A. H. Toselli, J. Civera, y E. Vidal, "Improvements in the computer assisted transcription system of handwritten text images," in *PRIS*, 2008, pp. 103–112.

[10] S. Barrachina, O. Bender, F. Casacuberta, J. Civera, E. Cubel, S. Khadivi, A. Lagarda, H. Ney, J. Tomás, y E. Vidal, "Statistical approaches to computer-assisted translation," *Computational Linguistics*, p. In press, 2008.

[11] A. H. Toselli, A. Juan, J. González, I. Salvador, E. Vidal, F. Casacuberta, D. Keysers, y H. Ney, "Integrated handwriting recognition and interpretation using finite-state models," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 4, pp. 519–539, 2004.

[12] P. C. Woodland, "Speaker adaptation for continuous density hmms: A review.," *ITRW on Adaptation Methods for Speech Recognition*, pp. 11–19., August 29-30, 2001.

SAUTRELA: UN ENTORNO DE DESARROLLO VERSÁTIL PARA LAS TECNOLOGÍAS DEL HABLA

Mikel Peñagarikano, German Bordel, Sonia Bilbao, Maider Zamalloa, Luis Javier Rodriguez

Grupo de Trabajo en Tecnologías Software
Departmento de Electricidad y Electrónica
Universidad del País Vasco
mikel.penagarikano@ehu.es

RESUMEN

El presente artículo describe Sautrela, un paquete de software desarrollado como código abierto en Java™, altamente modular y escalable, y enfocado a las tecnologías del habla (véase http://sautrela.es). Su arquitectura neutra y modular permite no sólo generar motores de reconocimiento del habla partiendo desde cero, sino también abordar otras tareas relacionadas con las tecnologías del habla, tales como el reconocimiento y verificación de la lengua y del locutor. Los sistemas basados en Sautrela son instanciados a partir de un descriptor XML parametrizable, de tal forma que el valor de algunos de sus parámetros pueda ser establecido en tiempo de ejecución. La arquitectura de modelos por capas y la disponibilidad de módulos de entrenamiento y decodificación de propósito general sirven de base para la construcción de sistemas de complejidad arbitraria.

1. INTRODUCCIÓN

Sautrela es un entorno de desarrollo portable, altamente modular y de código abierto enfocado al procesamiento del habla. Sautrela trata de unificar todas aquellas tareas relacionadas con el reconocimiento del habla, tales como el procesamiento de señal, el modelado (entrenamiento) y la decodificación. Al estar basado en Java™, ofrece una gran capacidad de portabilidad a diversas arquitecturas hardware. Sautrela hace uso de una arquitectura de componentes basada en JavaBeans™ [1], lo que la convierte en una plataforma modular fácilmente ampliable con plugins de terceros.

Si bien en la actualidad existen paquetes de código abierto enfocados al procesamiento del habla, tales como HTK [5] y Sphinx [4], ambos carecen de la flexibilidad necesaria para abordar diferentes tareas que surgen en el ámbito de las tecnologías del habla. Así, HTK es un toolkit que ofrece una amplia gama de herramientas para diseñar y manipular modelos ocultos de Markov (HMM), pero su arquitectura resulta rígida a la hora de integrar otras tecnologías. Sphinx plantea una arquitectura más modular, pero claramente orientada a sistemas de reconocimiento del habla. Sautrela define una arquitectura neutra y modular que permite no solo generar motores de reconocimiento del habla partiendo desde cero, sino también abordar otras tareas relacionadas con las tecnologías del habla, tales como el reconocimiento y verificación de la lengua y del locutor.

El resto del artículo está organizado como sigue. La Sección 2 describe la arquitectura y características principales de Sautrela. La Sección 3 revisa algunos de los módulos de procesamiento incluidos en el paquete. Finalmente, la Sección 4 resume las principales contribuciones del presente trabajo.

2. ARQUITECTURA

La mayoría de las tareas que se abordan dentro de las tecnologías del habla pueden verse como complejos sistemas de procesamiento de la información: un reconocedor del habla no es sino un decodificador que trata de extraer un mensaje que ha sido previamente codificado en una señal sonora, y un verificador de la lengua trata de comprobar si una lengua ha sido o no utilizada en la codificación acústica de un mensaje. Este tipo de sistemas se componen normalmente de fases independientes de procesamiento (eliminación de eco y reverberación, extracción de información espectral, reducción de ruido, decorrelación, compensación de canal, modelado, etc.) cuyo encadenamiento da lugar al procesamiento global. Sautrela se basa en la abstracción de componentes de procesamiento (*Procesadores*) que al encadenarse dan lugar a sistemas complejos de procesamiento (*Engines*).

2.1. Engines y Procesadores

Una *Engine* está compuesta por un conjunto de módulos de procesamiento interconectados entre sí y que son ejecutados en hilos independientes (véase la Figura 1). Dicha interconexión se realiza mediante buffers intermedios donde datos y señales de control son almacenados

Este trabajo ha sido parcialmente financiado por el Gobierno Vasco, dentro del programa SAIOTEK, a través de los proyectos S-PE06UN48 y S-PE07UN43.

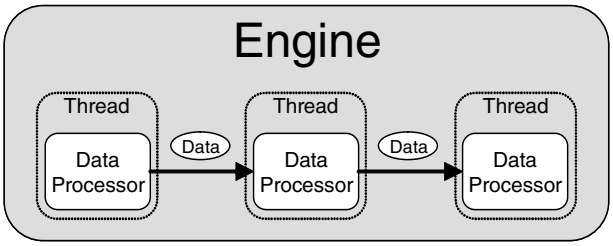


Figura 1. Una *Engine* consta de un conjunto de *Procesadores* encadenados ejecutándose en hilos independientes. Cada módulo de procesamiento representa una fase del proceso global.

temporalmente. La naturaleza multihilo se ajusta perfectamente a las arquitecturas hardware multicore actuales, y permite aprovechar los recursos del sistema. Los *Procesadores* son unidades independientes de procesamiento que implementan una interfaz que les permite ser integrados en el sistema.

Las *Engines* son instanciadas a partir de un simple descriptor XML que contiene la lista de *Procesadores* y los valores de sus parámetros. Es posible también parametrizar una *Engine*, de tal forma que los valores de algunos de sus parámetros puedan ser establecidos en el momento de la ejecución. La parametrización de engines es una técnica sencilla y elegante que permite diseñar sistemas configurables en tiempo de ejecución. La Figura 2 muestra un descriptor XML parametrizado.

2.2. Plugins

Un Plugin es un conjunto de procesadores diseñados para algún tipo de aplicación específica. Basándose en el modelo de componentes software JavaBeans™, Sautrela integra de forma natural tanto plugins propios como plugins desarrollados por terceros. En este sentido, la introspección es una tecnología clave, ya que gracias a ella es posible instanciar y configurar los procesadores contenidos en un plugin, e incluso obtener la documentación necesaria para el usuario final.

2.3. Niveles de ejecución

Sautrela define dos niveles de ejecución: usuario y desarrollador. La información mostrada ante excepciones (errores) dependerá de dicho nivel. Así, un usuario será informado únicamente del error que ha ocurrido, mientras que a un desarrollador (o usuario avanzado) le será mostrada la traza completa de la excepción ocurrida. Esta sencilla distinción de niveles de ejecución ofrece al usuario medio un entorno de ejecución amigable, permitiendo a los desarrolladores acceder a toda la información necesaria para depurar errores.

3. PLUGINS INTEGRADOS

Sautrela viene acompañado de un conjunto de plugins que implementan todas las funcionalidades necesarias para desarrollar un sistema de reconocimiento del habla. A pesar de tratarse de conjuntos diseñados con una finalidad clara, la naturaleza modular de los componentes permite su reutilización para la creación de otro tipo de sistemas, minimizando el esfuerzo requerido. A continuación se describen algunos de estos componentes.

```
<Engine name="Una sencilla Engine">
 <DataProcessor className="class.Name.of.proc1">
  <param name="paramName" value="value" />
 </DataProcessor>
 <Buffer size="3" blocking="true" />
 <DataProcessor className="className.of.proc2">
  <param name="paramName1" value="?-a [23]" />
  <param name="paramName2" value="?-b" />
  <param name="paramName3" value="value3" />
 </DataProcessor>
 <Buffer size="3" blocking="true" />
 <DataProcessor className="className.of.proc3">
  <param name="paramName1" value="value1" />
  <param name="paramName2" value="value2" />
 </DataProcessor>
</Engine>
```

Figura 2. La *Engine* del ejemplo consta de tres *Procesadores*. Los parámetros que no aparecen toman su valor por defecto. El segundo *Procesador* tiene dos parámetros cuyos valores podrán ser establecidos cuando se instancie la engine, mediante las opciones de línea de comando `-a` y `-b`. Si no se presentan dichas opciones, el primer parámetro tomará el valor `23`, y el segundo su valor por defecto.

3.1. Audio

Existe un conjunto mínimo de procesadores para la lectura, captura y reproducción de audio. La utilización de localizadores uniformes de recursos (URL) ofrece una amplia gama de posibilidades de acceso a los recursos de audio, pudiendo estos residir en un simple archivo, un conjunto de archivos comprimidos, la red, etc.

3.2. Base de datos acústica

Sautrela define una estructura de base de datos acústica que permite empaquetar todos los recursos acústicos y sus correspondientes etiquetados (locutor, transcripción ortográfica, transcripción fonética, etc.) mediante un descriptor XML. Cada recurso lleva asociado un identificador único, de tal forma que el módulo de lectura de bases de datos pueda acceder de forma automática a parte de su contenido.

3.3. Procesamiento de señal

El plugin de procesamiento de señal implementa cada una de las fases de la parametrización de la señal de voz: preénfasis, ventaneo, discriminación voz/no-voz, FFT, banco de filtros, DCT, normalización cepstral, Feature-Warping y obtención de derivadas. Además, se cuenta con herramientas de clustering y un módulo de cuantificación

vectorial. Todos estos componentes son parametrizables, y sus valores por defecto están optimizados para recursos de audio grabados en condiciones de laboratorio.

3.4. Modelado, entrenamiento y decodificación

Sautrela cuenta con un conjunto de modelos que da cobertura al modelado acústico, léxico y del lenguaje. Básicamente consta de modelos ocultos de Markov discretos y continuos, autómatas de estados finitos deterministas y no deterministas, y modelos K-explorables en sentido estricto (n-gramas). También incluye los modelos de Markov por capas (Layered Markov Models), los cuales permiten integrar en un solo autómata/modelo diversos niveles de conocimiento [2]. Los modelos por capas son una pieza clave de la arquitectura de modelos de Sautrela, ya que gracias a ellos es posible aplicar diferentes técnicas de aprendizaje y decodificación.

La arquitectura de modelos de Sautrela es la base sobre la cual se asientan dos procesadores fundamentales: el decodificador y el entrenador. Ambos pueden funcionar frente a cualquiera de los modelos previamente citados. Dichos modelos implementan una interfaz de decodificación que, en combinación con parámetros de poda, es utilizada para obtener la decodificación más probable dada una entrada de datos/audio. De esta forma, el mismo modulo sirve tanto para crear un sencillo decodificador acústico-fonético, un reconocedor de comandos, o un complejo reconocedor de habla continua, ya que la única diferencia reside en el modelo que utilicemos. De forma análoga, los modelos implementan una interfaz de entrenamiento que permite desacoplar la optimización de la interfaz (tarea realizada por el entrenador) del ajuste interno de parámetros (tarea realizada por el modelo) [3]. Un único módulo de entrenamiento se encarga de realizar las tareas de reestimación independientemente de cual sea la naturaleza interna del modelo en cuestión.

Cabe mencionar que la arquitectura de modelos de Sautrela es ampliable: todo modelo que implemente la interfaz de decodificación-entrenamiento es directamente integrable en el sistema.

3.5. Utilidades

Existe un conjunto de componentes genéricos que ofrecen ciertas utilidades, como la monitorización, volcado y recuperación de datos. El volcado y recuperación de datos resulta muy útil en situaciones en las que una misma *Engine* que contiene una sección inicial constante es ejecutada repetidamente. Por ejemplo, al entrenar modelos acústicos es posible realizar una sola vez la parametrización del conjunto de entrenamiento de la base de datos y volcar el flujo de datos resultante. Una vez volcado, ese flujo de datos puede ser usado como entrada de la fase de entrenamiento de los modelos, procedimiento que se repetirá tantas veces como sea preciso.

4. CONCLUSIONES

En este artículo se ha presentado Sautrela, un entorno de desarrollo modular, escalable y de código abierto enfocado a las tecnologías del habla. Se trata de un software de propósito general que cuenta con un conjunto de plugins diseñados para construir sistemas de reconocimiento del habla. En la actualidad, Sautrela está siendo utilizado también en las áreas de verificación de la lengua y del locutor, ya que sirve de base tecnológica para las distintas tareas abordadas por el Grupo de Trabajo en Tecnologías Software (GTTS).

5. BIBLIOGRAFÍA

[1] G. Hamilton. The JavaBeans^{MT} API specification. Technical report, Sun Microsystems, 1997.

[2] M. Penagarikano and G. Bordel. Layered Markov Models: A New Architectural Approach to Automatic Speech Recognition. In *Proceedings of the MLSP Workshop*, pages 305–314, Sao Luis, Brasil, Octover 2004.

[3] M. Penagarikano, G. Bordel, and L. J. Rodriguez. Unified training of WFSA through a generic interface. In *Proceedings of Spoken Language Technology Workshop, 2006. IEEE*, pages 122–125, December 2006.

[4] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf, and J. Woelfel. Sphinx-4: A flexible open source framework for speech recognition. Technical Report TR-2004-139, Sun Microsystems, 2004.

[5] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland. *The HTK Book, version 3.4*. Cambridge University Engineering Department, Cambridge, UK, 2006.

SERVICIO DE PÁGINAS AMARILLAS UTILIZANDO RECONOCIMIENTO DISTRIBUIDO DE VOZ

José A. González, Ángel M. Gómez, José L. Carmona y Antonio M. Peinado

Dpto. de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

RESUMEN

En este trabajo se describe un prototipo de servicio de acceso a información para dispositivos móviles basado en reconocimiento distribuido de voz (DSR). La aplicación implementa un servicio de páginas amarillas en el que se consulta información sobre establecimientos y lugares de interés en una ubicación determinada. La arquitectura de reconocimiento remoto utilizada hace uso de los estándares DSR de la ETSI, que integran soluciones frente a ruido acústico y errores de transmisión. El sistema se basa en redes de paquetes, de modo que dependiendo de la conectividad del dispositivo se autoconfigura para el uso de distintas redes IP. El sistema de reconocimiento, ubicado en el lado del servidor, se basa en el reconocedor HTK. Los modelos acústicos se han entrenado para el castellano, mientras que el modelo de lenguaje se adecúa al de un servicio de páginas amarillas. Aunque el prototipo sólo puede responder a peticiones y cuestiones en el marco del dominio escogido, el sistema es flexible y permite ser exportado a otros dominios.

1. INTRODUCCIÓN

El desarrollo e implantación de nuevos servicios sobre redes inalámbricas ha encontrado un gran obstáculo en su adaptabilidad a los terminales, ya que estos tienden a disminuir las dimensiones de los interfaces (pantalla, teclado...) para aumentar su portabilidad. Esta tendencia motiva el desarrollo de nuevos interfaces de usuario que provean de una interacción natural, ubicua y multimodal. En este contexto, la naturalidad de la voz hace que las tecnologías del habla jueguen un papel decisivo. Concretamente, el Reconocimiento Automático del Habla (RAH) habilita la aparición de servicios accedidos mediante el habla permitiendo una cómoda interacción hombre-máquina.

Existen dos posibles arquitecturas para la implementación de los servicios accedidos por voz: reconocimiento de voz empotrado (ESR, del inglés *Embedded Speech Recognition*), donde el sistema RAH se encuentra en su totalidad en el dispositivo móvil o portátil, y el reconocimiento distribuido de voz (DSR, del inglés *Distributed Speech Recognition*), basado en una arquitectura cliente-servidor. En este segundo enfoque el cliente sólo parametriza y codifica la señal de voz, mientras que el motor de reconocimiento se integra en el servidor. Frente a ESR, el esquema distribuido presenta como ventaja la reducción de los requerimientos computacionales, así como del consumo de los dispositivos en el lado del cliente, trasladando las partes de mayor complejidad computacional del reconocedor de voz al servidor. Además, permite el fácil mantenimiento y actualización del núcleo del reconocedor en el servidor [1], así como la posibilidad de añadir nuevos servicios sin modificar el cliente, como por ejemplo el reconocimiento de nuevas lenguas.

A su vez, la creación de los nuevos estándares de acceso inalámbrico a Internet abre un amplio abanico de posibilidades de prestación de servicios, debido a la convergencia de las distintas redes inalámbricas de una futura cuarta generación. Prueba de esta convergencia es la nueva aparición de teléfonos móviles celulares que incluyen conexión *bluetooth* (red de área personal), Wi-Fi (red de área local) y UMTS (*Universal Mobile Telecommunications System*). Esto permite al terminal escoger aquella tecnología que le ofrezca mejores condiciones de acceso en cada situación.

En este trabajo se ha implementado un prototipo DSR para el acceso, a través de redes IP, a un servidor de información remoto. El sistema consta de dos partes: el extractor de características, o *front-end*, incluido en un cliente del tipo PDA (*Personal Digital Assistant*) o un teléfono móvil; y el *back-end,* integrado en el servidor, que procesa la información recibida y que lleva a cabo la tarea de reconocimiento.

Como tipo de aplicación se ha escogido un servicio de páginas amarillas accedido mediante voz. En este caso las consultas se realizan en castellano, aunque también se dispone de una versión en inglés [2]. El cliente lleva a cabo consultas sobre distintos tipos de establecimiento en una cierta ubicación. El servidor a su vez se encarga de realizar una transcripción textual de la petición que es utilizada para obtener una salida multimodal, en la que se muestra un mapa de la ubicación seleccionada y un listado de los establecimientos que se adecúan a la consulta realizada. De este modo, un ejemplo de consulta podría ser: *"¿Dónde hay una cafetería en Madrid?"*, a la que el sistema respondería con la información mostrada en la fi-

Este trabajo ha sido subvencionado con fondos del proyecto MEC/-FEDER TEC2007-66600.

Figura 1. Ejemplo de salida multimodal a la consulta *"¿Dónde hay una cafetería en Madrid?"*.

gura 1. Además, en caso de que el cliente disponga de un sistema GPS (*Global Positioning System*) se utiliza la información de posicionamiento para precisar la respuesta.

Esta comunicación se organiza del siguiente modo: en la sección 2 se muestra la arquitectura general del sistema; en la sección 3 se describen las características del sistema DSR, los modelos acústicos empleados, así como el vocabulario y la gramática utilizados; el sistema de información se describe en la sección 4. Finalmente, en la sección 5 se presentan las conclusiones.

2. ARQUITECTURA DEL SISTEMA

Como se ha comentado anteriormente, en este trabajo se describe un sistema de información de páginas amarillas accedido por voz. Para implementar el sistema se ha seguido un enfoque distribuido como el que aparece en la figura 2. La aplicación consta de tres elementos principales: 1) un cliente, encargado de parametrizar la voz del usuario y realizar las consultas al sistema de información; 2) un servidor RAH, que reconoce la voz del usuario a partir de los parámetros transmitidos por el cliente y 3) un servidor que retorna información sobre la consulta efectuada.

Bajo este esquema, el terminal cliente (PDA o teléfono móvil) parametriza la señal de voz del usuario utilizando un *front-end*. Los parámetros de la voz calculados se envían al servidor de reconocimiento a través de una red de paquetes (en nuestro caso TCP/IP). Dado que se pueden producir pérdidas en la red, el servidor de reconocimiento incluye un *back-end* que aplica técnicas de mitigación para aliviar el efecto de estos errores. Además, el *back-end* procesa los parámetros recibidos añadiendo características que modelan la evolución temporal de la voz (parámetros dinámicos). Finalmente, el motor de reconocimiento decodifica el mensaje del usuario a partir de estos parámetros y envía el texto reconocido de vuelta al cliente.

En la aplicación que describimos, el usuario puede realizar consultas sobre un tipo de establecimiento en una determinada área. Para flexibilizar la interacción con el usuario, la posición en la que se efectúa la búsqueda de establecimientos se puede indicar de forma explícita mediante voz (por ejemplo diciendo el nombre de una ciudad) o de forma implícita, utilizando las coordenadas geográficas del usuario. La elección entre uno u otro método la realiza de forma automática el cliente analizando el texto reconocido. En concreto, si el usuario no ha especificado ningún lugar de búsqueda en su consulta, se utilizan las coordenadas geográficas proporcionadas por un GPS instalado en el terminal (si éste lo incluye). En el caso que no se disponga de GPS ni se haya especificado ninguna localización, la búsqueda se realiza en el lugar establecido por defecto durante la instalación del sistema.

Las consultas aceptadas por el servidor de información responden al siguiente formato: ⟨ *establecimiento, posición* ⟩. En esta consulta, *establecimiento* es el servicio buscado (por ejemplo restaurante, banco, estación de autobuses, ...) y *posición* es la localización alrededor de la cual se efectúa la búsqueda (el nombre de una ciudad o un par de coordenadas GPS). En este trabajo el servidor de información utilizado es Google Maps [3]. La información devuelta por éste consiste en un mapa con la posición de los establecimientos buscados, así como una lista con información sobre éstos.

3. SISTEMA DSR

En esta sección se describe el sistema de reconocimiento distribuido, los modelos acústicos y el modelo de lenguaje empleado.

3.1. Arquitectura Distribuida

Tal y como ilustra la figura 3, el sistema DSR está constituido por dos módulos principales: el extractor de características, o *front-end,* y el motor de reconocimiento. En nuestro caso, hemos utilizado el *front-end* avanzado (AFE, *Advanced Front-End* en inglés) propuesto por el organismo ETSI [4] y el reconocedor de voz HTK [5]. El AFE se encarga de la extracción de vectores de características adecuados para el reconocimiento de voz y de la detección de actividad de voz (VAD, del inglés *Voice Activity Detection*).

Uno de los principales problemas del RAH viene dado por el ruido acústico del entorno. En un contexto móvil, como el definido por la arquitectura de reconocimiento

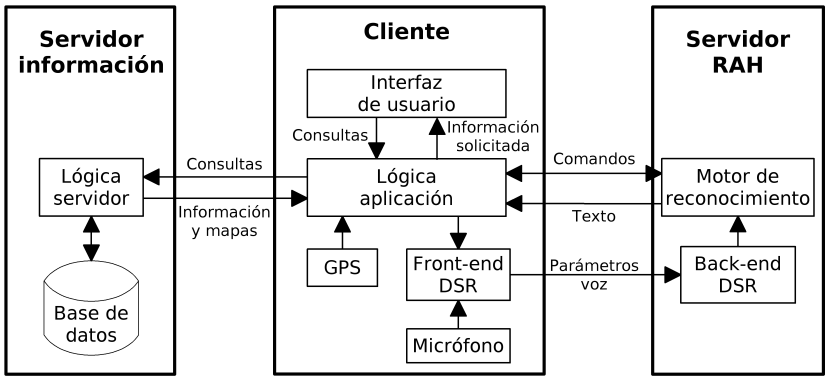


Figura 2. Esquema general de la aplicación desarrollada.

distribuida, el ruido acústico se traduce en una de las principales fuentes de degradación del rendimiento del sistema. Para combatir este problema el AFE, a diferencia de su predecesor [6], incluye un bloque de procesado de señal robusto ante ruido acústico basado en un doble filtro de Wiener [7]. Éste es el principal motivo que justifica la selección de este estándar en el diseño de nuestro prototipo.

Como resultado, el AFE obtiene un conjunto de parámetros extraídos a partir de segmentos de señal de 25 ms, distanciados 10 ms. Para cada una de estas tramas, se calcula un vector de características formado por 13 coeficientes MFCC (*Mel-Frequency Cepstral Components*) y un parámetro de energía en escala logarítmica. Finalmente, los vectores de características son codificados y encapsulados en paquetes (2 tramas por paquete).

El envío de estos parámetros se realiza utilizando el protocolo TCP. A pesar de que el retardo en el envío de información no está acotado, este protocolo permite una implementación más sencilla y soslaya la pérdida de paquetes al precio de introducir un mayor retardo en la comunicación.

En el lado servidor los paquetes recibidos son procesados por el *back-end* correspondiente al AFE. Este módulo, en primer lugar, se encarga de detectar y mitigar los posibles errores de transmisión. Posteriormente, a partir de la secuencia de vectores de características reconstruidos son calculadas las correspondientes componentes dinámicas, conformando vectores de características de 39 componentes (13 estáticas, 13 de velocidad y 13 de aceleración). Adicionalmente, se aplica CMN (*Cepstral Mean Normalization*) [7] como técnica básica de compensación de características. Finalmente, la secuencia compensada de parámetros es introducida como entrada al motor de reconocimiento HTK. El reconocedor presenta su resultado tras el final de la locución, marcada por el usuario. Este resultado es devuelto al cliente.

Tanto en el cliente como en el servidor se incluyen dos módulos de control que se encargan de dar soporte lógico al intercambio de comandos. Entre este tipo de instrucciones se encuentran el inicio y final de reconocimiento, así como distintos tipos de comandos para la correcta configuración del sistema.

El cliente DSR ha sido evaluado sobre una PDA HP IPAQ hw6915, que dispone de un microprocesador Intel PXA 270 416 MHz y 64 MB de memoria RAM. Este dispositivo permite el muestreo de la señal de voz a 8 kHz y es capaz de realizar el procesado de voz del AFE en tiempo real.

3.2. Modelo acústico de la voz

El modelado acústico de la voz se realiza a nivel de subpalabra con dependencia del contexto. En concreto, se dispone de un conjunto de modelos de trifonemas (sobre una base inicial de 32 alófonos) dependientes del contexto tanto a nivel interno de palabra, como entre palabras. Cada trifonema se modela mediante un modelo oculto de Markov (HMM, *Hidden Markov Model* en inglés) [7] continuo con una topología de izquierda a derecha, 3 estados y 5 gausianas por estado. Durante el entrenamiento de los HMMs, se lleva a cabo una agrupación de los estados similares mediante árboles de decisión binarios. Esto ayuda a reducir el número de parámetros que necesitan entrenarse además de mejorar este proceso cuando se dispone de una cantidad de datos limitada.

Para entrenar los modelos acústicos se ha utilizado la base de datos de voz ATLAS Spanish Microphone Database (MICROAES) [8]. Esta base de datos consta de grabaciones de 300 locutores realizadas con micrófonos de diferentes calidades, colocados a distintas distancias del locutor. Además, se hacen consideraciones respecto a dialectos del español (se recogen 5 dialectos diferentes), así como diferencias en cuanto a género y edad. En total la base de datos contiene 30 horas de voz.

3.3. Modelo del lenguaje

Por simplicidad, en la versión que describimos el modelo del lenguaje del sistema de reconocimiento viene dado por una gramática regular. Ésta modela las consultas aceptadas por el sistema de información mediante de reglas de producción de la forma,

$CONSULTA : [En $LUGAR] $TC [un | una] $EST |

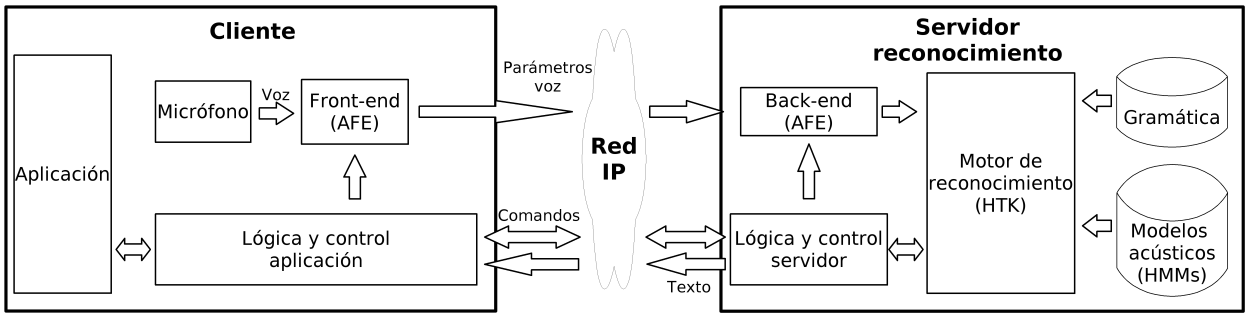


Figura 3. Arquitectura DSR.

```
$TC [un|una] $EST [en $LUGAR]
```

donde los corchetes indican contenido opcional, la barra vertical indica distintas alternativas, $LUGAR es el nombre de una población o ciudad (por ejemplo Bilbao, Granada, etc), $EST es el establecimiento buscado (por ejemplo caja de ahorros, gasolinera, etc) y $TC es una locución del tipo *dónde hay*, *busca*, *enséñame*, etc. El vocabulario resultante contiene unas 160 palabras, entre las cuales se incluyen los nombres de todas las capitales de provincia de España y gran parte de los establecimientos más utilizados en el día a día. La perplejidad de la gramática obtenida es 26,63.

4. SISTEMA DE INFORMACIÓN

El servidor de información se encarga de dar respuesta a las consultas formuladas por el usuario en el dominio de uso del sistema. En la fase de diseño del prototipo descrito, se planteó la necesidad de incluir distintos tipos de información en la respuesta a la consulta realizada. En particular, se consideró oportuno el ofrecer información gráfica en forma de mapas sobre la posición de los establecimientos buscados, así como otros datos relevantes del tipo nombre del establecimiento, dirección, teléfono, etc. Dada la dificultad del diseño y mantenimiento de un servidor de información con estas características, por simplicidad en este trabajo se utiliza como sistema de información externo el proporcionado por Google mediante la herramienta Google Maps [3].

Google Maps es un servicio de aplicaciones de mapas vía Web ofrecido por Google. De entre las características más relevantes de éste encontramos la visualización de mapas e imágenes de satélite del mundo entero, el cálculo de rutas entre distintas ubicaciones, la búsqueda de información sobre lugares de interés (e información sobre negocios) y la posibilidad de incorporar información sobre GPS a las consultas realizadas. Para los propósitos de la aplicación descrita, hemos utilizado las capacidades de este servicio para proporcionar información sobre empresas y lugares de interés. En concreto, la aplicación cliente efectúa consultas a Google Maps utilizando peticiones con el método GET del protocolo HTTP [9]. Cada consulta, por tanto, se traduce en una URL donde se modifica el valor de dos parámetros: el establecimiento buscado y el lugar donde se busca.

5. CONCLUSIONES

En este trabajo hemos presentado un servicio remoto de páginas amarillas accedido mediante voz. El sistema hace uso de la tecnología DSR, adoptando una arquitectura distribuida, en la que el reconocedor de voz y el sistema de información se encuentran ubicados en diferentes servidores. El prototipo desarrollado permite llevar a cabo consultas desde una PDA, obteniendo una respuesta multimodal con la información solicitada. El desarrollo de este prototipo demuestra la madurez de las tecnologías empleadas, así como su utilización para el desarrollo de aplicaciones comerciales.

6. BIBLIOGRAFÍA

[1] Z.-H. Tan, P. Dalsgaard, y B. Lindberg, "Automatic speech recognition over error-prone wireless networks," *Speech Communications*, vol. 47, pp. 220–242, 2005.

[2] *AVIOS Student Contest 2008, http://avios.org/contest2008/individual.htm*.

[3] *Google Maps, http://maps.google.es*.

[4] *ETSI Standard ES 202 212. Distributed speech recognition; extended advanced front-end feature extraction algorithm; compression algorithm, back-end speech reconstruction algorithm*, November 2003.

[5] S. Young, G. Evermann, y T. Hain, *The HTK book*, Cambridge Universtiy Engineering Department, 2007.

[6] *ETSI ES 201 108 - Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms*, 2000.

[7] Antonio M. Peinado y Jose C. Segura, *Speech Recognition Over Digital Channels: Robustness and Standards*, Wiley, 2006.

[8] European Language Resources Association (ELRA), *The ATLAS Spanish Microphone Database (MICROAES), http://www.elda.org/catalogue/en/speech/S0165.html*.

[9] The W3C consortium, *HTTP - Hypertext Transfer Protocol, http://www.w3.org/Protocols*.

SISTEMA DE RECONOCIMIENTO AUTOMÁTICO DEL HABLA DISTRIBUIDO APLICADO A ENTORNOS LOGÍSTICOS

José Enrique García, Alfonso Ortega, Antonio Miguel y Eduardo Lleida.

Grupo de Tecnologías de las Comunicaciones (GTC)
I3A, Universidad de Zaragoza
{jegarlai,ortega,amiguel,lleida}@unizar.es

RESUMEN

Los sistemas de reconocimiento automático del habla (RAH) pueden llegar a ser muy útiles aplicados a procesos productivos en el sector industrial como los desarrollados en entornos logísticos. Tareas como el etiquetado de paquetes o la anotación de matrículas de transportista se pueden llevar a cabo usando únicamente la voz con la consiguiente ventaja de disponer de las manos libres para la ejecución de otras tareas. En este artículo se presenta un sistema de control por voz aplicado a la logística sobre un dispositivo móvil, a partir de una arquitectura distribuida cliente-servidor donde un PC convencional recibe los parámetros acústicos enviados por el dispositivo móvil, en este caso una PDA, realiza la decodificación acústica, y procede a la actuación. Además generar una respuesta oral a partir de un sintetizador de voz enviándola a la PDA. Además de esquemas de implementación del sistema, se ofrecen unos estudios de prestaciones que caracterizan el número de operarios que pueden actuar simultáneamente con una calidad de servicio adecuada.

1. INTRODUCCIÓN

Este artículo presenta un sistema de control por voz sobre dispositivos móviles aplicado a tareas de picking y transporte, basado en una arquitectura cliente-servidor y capaz de extenderse a cualquier otro tipo de tareas productivas dentro del sector industrial.

En la tarea de recogida de mercancías en almacén (picking) el operario logístico debe realizar la preparación de pedidos de acuerdo con un plan prefijado. De esta manera las instrucciones deben ser seguidas meticulosamente, verificando el cumplimiento de las mismas a la vez que se manipulan los bultos. La libertad de movimientos que un sistema de gestión con interfaz oral provee, permitiría al operario la recepción de órdenes y la verificación de las mismas sin tener que utilizar sus manos para la manipulación de un dispositivo de gestión de tareas, quedando éstas disponibles para la extracción, preparación y transporte de unidades y lotes.

Este trabajo ha sido financiado a través de la colaboración con la empresa Alerce Informática S.A. y el proyecto de investigación TIN2005-08660-C04

En cuanto a la recepción de mercancías en muelle, los operarios del almacén realizan una serie de acciones de desplazamiento y etiquetado de artículos con el objetivo de llevar a cabo una exhaustiva identificación de la mercancía recibida. Un etiquetado haciendo uso del RAH podría ayudar a hacer el proceso productivo más eficiente ya que las manos del operario quedarían libres para hacer el desplazamiento de los artículos de forma simultánea.

Dado que los operarios que hacen uso del RAH necesitan movilidad, para desplazar cajas, para manejar maquinaria específica, para controlar los camiones que llegan o salen del muelle, etc., es conveniente que el sistema de reconocimiento esté incorporado sobre un dispositivo móvil o PDA, en lugar de hacerlo sobre un emplazamiento fijo en el almacén al que tendrían que moverse los operarios cada vez que necesitasen anotar algún evento.

Los sistemas de RAH embebidos en dispositivos móviles suelen ofrecer problemas de funcionamiento en tiempo real para tareas medianamente complejas debido a su limitada capacidad de cálculo. La tarea sólo será mejor asistida a través de una interfaz oral si presenta ventajas en cuanto a su eficiencia con respecto al empleo de una interfaz visual-táctil. El tiempo consumido en cada preparación de pedido o en cada etiquetaje es crucial y siempre deberá ser menor a través del uso de la interfaz oral, en cualquier otro caso, su uso no será considerado por motivos de productividad. De ahí que se suelan emplear arquitecturas distribuidas cliente-servidor, donde el dispositivo móvil actúa como cliente realizando el acondicionamiento y extracción de vectores de parámetros acústicos, mientras que un ordenador convencional actúa como servidor, realizando la decodificación acústica que normalmente es el proceso más costoso computacionalmente en el reconocimiento del habla. Haciendo uso de la interfaz inalámbrica (de la que suelen disponer la gran parte de PDAs), cada operario es capaz de lanzar distintos programas de control por voz creando una conexión con un ordenador que actúa como el servidor del sistema. El ordenador personal tiene la misión de realizar la decodificación acústica de las tramas enviadas por la PDA, además de generar las respuestas orales y las

actuaciones correspondientes en función de los comandos de voz reconocidos.

La precisión del sistema de reconocimiento automático del habla también es un aspecto crucial. Una elevada tasa de error hace que el usuario deba corregir continuamente las acciones del sistema con la consiguiente pérdida de eficiencia.

El artículo se organiza de la siguiente manera: En la Sección 2 se muestra la problemática y las tareas asociadas al control por voz en el ámbito de la logística. Una descripción de la arquitectura del sistema de control por voz distribuido cliente-servidor se presenta en la Sección 3. En la Sección 4 se presentan dos estudios experimentales que caracterizan las prestaciones del sistema, de forma que es posible dimensionar el número de servidores y redes inalámbricas necesarios dependiendo del número de operarios que se prevea puedan estar actuando simultáneamente en el almacén. Finalmente, en la Sección 5 se encuentran las conclusiones.

2. INCORPORACIÓN DEL 'RAH' A TAREAS LOGÍSTICAS

El sistema que se presenta en este artículo está destinado a su utilización en tareas de picking (acciones llevadas a cabo por un operario en un almacén) y de transporte, más concretamente al llenado de formularios y albaranes a la llegada o salida de camiones de un muelle.

Dado que en los ambientes a los que va destinado el control por voz se encuentra presente una gran cantidad de maquinaria pesada (carretillas, carruseles, robots industriales,…), otros trabajadores, artículos que se caen o se golpean, etc. se puede hablar de un entorno acústico no controlado, con ruido de magnitud considerable proveniente de varias fuentes y no estacionario. Sin embargo, el uso de un micrófono situado en las proximidades de la boca del locutor (close-talk) puede, de forma considerable ayudar a reducir las impurezas de la señal de audio capturada.

Todos los procesos de llenado de formularios por voz siguen el mismo protocolo, consistente en un diálogo guiado en el que el dispositivo móvil pregunta un campo, el operario lo introduce con la voz y el dispositivo móvil confirma lo que ha reconocido, pasando al siguiente campo del formulario. Al producirse la confirmación de lo que el dispositivo móvil ha reconocido, el operario dispone de la posibilidad de repetir la entrada por voz en el caso de que se hubiese producido un error.

Cada formulario dispone de un número variable de campos configurable de acuerdo con la tarea a desarrollar. Algunas de las tareas de reconocimiento que se llevaron a cabo para los campos fueron las siguientes: reconocimiento de dígitos, direcciones, nombres de empresas, localidades, códigos postales, dimensiones, y pesos. Todas las tareas suponían un reconocimiento del habla de pequeño-mediano vocabulario.

3. ARQUITECTURA DISTRIBUIDA CLIENTE-SERVIDOR

La motivación de emplear un sistema de reconocimiento de voz distribuido reside en la baja capacidad de cálculo de la que hoy en día disponen los dispositivos móviles. Este hecho, sumado a la carencia de éstos de unidad de punto flotante, conlleva la dificultad añadida de tener que programar los algoritmos de reconocimiento como operaciones en coma fija para que éstos sean capaces de actuar en tiempo real. Por ello, los sistemas de RAH sobre dispositivos móviles suelen plantearse de forma distribuida, donde el dispositivo móvil realiza las tareas de acondicionamiento, procesado y extracción de vectores de observación acústicos para enviarlos a un PC que realiza el algoritmo de reconocimiento.

Los vectores de observación acústicos empleados en el sistema son los conocidos Mel Frequency Cepstral Coefficients (MFCC), definidos en el estándar del ETSI ES 201 108 V1.1.2 (2000-04) [1]. Estos vectores se cuantifican en una última etapa de procesado en el dispositivo móvil, de forma que el ancho de banda de envío se reduce a la tasa de 4.4 Kbps. Ya que el canal sobre el que se envían se corresponde con un canal seguro (TCP) sobre la interfaz inalámbrica, se eliminaron las cabeceras empleadas en el ETSI DSR para mitigación de errores, con lo que se reduce ligeramente el ancho de banda de envío.

El PC que actúa como servidor recibe los parámetros cuantificados, y procede a la de-cuantificación para introducir los vectores acústicos en el motor de reconocimiento del grupo de investigación basado en HMM, el cual hace uso de unidades acústicas contextuales, donde cada unidad es representada mediante un estado y modelada a partir de una GMM de 16 componentes. Una vez el PC ha decidido que se ha reconocido algún comando de voz, genera la respuesta oral que es sintetizada y codificada mediante la batería de codecs libre S*peex* [2] la cual hace uso de un codificador de voz perceptual CELP. Finalmente el bitstream generado es enviado al dispositivo móvil el cual procede a la decodificación y reproducción de dicha respuesta por sus altavoces. El codec empleado era capaz de conseguir un ancho de banda de subida para la respuesta oral de aproximadamente 6.2 kbps con una calidad similar a una codificación PCM con frecuencia de muestreo 16KHz y 16 bits por muestra.

El diagrama de bloques del sistema se puede ver en la Figura 1. El proceso que aparece en la Figura 1 se repite, mientras el operario permanece conectado al sistema, cada vez que aparece un nuevo comando oral reconocido. Se puede ver que tanto el ancho de banda de envío de tramas de voz para reconocimiento (4.4 kbps) como en ancho de banda de envío de respuestas

orales codificadas son muy reducidos (6.2 kbps), por lo que una red WIFI puede dar cabida a un número muy elevado de operarios simultáneamente, en cuanto a prestaciones de ancho de banda se refiere. En el apartado 3 de este artículo se presentan algunas estimaciones para el dimensionado de la red inalámbrica.

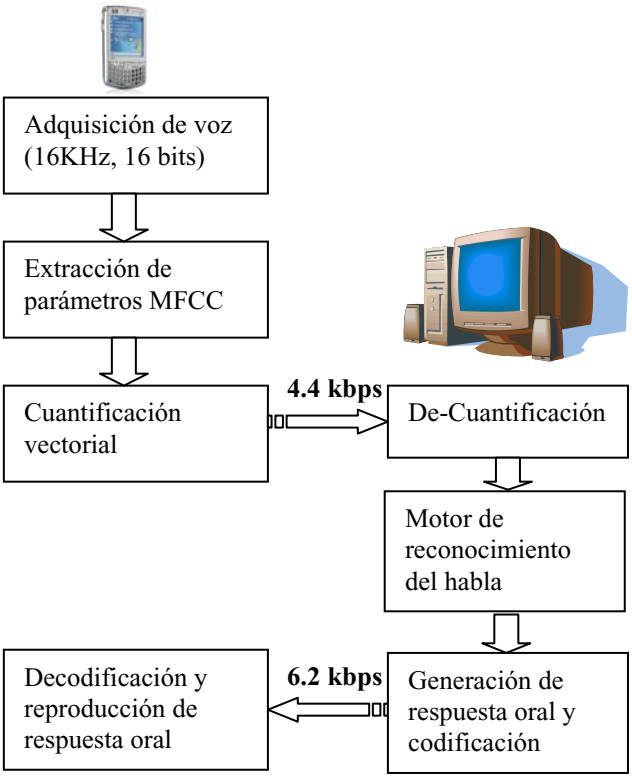


Figura 1. *Diagrama de bloques del sistema de control por voz distribuido.*

4. ESTUDIOS DE PRESTACIONES PARA DIMENSIONADO DEL SISTEMA

En cada PC que aparece como servidor es posible que se lancen tantos procesos de reconocimiento simultáneos como operarios conectados hay en el sistema. En principio, se podrían conectar un número muy elevado de operarios que disponen de su dispositivo móvil con un único PC, pero es posible que las prestaciones del sistema se viesen degradadas notablemente cuando el número de operarios conectados fuese demasiado grande. Cuando se hace referencia al término prestaciones, realmente podríamos evaluarlas en función del retardo en conocer la respuesta oral de los comandos de voz. Si este retardo es muy grande, el diálogo se ralentizaría de tal forma que sería imposible realizar un proceso de llenado de formularios o el etiquetado sería más rápido si se hiciese manualmente.

Por ello, el objetivo es realizar el dimensionado del sistema de forma que se coloquen los servidores necesarios para que todos los operarios, o una gran parte de ellos, obtengan respuestas orales a los comandos con retardos aceptables.

Los factores que pueden influir en el retardo de respuesta ofrecida por el servidor de reconocimiento son 2: El tiempo de procesado del ordenador que actúa como servidor y el ancho de banda de ocupación de la red inalámbrica.

Los sistemas de comunicaciones basados en WIFI, correspondientes a la familia de estándares 802.11b y 802.11g los cuales están implementados en la interfaz inalámbrica de la mayor parte de las PDAs, tienen velocidades de transmisión de 11 Mbps y 54 Mbps, respectivamente. Haciendo un cálculo simple, se puede ver como suponiendo el caso más desfavorable (de ocurrencia muy remota) en el que el ancho de banda total consumido por un cliente fuese 10.6 kbps (la suma del ancho de banda de envío de tramas de voz más el ancho de banda de recepción de respuestas orales) durante todo el tiempo, el número de clientes soportados para un funcionamiento en tiempo real sin retardos sería 1037 y 5094 clientes respectivamente, por red inalámbrica.

De ahí se puede observar cómo el factor más problemático que puede degradar las prestaciones es la capacidad de un servidor para procesar peticiones de clientes, y que salvo en entornos en los que el número de operarios fuese excesivamente grande, sería suficiente con un único punto de acceso.

4.1. Retardo debido al tiempo de procesado del servidor

El retardo de la respuesta oral ofrecida por el servidor se ve influenciado por el tiempo que le cuesta a éste procesar las peticiones de los clientes. Este tiempo de procesado depende principalmente de dos factores: el número de clientes conectados simultáneamente y la perplejidad de las tareas de reconocimiento con las que se está tratando en cada cliente.

Se fijaron dos experimentos, con tareas de RAH típicas del sistema de control logístico por voz. El ordenador elegido como servidor disponía de un procesador Intel Pentium IV a 3.4 GHz, bajo el sistema operativo Microsoft Windows XP. En ambos experimentos se procedió a desactivar el conversor de texto a voz, de forma que éste no influyese en los resultados ya que lo que realmente se quería evaluar eran las prestaciones del motor de reconocimiento.

El primer experimento consistió en lanzar simultáneamente varios procesos de reconocimiento en el PC, con las tareas de reconocimiento de dígitos primero, y reconocimiento del nombre de 40 calles después, midiendo el tiempo de ejecución para completar un número de procesos paralelos determinado, y obteniendo su cociente frente al tiempo real. En la Figura 2 se ve una gráfica que muestra los resultados para la tarea de reconocimiento de dígitos.

Se puede observar en la Figura 2 cómo a partir de 13 procesos paralelos el PC que actúa como servidor no sería capaz de llevar a cabo las peticiones en tiempo real, por lo que induciría un retardo en el sistema de control por voz. En la tarea de reconocimiento de calles, el número de procesos paralelos en los que se observó el corte fue 11, debido a la mayor perplejidad de la tarea. Estos valores obtenidos en los experimentos nos pueden dar una aproximación teórica de cuándo comenzarán a existir retardos no aceptables en las peticiones de los clientes para un servidor con unas características determinadas.

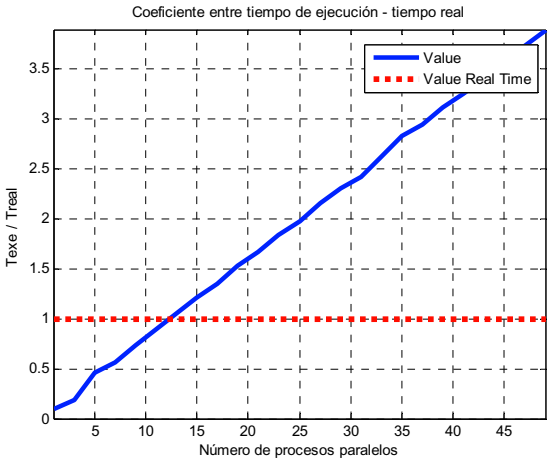


Figura 2. *Curva con el cociente Tiempo ejecución/ Tiempo real en función del número de procesos paralelos para la tarea de reconocimiento de dígitos.*

El segundo experimento llevado a cabo fue realizar medidas de retardos, a partir de una simulación de diálogo llevada a cabo lanzando hasta 17 clientes simultáneos desde 3 ordenadores personales, que se conectaban todos con el mismo PC servidor presentado anteriormente. La conexión se realizó por una red LAN de 100 Mbps. de manera que el retardo de la comunicación tuvo una influencia mínima en la medida de las prestaciones. Las tareas que se trataron fueron tanto el reconocimiento de dígitos como el reconocimiento de calles para el mismo experimento. Cada uno de los clientes que se iba lanzando realizaba medidas de retardo, sabiendo con cuántos clientes se estaba accediendo simultáneamente al servidor. A partir de estas medidas de retardo, se obtuvieron histogramas de retardo en función del número de clientes conectados al servidor de forma simultánea, obteniendo los resultados presentados en la Figura 3.

Se puede observar cómo a partir de unos 25 procesos paralelos, la distribución estadística de los retardos comienza a tener mayor varianza, a pesar de subir ligeramente en media (la media está representada en color blanco sobre la gráfica). Es importante anotar que el valor de retardo mínimo que aparece (algo menor de 1 segundo) se corresponde con la ventana temporal de decisión del detector de actividad de voz para decidir si un comando ha sido reconocido o no.

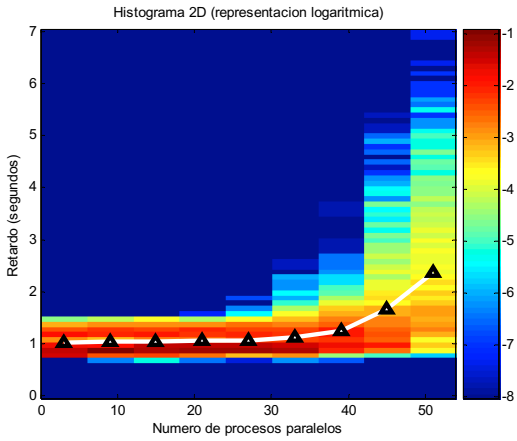


Figura 3. *Log-histograma de retardos de respuesta en función del número de clientes conectados simultáneamente al sistema. Superpuesto aparece representada la curva con el valor medio de dicho retardo.*

5. CONCLUSIONES

En este artículo se ha presentado un sistema de control por voz para entornos logísticos mediante una arquitectura distribuida cliente-servidor que puede llegar a ser muy útil para liberar las manos de los operarios en almacenes de tal forma que así puedan realizar varias acciones simultáneamente.

Es importante señalar que los recursos de ancho de banda consumidos por el sistema distribuido son muy pequeños (4.4 kbps de subida y 6.2 kbps de bajada) con unas buenas prestaciones de reconocimiento y una muy aceptable calidad de reproducción de la respuesta oral. Con este ancho de banda de bajada y subida, haciendo uso de una red inalámbrica WIFI, se pueden dar cabida a un número muy elevado de operarios simultáneamente.

También se ha realizado un estudio experimental de dimensionado, en el que se ha demostrado que con un ordenador personal convencional con un procesador Intel Pentium IV a 3.4 GHz. bajo el sistema operativo Microsoft Windows XP, se puede dar cabida con una calidad de servicio óptima y retardo mínimo a unos 25 usuarios. Si el servidor elegido para hacer los experimentos hubiese sido una máquina más potente de las que se dispone hoy en día normalmente en los servidores de elevada capacidad de cómputo este número de usuarios soportados hubiese sido mucho mayor.

Los autores desean agradecer a la empresa Alerce Informática S.A. su colaboración y aportaciones para la realización de este trabajo.

6. BIBLIOGRAFÍA

[1] ETSI standard doc. “Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Front-end Feature extraction algorithm; Compression algorithms”, ETSI ES 201 108 Ver.1.1.2 (2000-04).

[2] Valin, J.M. “The Speex Codec Manual”, Disponible en línea: http:// www.speex.org (Visitado: 27/07/2008).

SESIÓN ORAL 3
RECONOCIMIENTO DEL HABLA

A NOVEL TWO-LEVEL ARCHITECTURE PLUS CONFIDENCE MEASURES FOR A KEYWORD SPOTTING SYSTEM

Javier Tejedor[1,2], Simon King[2], Joe Frankel[2], Dong Wang[2], José Colás[1] and Javier Garrido[1]

[1]Human Computer Technology Laboratory, Universidad Autónoma de Madrid, Madrid, Spain
[2]The Centre for Speech Technology Research, University of Edinburgh, Edinburgh, United Kingdom
javier.tejedor@uam.es

ABSTRACT

In this work, we present a novel two-level architecture for a keyword spotting system. The first level is composed of an HMM-based keyword spotting process. The second level uses isolated word recognition. Two confidence measures in the decision stage, based on the posteriors and the keywords hypothesised by this second level, are presented and compared within the keyword spotting system. Both confidence measures outperform the performance of the first level in isolation.

1. INTRODUCTION

The increasing volume of audio content has brought with it the need to develop robust speech recognition techniques. Often, search for audio documents has to deal with many words (proper names, acronyms and so on) that do not appear in the vocabulary of the LVCSR systems. Thus, alternatives to LVCSR must be found. Keyword spotting techniques are applied to audio data to retrieve those audio files which contain words related to an application-specific domain. Some of the techniques proposed in the literature are based on phone lattice keyword spotting [1, 2], which exhibits a poor miss rate but has low computational cost. To improve the fast search of keywords within this lattice, a new algorithm presented in [3] achieves a better miss rate performance. Support Vector Machines (SVM) have also been applied to this task [4]. However, in recent years, HMM-based keyword spotting systems have been developed, where filler models vary from phonetic or syllabic units to whole words to deal with the out-of-vocabulary (OOV) words, achieving the best solution in many cases [5, 6]. Methods which combine keyword spotting (with high recall) and phone lattice search have successfully combined the strengths of both methods [6, 7].

Confidence measures play a very important role when dealing with OOV words and reducing the false alarm rate [5, 6, 8, 9, 10]. Posterior probabilities (posteriors) have been used as a confidence measure in speech recognition [11, 12]. Our proposal is to build a new two-level keyword spotting system. The first level in our novel architecture consists of an HMM-based keyword spotting module which uses a pseudo N-gram as language model [13]. Its goal is to achieve high recall, because keywords not proposed at this first level will not be recovered in the following level. The second level consists of an isolated word speech recogniser. It computes the posterior probability of each keyword in the dictionary for those regions of speech proposed as potential keywords by the first level. This produces the confidence measure which we will refer to as ***Posteriors***. It also computes the keyword which best matches with those regions of speech to produce the confidence measure which we will refer to as ***ExactMatch***.

Our experiments were performed using the Spanish geographical-domain ALBAYZIN corpus[14]. Results showed that the ***Posteriors*** confidence measure significantly improved the performance achieved by the first level.

The rest of the paper is organized as follows: The experimental framework is explained in Section 2, Section 3 presents the results and Section 4 gives our conclusions and describes future work.

2. EXPERIMENTAL FRAMEWORK

The architecture, illustrated in Figure 1, is composed of two different levels, each of them containing a different recognition process. The first level is an HMM-based keyword spotting process while the second one is isolated speech recognition using Viterbi decoding. The decision stage uses the information provided by these two levels to confirm or reject the keywords proposed by the first level.

2.1. Motivation

Identifying keywords from a sequence of phones retrieved by a phonetic decoder has been investigated in keyword spotting systems [15]. The use of phone lattices, from an N-best Viterbi recognition pass, leads to improved performance. However, such methods are significantly poorer than whole-word HMM-based methods.

However, producing an N-best list with a LVCSR system has a very high computational cost. In addition, when an HMM-based keyword spotting system is applied to continuous speech, it is very likely that the two few candidates in the N-best list only differ in filler models and

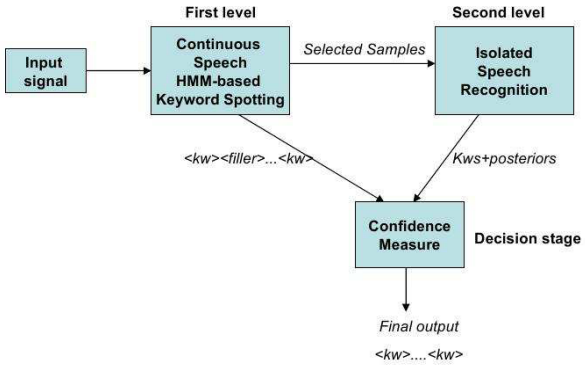


Figure 1. *The whole system architecture*

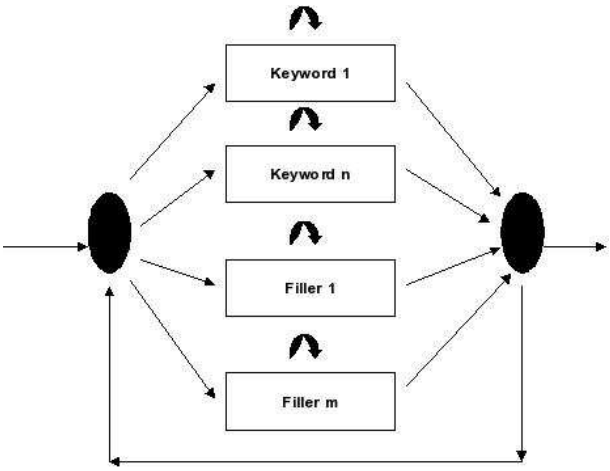


Figure 2. *The recognition network used in the first level (HMM-based keyword spotting). This diagram is taken from [17].*

not in the keyword(s) proposed. Therefore, we propose a two level architecture in which the 1-best keyword candidates from an HMM-based keyword spotting process are further processed using additional information provided by a low cost second process which computes the posterior probability of each candidate. A final decision stage using a confidence measure determines which keywords to output.

2.2. Data

The experiments were performed on the ALBAYZIN database which has a geographical-domain Spanish corpus containing the names of mountains, rivers, cities and so on. 80 keywords were chosen based on high frequency of occurrence and usefulness as a sufficient set for a hypothetical spoken language system for making spoken language searches in a geographical domain. The corpus contains four different sets of data: The ***phonetic training set*** was used to build the HMM acoustic models and contains about 3 hours and 20 minutes of speech. The ***phonetic test set*** was used to estimate the number of Gaussians in each state of each HMM. It contains about 1 hour and 40 minutes of speech. The ***keyword spotting development set*** was used to calculate the thresholds in the ***Posteriors*** confidence measure and the N value in the N-gram language model in the First level and contains about 3 hours and 40 minutes of speech. The ***keyword spotting test set*** was used to evaluate the system and contains about 2 hours of speech.

2.3. Signal representation and features

The input signal (16kHz, 16 bits per sample) was pre-emphasised and transformed into a sequence of frames, using a Hamming window (25 msec window size, 10 msec shift), then characterised by 12 Mel-frequency Cepstral Coefficients (MFCCs) plus energy and their first and second derivatives, giving 39 coefficients in total.

2.4. HMM based acoustic models

The acoustic modelling was the same as in our previous work: "An inventory of 47 allophones of Spanish [16] was used along with beginning and end of utterance silence models to build the monophone and the triphone systems. This set was selected as it achieved higher phone accuracy than a 26-phone inventory in preliminary experiments. All allophone and silence models had a conventional 3-state, left-to-right topology and there was an additional short pause model which had a single emitting state and a skip transition. The output distributions for the monophone system consisted of 15-component Gaussian mixture models (GMM), and those in the triphone used 11 components. In both cases, the number of mixture components were chosen empirically based on phone accuracy on the ***phonetic test set***. The triphone models were cross-word and were state-clustered using HTK's standard decision tree method with phonetically-motivated questions, which leads to 5632 shared states. Keywords are built from the concatenation of these 47 allophones, so no special training is needed to model the keywords. In the same way, a loop of these 47 units was used as filler (garbage) model in the first level of the architecture." [17]

2.5. First level: Continuous speech HMM-based keyword spotting

The Viterbi algorithm in HTK tool [18] is used to find the best path through the labelled segmented network with the recognition network and the language model serving as constraints. The recognition network is composed of a loop of keywords and filler models and is illustrated in Figure 2. This allows any number of keywords to appear in a single utterance.

It is well known that this kind of system tends to retrieve the sequence of phones instead of the keyword that they represent, if the filler model is built from the same

acoustic units as the keywords. To solve this problem, a pseudo N-gram language model, similar to the one proposed by Kim et. al. [13] was used, in which probabilities are simply assigned to the two classes of keyword and filler. As in our previous work, "the probability for the keyword class was set to be 6 and 12 times that of the fillers in the monophone and triphone systems respectively. These ratios were optimised on the ***keyword spotting development set***." [17]. The output of this level is a continuous stream of keywords and filler models, with start and end times.

2.6. Second level: Isolated speech recognition

An isolated word speech recognition system is used to compute various confidence measures. Given the start and end times of the keywords proposed by the first level, it computes the posterior probability for each possible keyword in the dictionary. The computational cost is small because only the speech signal corresponding to potential keywords is processed. A uniform language model is used because no a-priori knowledge about the keywords is available.

A final list composed of the three keywords which achieve the three highest posteriors for each potential keyword proposed in the first level is produced. This is referred to as "kws + posteriors" in Figure 1. We found in previous work that considering only three keywords is sufficient for the ***Posteriors*** confidence measure.

2.7. Decision stage: Confidence measures

Confidence measures have been demonstrated to be a powerful method for reducing the false alarm rate in keyword spotting systems [6, 8].

Let kw be a keyword proposed by the first level and let kw' be the corresponding keyword with highest posterior probability found in the second level. X is the difference between the logarithm of the highest probability in this second level and the logarithm of the second highest one and Y the difference between the logarithm of the highest posterior probability and the logarithm of the third highest one. We define two confidence measures:

The ***ExactMatch*** confidence measure accepts keyword kw if $kw = kw'$; otherwise, the keyword is rejected. The ***Posteriors*** confidence measure accepts keyword kw if $kw = kw'$ and $X \leq X_{beam}$ and $Y \leq Y_{beam}$ where the thresholds X_{beam} and Y_{beam} are set on the ***keyword spotting development set***; otherwise, the keyword is rejected. The difference between the two confidence measures is in the use of the thresholds.

3. RESULTS

The Figure-of-Merit (FOM) was used as the evaluation metric. FOM, defined by Rohlicek in [19] measures the average hit ratio over the range $[1, 10]$ false alarms per hour per keyword. In Table 1 we present the final results

Confidence Measure	CI	CD
None	64.2	68.3
ExactMatch	64.2	68.4
Posteriors	65.5	68.6

Table 1. *Results in terms of FOM for both monophone (CI) and triphone (CD) systems for the first level in isolation (None confidence measure) and for the whole systems with one of the two confidence measures.*

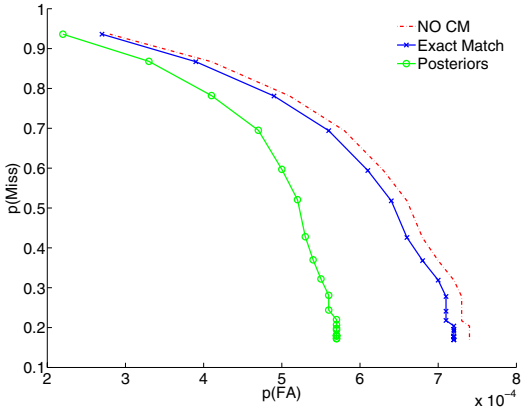


Figure 3. *DET curves of the triphone system with the first level in isolation (NO CM) and the two confidence measures. p(Miss) and p (FA) are miss ratio and false alarm ratio respectively.*

achieved with the two confidence measures. As it is important to know how the second level improved the system, the results achieved by the first level in isolation are also presented in Table 1.

The results in Table 1 suggest that a similar performance is achieved by the ***ExactMatch*** confidence measure as for just the first level in isolation. The differences are not statistically significant using a paired t-test. A paired t-test shows that there is a significant difference in the FOM between the ***Posteriors*** confidence measure and the ***ExactMatch*** confidence measure, for monophone and triphone systems ($p < 0{,}05$).

To show the performance of the system from different operating points, we present in Figure 3 the DET curves. It is shown that the two confidence measures outperform the first level in isolation, being the ***Posteriors*** confidence measure the best.

4. CONCLUSIONS AND FUTURE WORK

We have presented a new two-level architecture developed for a keyword spotting system in which two different confidence measures are proposed to reject false alarms from the first level. The results showed that the ***Posteriors*** confidence measure achieved the best rates both for monophone and triphone acoustics models, with significant improvements over a simpler confidence measure.

In future work, we will investigate new confidence measures for HMM-based keyword spotting systems and will apply the techniques to a spoken term detection task, in which the list of keywords is not known at the time the system is trained. This means that sub-word units [17] must be used to index the audio in a first step, and then a search is performed on this sub-word unit representation for the keywords (spoken terms) in a second step.

5. ACKNOWLEDGEMENTS

JT is a visiting researcher at CSTR, University of Edinburgh. SK holds an EPSRC Advanced Research Fellowship. DW is a Fellow on the EdSST interdisciplinary Marie Curie training programme. JF was funded by the Edinburgh Stanford Link. This work was partly funded by the Spanish Ministry of Science and Education (TIN 2005-06885).

6. REFERENCES

[1] K. Tanaka, Y. Itoh, H. Kojima, and N. Fujimura, "Speech data retrieval system constructed on a universal phonetic code domain," in *Proceedings of IEEE ASRU*, 2001.

[2] P. Yu, K. Chen, C. Ma, and F. Seide, "Vocabulary independent indexing of spontaneous speech," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 5, no. 13, pp. 635–643, 2005.

[3] K. Thambiratman and S. Sridharan, "Rapid yet accurate speech indexing using dynamic match lattice spotting," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 1, no. 15, pp. 346–357, 2007.

[4] Y. Ben-Ayed, D. Fohr, J.P. Haton, and G. Chollet, "Keyword spotting using support vector machines," in *Proc. of International Conference on Text, Speech and Dialogue*, 2002.

[5] H. Cuayahuitl and B. Serridge, "Out-of-vocabulary word modelling and rejection for spanish keyword spotting systems," in *Proc. of Mexican International Conference On Artificial Intelligence*, 2002.

[6] J. Tejedor and J. Colás, "Spanish keyword spotting system based on filler models, pseudo N-gram language model and a confidence measure," in *Proc. of IV Jornadas de Tecnologia del Habla*, 2006.

[7] P. Yu and F. Seide, "A hybrid word/phoneme-based approach for improved vocabulary-independent search in spontaneous speech," in *Proc. of International Conference on Speech and Language Processing*, 2004.

[8] Y. Ben-Ayed, D. Fohr, and J.P. Haton, "Confidence measures for keyword spotting using support vector machines," in *Proc. of International Conference on Acoustics, Speech and Signal Processing*, 1993.

[9] T. Schaaf and T. Kemp, "Confidence measures for spontaneous speech recognition," in *Proc. of International Conference on Acoustics, Speech and Signal Processing*, 1997.

[10] D. Wang, J. Frankel, J. Tejedor, and S. King, "A comparison of phone and grapheme-based spoken term detection," in *Proc. of International Conference on Acoustics, Speech and Signal Processing*, 2008.

[11] F. Wessel, R. Schluter, K. Macherey, and H.Ñey, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 9, pp. 288–298, 2001.

[12] J. Pinto and R.N.V. Sitaram, "Confidence measures in speech recognition based on probability distribution of likelihoods," in *Proc. of Interspeech*, 2005.

[13] J.G. Kim, H. Jung, and H.Y. Chung, "A keyword spotting approach based on pseudo n-gram language model," in *Proc. of Conference Speech and Computer*, 2004.

[14] A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J.B. Mariño, and C.Ñadeu, "Albayzin speech database: Design of a phonetic corpus," in *Proc. of Eurospeech*, 1993.

[15] S.J. Young, M.G. Brown, J.T. Foote, J.F. Jones, and K. Sparck Jones, "Acoustic indexing for multimedia retrieval and browsing," in *Proc. of International Conference on Acoustics, Speech and Signal Processing*, 1997.

[16] A. Quilis, *El comentario fonológico y fonético de textos*, ARCO/LIBROS, 1998.

[17] J. Tejedor, D. Wang, J. Frankel, S. King, and J. Colás, "A comparison of grapheme and phone-based units for spanish spoken term detection," *Speech Communication, Special Issue on Iberian Languages*, 2008.

[18] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *HTK Book v3.2*, Microsoft Department and Cambridge University Engineering Department, 2002.

[19] J.R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden markov modelling for speaker-independent word spotting," in *Proc. of International Conference on Acoustics, Speech and Signal Processing*, 1989.

CUANTIFICACIÓN VECTORIAL DIFERENCIAL PARA LA TRANSMISIÓN EFICIENTE DE PARÁMETROS ACÚSTICOS EN SISTEMAS DE RECONOCIMIENTO AUTOMÁTICO DEL HABLA DISTRIBUIDO

José Enrique García, Alfonso Ortega, Antonio Miguel y Eduardo Lleida.

Grupo de Tecnologías de las Comunicaciones (GTC)
I3A, Universidad de Zaragoza
{jegarlai,ortega,amiguel,lleida}@unizar.es

RESUMEN

El reconocimiento automático del habla distribuido surge como solución a limitaciones de capacidad computacional en dispositivos portátiles de uso cotidiano como teléfonos móviles o PDAs. Debido a las restricciones de ancho de banda que en ocasiones estos dispositivos pueden presentar, se considera necesario el desarrollo de técnicas de transmisión eficientes para el envío de los vectores de parámetros acústicos desde el *Front-End* hacia el *Back-End* del sistema de reconocimiento automático del habla. En este trabajo se presenta un estudio para mejorar la eficiencia en la transmisión de parámetros acústicos basado en el empleo de técnicas de cuantificación vectorial diferencial (DVQ) con el objetivo de reducir al máximo el ancho de banda empleado sin deterioro de las prestaciones del sistema de reconocimiento automático del habla en términos de WER. Se han alcanzado tasas de error comparables a las que se obtendrían sin realizar ningún tipo de compresión con velocidades de transmisión tan bajas como 2.1 Kbps.

1. INTRODUCCIÓN

Los sistemas de reconocimiento automático del habla distribuido se basan en una estructura cliente-servidor en la que uno de los extremos de la comunicación, generalmente el cliente, extrae y envía vectores de características acústicas al otro extremo, generalmente el servidor, que realiza la decodificación acústica. El primero de los extremos recibe el nombre de *Front-End*, mientras que el segundo se denomina *Back-End*. Este reparto de las tareas permite llevar a cabo el desarrollo de aplicaciones con interfaces orales para dispositivos portátiles con baja capacidad. En ocasiones, este tipo de dispositivos cuentan con un reducido ancho de banda, por lo que la transmisión deberá ser realizada del modo más eficiente posible. Además, un servidor debe dar servicio a un elevado número de clientes, siendo conveniente la minimización del ancho de banda utilizado por cada uno de ellos.

Este trabajo ha sido parcialmente financiado a través del proyecto TIN2005-08660-C04.

En este trabajo se presenta una técnica de codificación de parámetros acústicos para su posterior transmisión al *Back-End.* Dicha técnica hace uso de una cuantificación vectorial diferencial (DVQ) para conseguir la mayor compresión posible sin degradar las prestaciones del sistema de reconocimiento automático del habla en términos de su tasa de error.

El presente artículo está organizado del siguiente modo: En la Sección 2 se realiza un breve repaso a las técnicas de cuantificación vectorial. La Sección 3 se dedica a la presentación del bloque extractor de parámetros acústicos y la descripción de las técnicas de compresión de los mismos utilizadas. En la Sección 4 se ofrece un estudio de las prestaciones de las mismas y por último, la Sección 5 presenta las conclusiones.

2. CUANTIFICACIÓN VECTORIAL

La cuantificación vectorial es una técnica de codificación de fuente empleada para representar de un modo compacto un conjunto de valores y tiene sus bases en la Teoría de la Información. Ésta demuestra que siempre que la información mutua entre las diferentes componentes sea no nula, el uso de cuantificación vectorial conjunta conseguirá una representación más compacta que la cuantificación escalar de cada componente por separado. No fue hasta la década de los años 80, cuando su realización práctica se hizo posible a través del trabajo de Linde, Buzo y Gray [1]. Dicho método surge como generalización del algoritmo de Lloyd [2] que a su vez puede verse como una solución heurística del problema de *k-means* [3].

El algoritmo *k-means* describe el procedimiento para realizar la agrupación (*clustering*) de un conjunto dado de elementos en k clases. Así, dicho algoritmo constituye un método para obtener un *codebook* (diccionario) formado por *codewords* (palabras), de manera que en el proceso de cuantificación se represente cada uno de los vectores de entrada con el *codeword* más próximo en el sentido de mínima distorsión. Es el índice de dicho *codeword*, lo que es enviado al decodificador para posteriormente reconstruir el vector de entrada. Una de las medidas de distorsión más comúnmente utilizadas es la distancia euclídea.

$$d(\mathbf{x},\tilde{\mathbf{x}}) = (\mathbf{x}-\tilde{\mathbf{x}})^T(\mathbf{x}-\tilde{\mathbf{x}}) \quad (1)$$

El algoritmo *k*-means guarda ciertas similitudes con el algoritmo *Expectation-Maximization (EM)* para mezclas de gaussianas [4] bajo ciertas condiciones: a) Las componentes de la mezcla poseen matrices de covarianza diagonales con elementos unitarios con la distancia euclídea como medida de distorsión. b) Los pesos de las componentes son todos iguales. c) Mientras que el algoritmo *k-means* realiza asignaciones "*hard"* (deterministas) de los elementos a los *clusters,* el algoritmo *EM* realiza un cálculo de las probabilidades de pertenencia a cada *cluster*. Sin embargo, puede modificarse el algoritmo *EM,* realizando asignaciones "*hard"* de elementos a *clusters* para llegar a la solución *k-means* [5]

3. COMPRESIÓN DE PARÁMETROS ACÚSTICOS.

Con el objetivo de dotar de la máxima eficiencia al uso del ancho de banda requerido, se ha llevado a cabo el estudio de un conjunto de técnicas de compresión de parámetros acústicos para el reconocimiento automático del habla distribuido. Partiendo del Front-End estandarizado por ETSI ES 201 108 V1.1.2 [6] se han estudiado diferentes modos de compresión de parámetros basados en la modificación del bloque de cuantificación vectorial en él previsto.

3.1. *Front-End* para RAH distribuido ETSI ES 201 108 V1.1.2.

El estándar ETSI ES 201 108 V1.1.2 presenta un conjunto de algoritmos para la extracción de características acústicas y su posterior transmisión para sistemas distribuidos de reconocimiento automático del habla. El algoritmo de extracción de características ofrece a su salida vectores de parámetros consistentes en 13 coeficientes cepstrales junto con el coeficiente de log-energía cada 10 ms. Asimismo, define un algoritmo de compresión para reducir la tasa de transmisión. Esta compresión está basada en la cuantificación vectorial de dichos vectores tomados por parejas, dando lugar así a 7 valores cuantificados (el coeficiente C0 se cuantifica junto con el coeficiente de log-energía y el resto tomando parejas de forma correlativa).

3.2. Cuantificación vectorial diferencial (DVQ) de parámetros acústicos.

En primer lugar y para comprobar que es posible la compresión de la información de salida del algoritmo de extracción de parámetros acústicos definido en el estándar, se llevó a cabo el estudio de la Información mutua de cada una de las parejas que define dicho estándar. En la Figura 1 se muestra la Información mutua estimada haciendo uso de un subconjunto de la base de datos Albayzin [7] para las parejas de coeficientes que van desde el 1 hasta el 12. En ella puede observarse cómo los coeficientes cepstrales tomados de dos en dos presentan una información mutua no nula que permite su compresión a través de técnicas de cuantificación vectorial.

Seguidamente, para evaluar la eficiencia de los cuantificadores vectoriales propuestos por el estándar se propuso la realización de un conjunto de *codebooks* nuevo para cada una de las parejas que toma el estándar a través de la aplicación del algoritmo *k-means*. Para la obtención de estos *codebooks* se hizo uso de un subconjunto de la base de datos Speech-Dat Car en español [8]. Con el objetivo de encontrar la longitud más apropiada para dichos *codebooks* se realizó un barrido con 8, 16, 32 y 64 *codewords*.

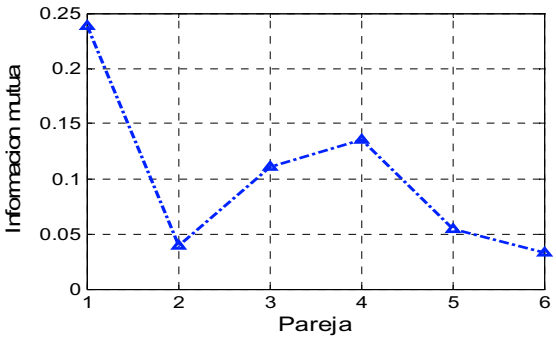


Figura 1. *Información mutua estimada para las parejas de coeficientes cepstrales.*

A continuación, se realizó una modificación sobre la estructura anterior para hacer más específicos los *codebooks* al cuantificar de manera distinta los sonidos de alta energía y los de baja energía. Para ello se optó por el uso de un umbral sobre el parámetro de energía que realizase una preclasificación de las tramas. El conjunto de datos de entrenamiento usado fue el anteriormente mencionado.

Por último, se llevó a cabo la cuantificación vectorial diferencial (DVQ), que utiliza codificación DPCM (Differential Pulse Code Modulation) empleada usualmente en compresión de audio y video digital, realizando la etapa de cuantificación de forma vectorial. Esquemas similares en cuantificación de coeficientes MFCC se habían propuesto anteriormente, como realizar predicción lineal para posteriormente hacer una cuantificación vectorial en dos etapas [9]. La diferencia fundamental de este sistema frente a DVQ reside en que a diferencia de lo que sucede en ésta, el error de cuantificación se va acumulando para tramas sucesivas lo que puede degradar las prestaciones del sistema en el caso de realizar predicción lineal adaptativa. En ese caso los coeficientes del filtro predictor no se podrán calcular con la aproximación 'Backward' en el decodificador al no disponer de las mismas señales que en transmisión y deberán ser enviados (aproximación 'Forward') con el consiguiente incremento del ancho de banda.

En la Figura 2 se muestra un esquema del cuantificador vectorial diferencial empleado, que realiza la predicción lineal de forma individual sobre cada uno de los coeficientes, mientras que la cuantificación es vectorial, con las parejas de coeficientes definidas en el

estándar ETSI. Cada pareja de coeficientes MFCC se denotada por la dupla $\mathbf{x} = (x_a, x_b)$. De esta pareja se sustraen sendas predicciones $\hat{\mathbf{x}} = (\hat{x}_a, \hat{x}_b)$ realizadas a partir de los valores cuantificados de la trama anterior, obteniéndose así la pareja de errores de predicción

$$\mathbf{d} = (d_a, d_b) = \mathbf{x} - \hat{\mathbf{x}} \quad (2)$$

posteriormente, estos errores se cuantifican dando lugar a $\tilde{\mathbf{d}} = (\tilde{d}_a, \tilde{d}_b)$, el error de predicción cuantificado

$$\tilde{\mathbf{d}} = \mathbf{d} + \mathbf{e}_q \quad (3)$$

donde $\mathbf{e}_q = (e_{q_a}, e_{q_b})$ es el error de cuantificación.

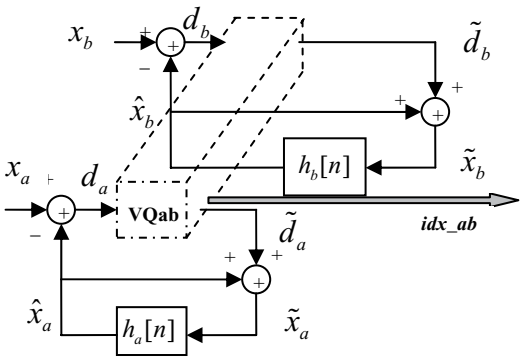


Figura 2. *Esquema del cuantificador vectorial diferencial para una pareja de coeficientes MFCC.*

Los errores de predicción de (3) serán usados para obtener la pareja de coeficientes cuantificados a través de los cuales se obtendrá la predicción de los parámetros de la trama siguiente. Los filtros de predicción lineal aparecen denotados como $h_a[n]$ y $h_b[n]$ aunque para este primer estudio se han sustituido por simples elementos de retardo de manera que el valor predicho de cada coeficiente es directamente su valor cuantificado en la trama anterior.

Uno de los principales efectos sobre los coeficientes que tiene la aplicación de técnicas de codificación diferenciales es la reducción de su varianza. Esto se ilustra en la Figura 3 dónde se representan el histograma de los coeficientes C1 y C2 (izquierda) junto con el histograma del error de predicción de los coeficientes C1 y C2 (derecha) estimados a partir de un subconjunto de la base de datos Albayzin.

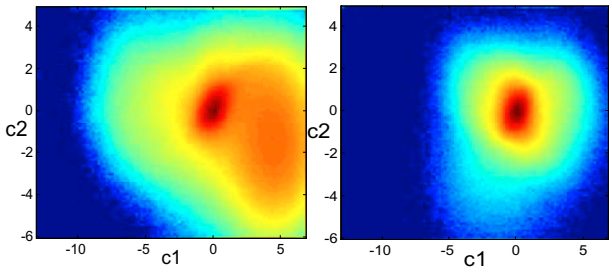


Figura 3. *Histogramas de la primera pareja de coeficientes MFCC (izquierda) y de los errores de predicción de la primera pareja (derecha).*

Esta reducción de varianza permite que con el mismo número de *codewords* se obtenga una menor distorsión media al poder tener los centroides de las mismas más próximos entre si, cubriendo la mima proporción de valores de la señal a representar.

4. EVALUACIÓN DE PRESTACIONES.

Para evaluar las prestaciones de los distintos esquemas de compresión presentados, se llevaron a cabo un conjunto de experimentos en los cuales se valoró tanto el error de cuantificación como las prestaciones del sistema completo, en términos de la tasa de error obtenida por un sistema de RAH determinado para una tarea concreta.

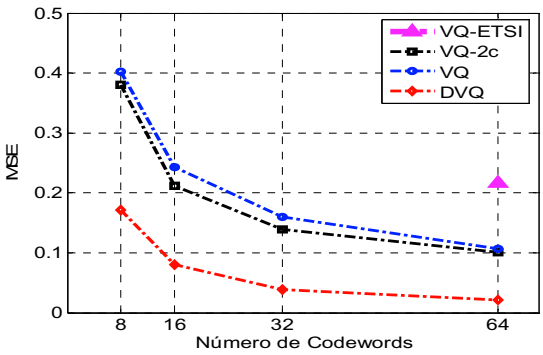


Figura 4. *Error cuadrático medio de las distintas aproximaciones de compresión en función del número de palabras del codebook .*

En cuanto a la distorsión introducida por la cuantificación, que presentan los distintos esquemas, en la Figura 4 se muestra la evolución del error cuadrático medio (MSE) para el conjunto de los 14 parámetros en función de la longitud del *codebook*. Como puede observarse, la aproximación que menor error cuadrático medio presenta es la de la cuantificación vectorial diferencial, mejor incluso que la cuantificación vectorial a partir de la definición de dos clases en función de la energía que a su vez presenta menores errores en general que un *codebook* no específico. Junto a estas representaciones se presenta el error cuadrático medio obtenido con el *codebook* propuesto en el estándar, con una distorsión superior a todas las aproximaciones presentadas, para el mismo número de *codewords*. Los valores han sido obtenidos haciendo uso de la base de datos Albayzín.

Por otro lado, se realizaron un conjunto de experimentos de reconocimiento para comprobar la validez de los esquemas de compresión propuestos. El B*ack-end* empleado hace uso de modelos continuos de palabra con 3 estados cada uno y modelos de mezclas de gaussianas como probabilidades de observación de 16 componentes cada una. La tarea tomada para la evaluación es el reconocimiento de dígitos continuos y aislados en castellano pertenecientes a la base de datos Speech-dat Car. La evaluación se llevó a cabo tanto con señal limpia (tomada de un micrófono de cercanía o Close-Talk) como con señal ruidosa (tomada de un micrófono situado en el techo del vehículo).

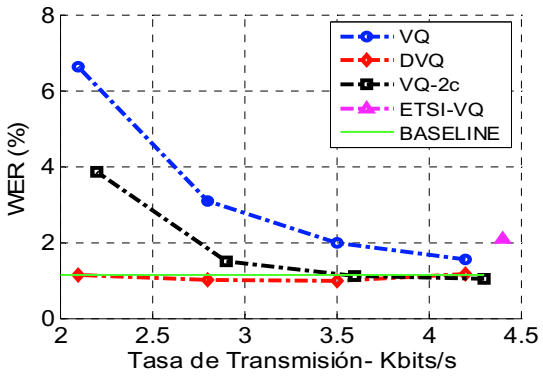


Figura 5. *Valores de Word Error Rate (WER) en función de la tasa de transmisión necesaria para los distintos esquemas de compresión y señal límpia.*

En la Figura 5 se muestra la tasa de error obtenida con señal limpia para los distintos esquemas de compresión propuestos en función del ancho de banda empleado para la transmisión junto con el error que se obtendría si no se realizase ningún tipo de compresión (*Baseline*). Como puede observarse la tasa de error desciende a medida que se incrementa el tamaño del *codebook* y por tanto se aumenta el ancho de banda necesario para la transmisión en los esquemas de cuantificación vectorial básico(VQ) y con preclasificación de la entrada en dos clases en función de la energía (VQ-2c). Sin embargo, la tasa de error se mantiene prácticamente invariante con la tasa de transmisión para el esquema propuesto (DVQ) y en valores muy similares al caso de hacer uso de ningún tipo de compresión (Baseline). Como referencia, se presenta la tasa de error obtenida con la cuantificación vectorial propuesta por el estándar (ETSI-VQ).

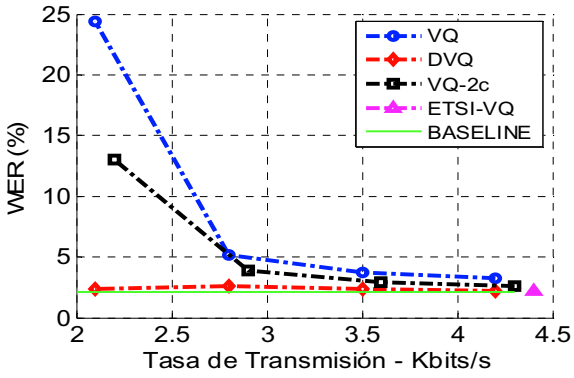


Figura 6. *Valores de Word Error Rate (WER) en función de la tasa de transmisión necesaria para los distintos esquemas de compresión y señal ruidosa.*

Por último, en la Figura 6 puede verse la evolución de la tasa de error obtenida con señal ruidosa para los distintos esquemas de compresión propuestos en función del ancho de banda empleado para la transmisión junto con el error que se obtendría si no se realizase ningún tipo de compresión (*Baseline*). En ella se aprecia, al igual que en la figura anterior, el descenso en la tasa de error cuando se aumenta la tasa de transmisión empleada para VQ y VQ-2c, hasta llegar a valores comparables con los obtenidos con el éstandar (ETSI-VQ). Sin embargo, la aproximación diferencia (DVQ) mantiene valores comparables al *Baseline* incluso con tamaños muy pequeños del *codebook*, es decir con tan sólo 8 *codewords,* 2.1 kbps, la tasa de error obtenida ya se encuentra en valores comparables a los obtenidos con el estándar con 4.4 kbps.

5. CONCLUSIONES

En el presente trabajo se ha presentado un estudio para mejorar la eficiencia en la transmisión de parámetros acústicos basado en el empleo de técnicas de cuantificación vectorial diferencial (DVQ). Dicha aumento de eficiencia tiene por objetivo la reducción del ancho de banda empleado en la transmisión de vectores de características acústicas en sistemas de reconocimiento automático del habla distribuido, sin degradar las prestaciones del sistema en términos de WER. Se han alcanzado tasas de error comparables a las que se obtendrían sin realizar ningún tipo de compresión con velocidades de transmisión tan bajas como 2.1 Kbps, lo que indica que la técnica propuesta puede ser apropiada para ser incluida en determinadas aplicaciones con interfaces orales sobre dispositivos móviles con restricciones de ancho de banda.

6. BIBLIOGRAFÍA

[1] Linde, Y., Buzo, A., Gray, R., "An Algorithm for Vector Quantizer Design", IEEE Trans on Comm., v. 28, (1980).

[2] Stuart P. Lloyd. "Least Squares Quantization in PCM." IEEE Trans. on Inf. Theory, vol. 28(2), pp. 129-137, 1982.

[3] Huang, X., Acero, A., Hon, H., "Spoken Language Processing: a guide to theory, algorithm, and system development" (2001) Prentice Hall.

[4] Dempster, A. P., Laird, N. M., and Rubin, D. B., "Maximum likelihood from incomplete data via the EM algorithm," J. R. Statist. Soc., vol. 39, (1) , pp. 1–21, 1977

[5] Qiu, D. and Ajit Tamhane, C., "A comparative study of the K-means algorithm and the normal mixture model for clustering: Univariate case" Journal of Statistical Planning and Inference. vol. 137 (11), pp. 3722-3740, 2007.

[6] ETSI standard doc. "Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Front-end Feature extraction algorithm; Compression algorithms", ETSI ES 201 108 Ver.1.1.2 (2000-04).

[7] Casacuberta, F., García, R., Llisterri, J.,Nadeu, C., Pardo, J.M., Rubio, A. "Development of Spanish Corpora for Speech Research (Albayzin)",Workshop on Standardization of Speech Databases and Speech Assessment Methods. (1991).

[8] Moreno, A., Lindberg, B., Draxler, C., Richard, G., Choukri, K., Euler, S., and Allen, J., "Speechdat-car: A large speech database for automotive environments", LREC (2000).

[9] Ramaswamy, G. N. and Gopalakrishnan, P. S., "Compression of Acoustic Features for Speech Recognition in Network Environments" in Proc of ICASSP, Seattle. 1998.

IMPROVED UNSUPERVISED SPEECH RECOGNITION SYSTEM USING MLLR SPEAKER ADAPTATION AND CONFIDENCE MEASUREMENT

Mukund Jha[a], Sourabh Sriom[b], Míriam Luján[c], Carlos D. Martínez-Hinarejos[c], Alberto Sanchís[c]

[a]Department of Computer Science, MNNIT, Allahabad, 211004, Allahabad, India
[b]Department of Electronics and Communications, IIT Guwahati, 781039, Guwahati, India
[c]Instituto Tecnológico de Informática, Universidad Politécnica de Valencia,
Camino de Vera, s/n, 46022, Valencia, Spain

ABSTRACT

A robust ASR system needs to perform well in different environment and with different speakers. For this reason speaker adaptation has become an essential part of a state of art ASR system. Here we show how confidence measurement technique can be used to improve the quality of unsupervised speaker adaptation. An initial speaker-independent system is adapted to improve the modelling of a new speaker by modifying HMM parameters using Maximum Likelihood Linear Regression(MLLR) technique. Improvement gained from unsupervised speaker adaptation technique are lowered because of their dependency on the accuracy of recognition in first pass. We use confidence measures to improve the performance by selective adaptation. We present experimental results on the 8 speakers' data from Wall Street Journal.

1. INTRODUCTION

Even after significant amount of improvement in speaker independent speech recognition systems, error rates are still quite higher when compared to speaker dependent (SD) systems, and dependence on large amount of speaker specific data for training make SD unsuitable for many application. Many system make use of speaker adaptation techniques to adapt to new speakers. These techniques can be either supervised, where the correct word transcription of the adaptation data is known or unsupervised, where it is not known. Unsupervised adaptation relies on recognizer to provide a transcript for the spoken utterances in the first pass, which is used to adapt the model during the training. But these transcriptions contain recognition errors and out-of-vocabulary words which degrade the performance of adaptation technique. Confidence measures can be used to classify the words in the recognized transcript as correct or incorrect, which allows the system to use only those words for adaptation which are most probably correct.

The remainder of the paper is organized as follows: first, we give a description of the MLLR technique for speaker adaptation, next we give a brief description of the confidence measurement technique and describe the use of confidence measurement for MLLR adaptation, followed by the details of experiments. We conclude this paper with results and a summary of the work.

2. MAXIMUM LIKELIHOOD LINEAR REGRESSION

Speaker adaptation applied to HMMs mostly involve the techniques that uses the original models as the starting point and add speaker specific information by transforming some of the parameters in the models. The general idea is that the fully trained model should contain some general speech information which will be used for the new system as well. It is also assumed that even the smallest amount of adaptation data would contain some speaker specific information.

The MLLR technique follows the above assumptions for adapting only the mean vectors of continuous density HMMs. However, the adaptation can also be performed for the covariance matrices to improve the results. If a transformation matrix can be estimated specifically for the covariance matrix, it is likely that an improvement in performance can be achieved by transforming the covariances as well. A detailed discussion on this is presented in [1, 2]. In this section we shall discuss the general theory behind the MLLR technique and its evaluation given a set of observation data.

The MLLR adaptation involves transforming the means of the HMM Gaussians. This transformation is preformed by applying a *transformation matrix W*. Therefore, given a gaussian s with mean μ_s, the adaptation consists of re-evaluating the new mean $\hat{\mu_s}$ as below:

$$\hat{\mu_s} = W_s \mu_s \tag{1}$$

where W_s is the adaptation matrix for the gaussian s and an *offset*(or bias) value, ω_s is introduced in the mean vector. This gives us the *extended mean vector*, $\tilde{\mu} = [\omega_s : \mu_s]$. Now the equation (1) can be modified to:

$$\hat{\tilde{\mu}_s} = \tilde{W}_s \tilde{\mu_s} \tag{2}$$

here, if the dimension of μ_s is n, the dimension of $\tilde{W}_s$ would be $n \times (n+1)$.

The transformation can be evaluated for each gaussian in the acoustic models. However, this would require a huge amount of data for the adaptation process. To solve this problem we group the gaussians into what is referred to as *regression classes*, which are the sets of gaussians which share the same transformation matrix. Regression classes are discussed in detail in [3].

Therefore, for a given set of adaptation samples from a particular regression class c, denoted by the sequence of acoustic features vectors $x_1^T = x_1, x_2, ..., x_T$, the adaptation matrix $\hat{\tilde{W}}$ for a Viterbi approximation can be estimated as below:

$$\hat{\tilde{W}}_c = (\sum_{t=1}^{T} x_t \mu'_{s_t})(\sum_{t=1}^{T} \mu_{s_t} \mu'_{s_t})^{-1} \quad (3)$$

where s_t denotes the most likely state (and gaussian) in the Viterbi path at time t and μ' is the transpose of the mean vector.

3. CONFIDENCE MEASUREMENT

The speaker recognition systems that are available to us are not completely free of errors. To develop an efficient speaker independent speech recognition system using MLLR approach, we must have the knowledge about the reliability of the recognized words. Therefore the goal of confidence measurement is to detect words that are likely to have errors in their recognition. In other words confidence measurement would be used for each hypothesized word to classify it as either *correct* or *incorrect*. Such a classification is done using *confidence measures*, which are essentially normalized scores to help the system decide on the reliability of the recognized words. Finally, only those words, which have been tagged as correct would be used for the MLLR adaptation and hence yield better results in the subsequent recognitions.

The Bayes' decision rule is the fundamental rule in all statistical speech recognition systems. The Bayes' rule is based on the posteriori probability $p(w_1^M|x_1^T)$ of a word sequence $w_1^M = w_1, w_2, ..., w_M$ given a sequence of acoustic observations $x_1^T = x_1, x_2, ..., x_T$. That word sequence $[w_1^M]_{opt}$ which maximizes this posteriori probability would also minimize the probability of an error in the recognized sentence:

$$[w_1^M]_{opt} = argmax_{w_1^M} p(w_1^M|x_1^T)$$

$$= argmax_{w_1^M}[\frac{p(x_1^T|w_1^M).p(w_1^M)}{p(x_1^T)}]$$

$$= argmax_{w_1^M}[p(x_1^T|w_1^M).p(w_1^M)]$$

where, $p(w_1^M)$ denotes the language model probability, $p(x_1^T|w_1^M)$ the acoustic model probability and $p(x_1^T)$ is the probability of acoustic observations.

If all these posteriori probabilities are known to us, the posteriori probability $p(w_m|x_1^T)$ for a specific word w_m could be estimated by summing up the posteriori probabilities of all sentences w_1^M containing this word at position m. This posterior word probability can now be used as an efficient measure of confidence.

The probability of the sequence of acoustic observations $p(x_1^T)$ is normally omitted since it is invariant to the choice of a particular sequence of words. Thus, the decisions during the decoding phase are based on unnormalised scores. These scores can be used for a comparison of competing sequences of words, but can not be used to predict which of the recognized words are correct. The estimation of probability of the acoustic observations thus, is the main problem for the computation of confidence measures.

The usefulness of word graphs in confidence measurement is well known. In [4] the proposed features based on word graphs are the most important predictors. In [5] the confidence measure is estimated on word graphs directly by the posterior probability of a hypothesized word given all the acoustic observations of the utterance. The word posterior probability based on word graphs is used in [6] along with a large set of other authors use a single word graph which is obtained through the recognition process. We have used single word graphs for the evaluation of confidence measures in our experiments and an overview of estimating the posterior probabilities based on a single word graph is discussed below.

3.1. Posterior probabilities on word graphs

A word graph G is a directed, acyclic, weighted graph. The nodes corresponds to discrete points in time. The edges are triplets $[w,\ s,\ e]$, where w is the hypothesized word from node s to node e. The weights are scores associated to the word graph edges. Any path from the initial to the final node forms a hypothesis h.

Given the acoustic observations $\vec{\Theta}_1^T$, the posterior probability for a specific word (edge) $[w,\ s,\ e]$ can be computed by summing up the posterior probabilities of all hypotheses of the word graph containing the edge $[w,\ s,\ e]$:

$$P([w,\ s,\ e] \mid \vec{\Theta}_1^T) = \frac{1}{P(\vec{\Theta}_1^T)} \sum_{\substack{h \in G: \\ \exists [w', s', e']: \\ w'=w, s'=s, e'=e}} P(h,\ \vec{\Theta}_1^T) \quad (4)$$

The probability of the sequence of acoustic observations $P(\vec{\Theta}_1^T)$ can be computed by summing up the posterior probabilities of all word graph hypotheses:

$$P(\vec{\Theta}_1^T) = \sum_h P(h\ , \vec{\Theta}_1^T) \quad (5)$$

These posterior probabilities can be efficiently computed based on the well-known *forward-backward* algorithm [5].

3.2. Scaling of the probabilities

In the evaluation of confidence measures for the recognized words we also include a scaling factor α which plays an important role in the evaluation of the posterior probabilities and their performance as a confidence measure. If the acoustic model probabilities are not scaled appropriately, the sums of the equations mentioned above will be dominated by only a few word graph hypotheses because of very large dynamic range of the acoustic scores (which is the negative logarithm of the unnormalized acoustic probabilities). The differences in acoustic scores arise mainly due to the variance of acoustic features which are presumable underestimated. To avoid the re-evaluation of these variances, it is better to scale the acoustic probabilities in order to have efficient results. Training data for each speaker was tested for different values of α to decide on the optimum α value for each speaker, α giving the minimum *Classification Error Rate (CER)* for a speaker was taken to be optimum. Same value of scaling factor was used during the testing phase for each speaker.

3.3. Threshold for confidence scores

The confidence measurement technique gives us a score (or probability) of a word's reliability. The system now sets a certain threshold τ, a value between 0 and 1 (probability score) for each speaker in the corpus after analyzing the *Classification Error Rate (CER)* for each speaker separately. Threshold for a given speaker and given α is the probability score which gives the minimum *CER*. All words recognized for a given speaker, which have their confidence measures above this threshold are classified as correct while the ones having their measures below this threshold are labeled incorrect. The correct words are now the ones that are used for the MLLR adaptation.

4. EXPERIMENTS

We performed some interesting experiments with confidence measures and MLLR technique. Wall Street Journal was used for all the experiments. Recognition was performed using a trigram language model and 20k lexical model using iATROS, an HMM based continuous speech recognizer. First different parameters of the recognizer were optimized to give low word error rate. Acoustic scaling factor, α, used for confidence measurement, was optimized for each corpus to give better confidence scores and baseline word error rates were determined for each speaker.
Experiments were made with the following kind of trainings.

1. **Full MLLR**: This is the standard MLLR adaptation. We use all the time frames to train the models, thus includes error from the recognizer. This gives us a base on MLLR adaptation on which we try to improve using confidence measure.

2. **MLLR with Confidence Measure**: In this we apply confidence measure on the output of the recognizer before estimation of the adaptation matrix. Only time frame of high confidence words were used for adaptation. Experiment was done with two different types of gaussian means.

   - **Max**: It is the standard method, means from the most probable gaussian in the gaussian mixture of the HMM state were used during adaptation.
   - **Normalized**: Instead of using the means only from the most probable gaussian and ignoring the means from the rest of the mixture, we use a normalized mean from all the gaussians of the mixture. We normalize the means of the gaussian mixture by taking the means from each mixture in the ratio of its emission probability, i.e. in ration of its contribution to the state emission probability.
     Assume a mixture having N gaussians and let μ_{ij} denote the *ith* mean from *jth* mixture, let p_j be the probability of emission of the observed feature frame by *jth* gaussian and p_t be the probability of emission of the state, sum of emissions of all gaussians. Then normalized mean μ_i' is given by the following expression.

     $$\mu_i' = \sum_{j=1}^{N} \frac{p_j}{p_t} \mu_{ij}$$

     Normalized means has given better results for some speakers than using the means from maximum gaussian.

3. **MLLR with ideal CM**: We performed this experiment to find the upper limit which can be achieved through MLLR in unsupervised training. In this only the frames of correctly recognized words were used in calculating the adaptation matrix. We used the correct transcription to compare the recognition output and only correctly recognized words were used for the adaptation. Experiments were performend using means from the most probable gaussian (Max). The results were very close to the supervised training.

4. **Supervised (Ideal Recognizer)**: Lastly, we performed supervised training on the corpus. This was done by performing forced recognition with the actual transcription of the sentences as language model during training. The case idealizes a recognizer and gives an upper limit that can be achieved using MLLR adaptation technique.

Speakers	Baseline	Full	Max	Norm	Improvement	Correct	Supervised
46h	39.94	34.65	33.33	32.82	5.28 %	32.91	32.47
47b	32.52	29.60	29.55	29.85	0.17 %	28.93	28.87
47h	14.26	12.37	12.25	12.52	0.97 %	11.46	11.56
47n	52.85	41.36	40.37	40.81	2.39 %	38.70	34.70
48r	10.63	9.63	9.76	9.86	-1.34 %	8.83	8.69
48v	18.33	18.19	18.33	17.83	1.98 %	17.71	17.56
49n	9.28	8.82	8.93	8.98	-1.22 %	7.40	7.40
4am	31.26	22.92	22.13	21.75	5.10 %	21.92	21.98
Average	26.51	22.19	21.83	21.80	1.66 %	20.98	20.40

Table 1. The numbers indicate the WER and the improvement is the relative decrease in the WER while comparing the Full and the minimum of Max and Norm values. The figures under *Full* indicate the results for MLLR adaptation without the use of confidence measurement, while those under *Max* are the results for MLLR adaptation using CM with most probable means, and the figures under *Norm* represent the results for MLLR adaptation using CM with normalized means.

5. RESULTS

Experiments were performed on data from 8 different speakers of Wall Street Journal corpus, each having about 150 utterances. During the training phase 50 sentences were used and testing was done with the remaining 100 sentences. MLLR training is done using a single regression class for all the frames. We obtained significant improvement in word error rates in case of some speakers after using confidence measures. We gained a relative improvement as high as 5 % in some cases when compared with MLLR technique without using confidence measures.

Also we observed adaptation using normalized means outperformed most probable gaussian's mean (Max), for some speakers. Although on an average, their performances are comparable. A more detailed study is needed for selection criteria between normalized and maximum probable gaussian mean. We also observed MLLR with only correct words (ideal CM) gave word error rate very close to those obtained with supervised adaptation. Table 1 summarizes the results.

6. CONCLUSIONS AND FUTURE WORK

In this work we have shown that use of confidence measures for MLLR adaptation improves the adaptation performance. We have also shown that normalizing the means of gaussian mixture can be an alternative, though more experiments have to be performed to judge the selection criteria for the types of mean. We have shown the performance of supervised adaptation is superior than other unsupervised adaptation, because supervised MLLR is able to reduce the mismatch between the acoustic models and acoustic vectors of incorrectly recognized words in first pass. We also found the performance of unsupervised adaptation with only correct words is close to that of supervised, thus shows a good confidence measure technique can raise the level of unsupervised MLLR adaptation. In future more experiments can be done using different regression classes and affect of different confidence measurement parameters can be tested to see the improvement.

7. REFERENCES

[1] C.J. Legetter and P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language*, pp. 171–185, 1995.

[2] C.J. Legetter and P.C. Woodland, "Flexible speaker adaptation for large vocabulary speech recognition," *Proceedings EUROSPEECH95*, pp. 1155–1158, 1995.

[3] M. Pitz, F. Wessel, and H. Ney, "Improved mllr speaker adaptation using confidence measures for conversational speech recognition," *Proceedings of ICSLP*, pp. 548–551, 2000.

[4] T. Kemp and T. Schaaf, "Estimating confidence using word lattices," *Proceedings of EUROSPEECH*, pp. 827–830, 1997.

[5] F. Wessel, "Confidence measurement for large vocabulary continuous speech recognition," *IEEE Trans. on Speech and Audio Processing*, pp. 9(3):288–298, 2001.

[6] D. Vergyri, "Use of word level side information to improve speech recognition," *Proceedings of ICASSP*, pp. 1823–1826 vol.3, 2000.

NEW FEATURES FOR IMPROVING VAD WHEN DEALING WITH FAR-FIELD AND MULTI-SPEAKER SPEECH

Oscar Varela Serrano [1], Rubén San-Segundo Hernández [2], Luis Alfonso Hernández Gómez[3]

[1] Telefónica I+D. Madrid, Spain
[2] Grupo de Tecnología del Habla, UPM
[3] Grupo de Aplicaciones del Procesado de Señal, UPM
ovs@tid.es

ABSTRACT

This paper describes new acoustic features for improving VAD (Voice Activity Detection) when dealing with speech mixed with far-field and multi-speaker speech. Background voices are one of the major causes for the degradation of speech recognition performance in spoken dialog systems (specially over mobile phones). Also, in any audio indexing application, to separate the voice of a target speaker from other background speakers can be necessary. This paper studies three new features to discriminate between near-field, far-field and background multi-speakers speech: 1) the percentage of frame-by-frame change for the best HMM mixture in a HMMs-based VAD; 2) the Mahalanobis distance between MFCCs from consecutive speech frames, and 3) the maximum auto-correlation value for each speech frame. Experimental results on the Av16.3 speech database for the best feature, obtain classification errors below 19% for near-field vs. far-field speech, and 3.5% for one-speaker vs. multi-speaker.

Index Terms: VAD, far-field speech, multi-speaker speech.

1. INTRODUCCIÓN

This paper addresses the problem found in many speech-based applications when speech of the user to be recognized is contaminated with background voices from other speakers standing still or moving. Far-field speech is specially problematic and usual in mobile phone scenarios, where the main speaker can be situated in open environments surrounded with far-field interfering speech from other speakers. In this case, VAD systems can detect far-field speech as coming from the user increasing the speech recognition error rate. Generally, errors caused by background voices mainly increase word insertions and substitutions, leading to important dialogue misunderstandings.

In several previous works, similar measures as the ones this work considers have been used for dereverberation techniques. In [1] for example, authors use the idea of reverberation for restoring speech degraded by room acoustics using stereo (two microphone) measures. To do this, cepstra operations are made when observations have nonvanishing spectra. Other dereverberation technique, presented in [2], uses the pitch as primary analysis feature. That method starts estimating pitch and harmonic structure of the speech signal to obtain a dereverberation operator. After that, this operator is used to enhance the signal through an inverse filtering operation. Single channel blind dereverberation was proposed in [3] based on auto-correlation functions of frame-wise time sequences for different frequency components. A technique for reducing room reverberation using complex cepstral deconvolution and the behavior of room impulse responses was presented in [4]. Reverberation reduction using least square inverse filtering has been also used to recover clean speech from reverberant speech. Yegnanarayana shows in [5] a method to extract time-delay between two speech signals collected at two microphone locations. The time-delay is estimated using short-time spectral information (magnitude, phase or both) based on the different behavior of the speech spectral features affected by noise and reverberation degradations. Finally, Cournapeau shows in [6] a VAD based on High Order Statistics to discriminate close and far-field talk, enhanced by the auto-correlation of LPC residual.

Nowadays there is an increasing interest on the relevance of VAD systems in real applications. New VAD techniques are being proposed, see for example the work of Ramirez et al. [7] on robust VAD using the Kullback-Leibler divergence measure. However, although experimental results are usually given for the AURORA database, to our knowledge there are no similar results for speech in the presence of far-field voices.

In this paper, trying to contribute to the improvement of VAD systems in the presence of background speech, we present a preliminary analysis of new features suitable to classify near-field, far-field and multi-speaker speech. We consider simple acoustic feature that could be easily and cost-effective integrated in state-of-the art VAD.

The rest of this paper is organized as follows: the speech database and our experimental evaluation framework are described in Section 2. Section 3, 4 and 5 present three different features for far-field and multi-speaker discrimination together with their corresponding evaluation results. Finally, some conclusions are given in Section 6.

2. SPEECH DATABASE

The database we used in this work is the Av16.3 speech database composed of audio-visual data recorded in a meeting room context. For this work, only the audio data has been considered. This audio has been recorded with 16 microphones perfectly synchronized and calibrated conveniently. For each recording, there are 16 audio WAV files from the two circular 8-microphone arrays (Fig. 1) sampled at 16 KHz and WAV files recorded from lapels also sampled to 16 KHz. It is specially important to point out that overlapped speech has been recorded when there are several speakers speaking simultaneously.

In order to allow for such a broad range of research topics, "meeting room context" is defined in a wide way. This includes a high variety of situations, from "meeting situations" where speakers are seated most of the time, to "motion situations" where speakers are moving most of the time (Fig.1). Audio files are named in function of the speakers characteristics (for more details see [8]). These files have been resampled down to 8 Khz (for simulating a telephone channel) and randomly divided into three sets: training (80%), validation (10%) and test (10%). The feature analysis has been performed over the training set.

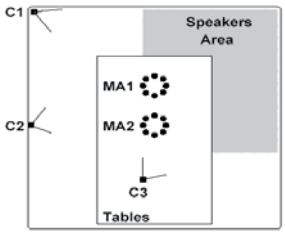


Figure 1. *MA1 and MA2 8-microphone circular array. See Speakers Area. This figure has been obtained from [8].*

3. PERCENTAGE OF CHANGES FOR THE BEST MIXTURE IN A HMM-BASED VAD SYSTEM

This section presents a study about the discrimination power between near-field vs. far-field speech using as feature the percentage of times the best mixture (in a maximum likelihood sense) of a speech HMM model in a HMM-based VAD change across a set of successive frames. Our VAD system uses two one-state HMMs (noise and speech models) including 200 Gaussian mixtures. This high number of components in the Gaussian mixture introduces more mixture variability producing higher frame-to-frame best mixture variability for multispeaker signals. The VAD system uses a MFCC vector (generated from a 12 Mel filter-bank analysis) formed by the first 8 cepstrum coeficients, normalized energy and delta energy. The HMMs models have been trained by means of Baum Welch re-estimation (ec. 1).

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\gamma_s} \quad (1)$$

where M_S is the number of mixture components in stream s, c_{jsm} is the weight of the m´th component and $N(\bullet;\mu;\Sigma)$ is a multivariate Gaussian with mean vector μ and covariance matrix Σ.

Initially the best mixture was selected after applying its mixture weight but small Gaussian variability was found. In order to increase this variability, mixtures weights were removed from the best mixture computation. In this case, a lot of candidates of winner gaussians were obtained when processing all frames. The measure we propose is the percentage of changes of the best Gaussian along N consecutive frames.

Figures 2 and 3, show the distribution of the percentage of changes considering N=100 and N=1000 frames respectively. Only speech frames are considered in this study. The noisy frames are discarded.

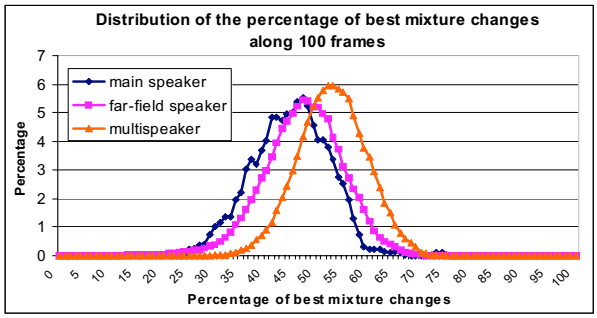


Figure 2. *Distribution of the percentage of changes considering N=100 frames.*

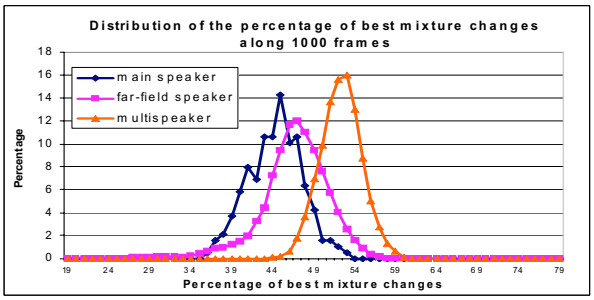


Figure 3. *Distribution of the percentage of changes considering N=1000 frames.*

As it is shown in figures 2 and 3, the percentage of best mixture changes is higher for speech coming from several speakers at the same time. This feature can discriminate very well between main speaker and multispeakers voices. In this case, the error is lower than 26% and 10% for 100 and 1000 frames respectively. When the number of frames considered for the percentage computation (N) increases from 100 to 1000 the measure is better estimated and the

discrimination power increases. On the other hand, the discrimination power between main speaker and far-field speaker voices is not good enough. Anyway, it is possible to see a higher percentage of changes for far-field speech.

4. MAHALANOBIS DISTANCE BETWEEN MFCCs

This feature consists of computing the Mahalanobis distance between MFCC vectors obtained from consecutive speech frames. Every vector contains the first 8 MFCC coeficients, normalized energy and delta energy. Mahalanobis distance, ec. (2), is used to evaluate the similarity between multidimensional random variables:

$$d_M(\vec{x}_{i.};\vec{x}_j) = \sqrt{(\vec{x}_{i.} - \vec{x}_j)S^{-1}(\vec{x}_{i.} - \vec{x}_j)^t} \quad (2)$$

where S is the covariance matrix of the variable vector $(x_1, x_2, ..., x_k)$. The distributions of Mahalanobis distance between consecutive frames for the main speaker, far-field speaker and multi-speaker speech are shown in figure 4. Figure 4 shows the histogram of the Mahalanobis distances between consecutive frames. As it is shown, main speaker speech presents lowest distance while multi-speaker presents the highest ones. At this point, the analysis were extended to groups of N frames, considering N=50 and N=500 frames. Again only speech frames were considered and noisy frames were discarded. In this process, the minimal distance along N consecutive frames is computed. Figures 5 and 6 shows the distributions of the minimal distance.

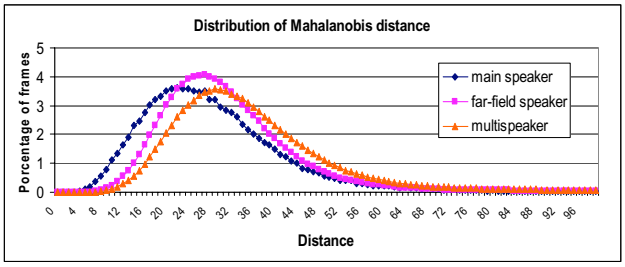


Figure 4. *Distribution of Mahalanobis distance distributions for main speaker, far-field speaker and multi-speaker speech.*

As it is shown in figures 5 and 6, the minimal distance along the N frames is higher for speech coming from several speakers at the same time. This feature can discriminate very well between main speaker and multispeakers voices. In this case, the classification error is lower than 24% and 14% for 50 or 500 frame segments respectively. When the number of frames considered for the minimal computation (N) increases from 50 to 500 the minimum is better estimated and the discrimination power increases.

The discrimination power between main speaker and far-field speaker voices with this feature is better compared to the previous feature. In this case, errors are lower than 35% and 27% for 50 or 500 frames. Other related measures, like the maximum, average, variance or kurtosis of the Mahalanobis distance, were also tested, but only the minimum distance showed an interesting relationship with the voice type. We think this is due to the fact that a low minimum distance is obtained during stationary speech zones: very infrequent in far-field and multi-speaker speech.

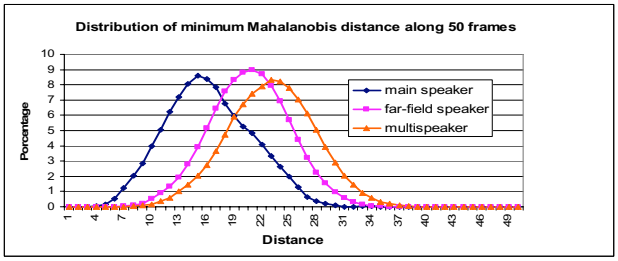


Figure 5. *Distribution of minimum Mahalanobis distance considering N=50.*

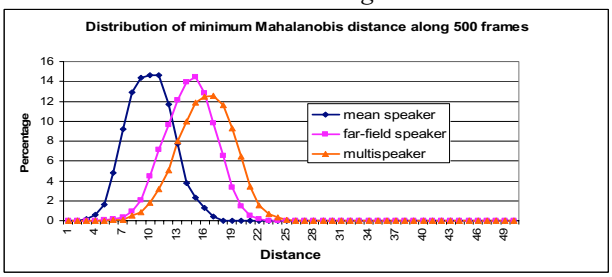


Figure 6. *Distribution of minimum Mahalanobis distance considering N=500.*

5. MAXIMUM AUTO-CORRELATION OBTAINED WHEN COMPUTING THE PITCH

In this case, the study was focused on the behavior of the auto-correlation values when computing the pitch at every frame. Considering only voice frames, the maximum of auto-correlation in pitch regions is considered. Fig. 7 presents the maximum auto-correlation distributions for main speaker, far-field speaker and multispeaker speech.

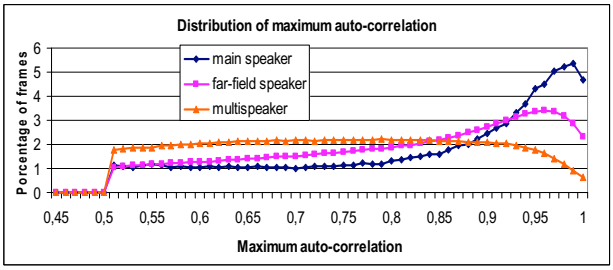


Figure 7. *Distribution of maximum auto-correlation for main speaker, far-field speaker and multispeaker.*

Fig 7. shows very different behaviors for the maximum auto-correlation value in the three cases, specially for auto-correlation values higher than 0.9. There are many more frames in the case of the main speaker speech and very few in the case of multi-speaker speech. So after considering this effect, the percentage of frames (along N frames) with a maximum auto-correlation higher than 0.9 was computed for the three types of speech. Figures 8 and 9 show the distributions of the percentage of maximum auto-correlation values higher than 0.9 for

main speaker, far-field speaker and multi-speaker speech along 50 and 500 frames.

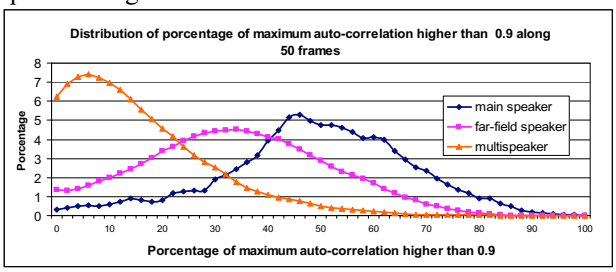


Figure 8. *Distribution of porcentage of maximum auto-correlation higher than 0.9 for main speaker, far-field speaker and multi-speaker N=50.*

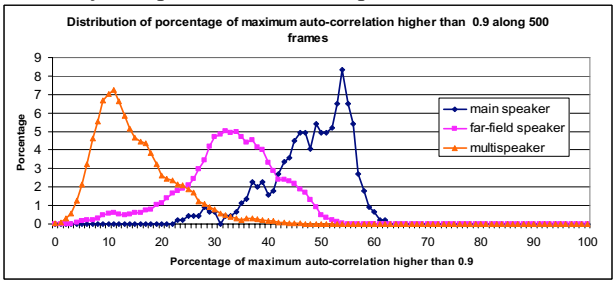


Figure 9. *Distribution of porcentage of maximum auto-correlation higher than 0.9 for main speaker, far-field speaker and multi-speaker N=500.*

As it is shown in figures 8 and 9, the percentage along the N frames is lower for speech coming from several speakers at the same time. This feature can discriminate very well between main speaker and multispeakers voices. In this case, the error is lower than 15% and 3.5% for considering 50 or 500 frames respectively. When the number of frames considered (N) increases from 50 to 500 the percentage is better estimated and the discrimination power increases. The discrimination power between main speaker and far-field speaker voices is better compared to the previous two features. In this case, classification errors are lower than 33.5% and 19% for 50 and 500 frames respectively.

6. CONCLUSIONS

This paper presents new successful features for improving VAD (Voice Activity Detection) when main speaker speech is mixed with far-field and multi-speaker speech. Generally, these features can be used to improve the behavior of any application in which it is necessary to discriminate the main speaker speech from far-field speech and multi-speaker speech. This study has been done with the Av16.3 speech database but the audio files have been resampled to 8Khz in order to simulate a telephone channel. The first feature proposed has been the percentage of changes of the mixture with the maximum likelihood, considering a VAD system based on HMMs. Results show better performance for multi-speaker speech rejection.

The second one was the Mahalanobis distance between the MFCCs of consecutive speech frames. In this case, the results were better than previous feature ones. Error between main speaker speech and multi-speaker speech was lower than 24% and 14% for considering 50 or 500 frames respectively. On the other hand, comparing main speaker speech vs. far-field speech, classification error was lower than 35% and 27% for 50 and 500 frames.

Finally, the best feature has been the maximum auto-correlation value obtained when computing the pitch at every frame. This feature can discriminate very well between main speaker and multi-speaker voices. Although some measures over this maximum has been processed, porcentage of frames with a maximum of auto-correlacion value higher than 0.9 is the one which gets the best results. In this case, the error was lower than 15% and 3.5% for considering 50 or 500 frames respectively. When comparing main speaker speech and far-field speech, classification errors are lower than 33.5% and 19% for 50 and 500 frames.

For all the features, when the number of consecutive frames considered for feature computation increases, the discrimination power increases. It is important to remark that a good performance for real time applications is obtained for the second and third features whose behavior when considering 50 frames is very good.

7. REFERENCES

[1] Petropulu, A. P., and Subramaniam, S.,"*Cepstrum based deconvolution for speech dereverberation*", IEEE Trans. Speech and Audio Proc., pp. 9-12, 1994.

[2] Nakatani, T. and Miyoshi, M.,"*Blind dereverberation of single channel speech signal based on harmonic structure*", pp. 92-95, ICASSP 2003.

[3] Ohta, K. and Yanagida, M., "*Single channel blind dereverberation based on auto-correlation functions of frame-wise time sequences of frequency components*", Iwaenc 2006 – Paris – September 12-14, 2006.

[4] Bees, D., Kabal, P., and Blostein, M., "*Application of complex cepstrum to acoustic dereverberation*", Proc. Biennial Symp. Commun. (Kingston, ON), pp. 324-327, June 1990.

[5] Yegnanarayana, B., Mahadeva Prasana, S. R., Duraiswami, R. and Zontkin, D., "*Processing of Reverberant Speech for Time-Delay Estimation*", IEEE Trans. Speech and Audio Proc., pp. 1110-1118, vol. 13, nº 6, 2005.

[6] Cournapeau, D. And Kawahara, T., "*Evaluation of Real-Time Activity Detection based on High Order Statistics*", pp. 2945-2948, Interspeech 2007.

[7] Ramírez, J., Segura, J., Benítez, C. and Rubio, A., "*A New Kullback-Leibler VAD for Speech Recognition in Noise*", IEEE Signal Proc., vol 11, nº 2, pp. 266-269, 2004.

[8] AV16.3: *An Audio-Visual Corpus for Speaker Localization and Tracking.*

SVM BASED POSTERIOR PROBABILITIES FOR SYLLABLE CONFIDENCE ANNOTATION

Daniel Bolanos [1, 2], Wayne Ward [1], and Javier Tejedor[2]

[1] Center for Spoken Language Research, University of Colorado at Boulder, USA
[2] HCTLab-Escuela Politécnica Superior, Universidad Autónoma de Madrid, SPAIN
{bolanos, whw}@cslr.colorado.edu, e-mail: {javier.tejedor}@uam.es

ABSTRACT

In this paper we present a mechanism to incorporate support vector machine (SVM) based phone posterior estimates in the computation of posterior probabilities over syllable lattices. A continuous speech recognizer is used to generate a syllable lattice. Using the state alignment information associated to each syllable in the lattice, SVM-based posteriors are calculated for each phone and then combined to obtain syllable posterior probabilities. Finally, these probabilities are incorporated into the computation process of posterior probabilities over syllable graphs using the forward-backward algorithm.

Experimental results show that the SVM-based confidence measures computed over syllable lattices can substantially reduce the classification error rate of HMM-based state-of-the-art confidence measures.

Index Terms— Confidence annotation, machine learning, support vector machines, posterior probabilities, syllable lattices

1. INTRODUCTION

The rapid development of speech technology in the recent years has enabled the development of a wide variety of speech applications. Unfortunately speech recognition results are still far from perfect, which, for practical applications, requires the use of confidence annotation techniques that help in the detection of misrecognized words. In our previous work [1] we have introduced a children's speech reading tracker that makes use of a syllable rejection module to classify syllables in a reference string as correctly or incorrectly read. In this article we have tried to improve the performance of this rejection module by integrating SVM-based posterior estimates in the computation of posterior syllable probabilities over syllable lattices.

Recently, SVM-based classifiers have been successfully applied for N-best lists rescoring at the output of a conventional HMM decoder [2]. These classifiers produce posterior class probability estimates that can also be used to generate confidence annotation labels. Previous work has shown [3] that confidence measures based on word posterior probabilities estimated over word graphs outperform alternative confidence measures [4] such as acoustic stability and hypothesis density. In this article we try to incorporate syllable posterior probability estimates obtained from SVM classifiers into a posterior probability computation procedure over syllable lattices.

In section 2 we present three different SVM-based phonetic classifiers and show how they can be used to generate syllable posterior probabilities. In the next section we define three different confidence measures resulting from the integration of those syllable posteriors into the computation of posterior probabilities over syllable graphs. Finally, we compare the performance of the confidence measures proposed with an HMM-based confidence measure taken as baseline and present our conclusions.

2. SVM FOR SYLLABLE CONFIDENCE ESTIMATION

An SVM learns the decision boundary between samples belonging to two classes by mapping the training sample vectors into a higher dimensional space and then determining an optimal separating hyper-plane [5]. When SVMs are used in classification tasks for speech processing applications it is necessary to map the margin or distance they produce to a posterior class probability. This can be done by the use of a sigmoid [6], where the parameters A and B need to be estimated by cross-validation.

$$p(y=1 \mid f) = \frac{1}{1+e^{(Af+B)}} \tag{1}$$

In the context of syllable classification, this posterior probability can be used to express the probability that a sequence of speech frames belongs to a syllable class. In our case, due to the considerable number of syllables present in the speech corpora, we decided to use SVMs to calculate posterior phone probabilities and combine them to calculate syllable posterior probabilities. We do not attempt to model coarticulation.

2.1. Phonetic classification

We have proposed 3 different phonetic classifiers. The simplest one (2) consists of training an SVM classifier for each phone using speech features directly as training vectors. This way, the probability of a phone *ph* given the sequence of feature vectors $x_1^T = \{x_1, x_2, \ldots, x_T\}$ to which it is aligned, is estimated as the average of the posterior phone probabilities obtained from the SVM for each of its frames.

$$p_{frames}(ph \mid x_1^T) = \frac{1}{T}\sum_{t=1}^{T} p(ph \mid x_t) \tag{2}$$

In the second approach (3) we have tried to take advantage of the state alignment information produced by an HMM aligner to capture the time-varying structure of a phone, which is missing in the previous approach. We use the Sonic speech recognition system for the alignment [10]. Sonic uses 3-state HMM phone models. Feature vectors aligned to each state of a phone are

averaged and the resulting average vectors are concatenated to form a composite vector that is used to train the SVM classifier. The dimension of this vector is, consequently, three times the dimension of the original feature vectors. A similar approach was proposed in [2], but assigned a fixed percentage of the frames aligned with a phone to each state.

$$p_{segments}(ph \mid x_1^T) = p(ph \mid composite(x_1^T)) \quad (3)$$

The third approach (4) uses the phone temporal structure information while still using speech features directly as training vectors for the SVM classifier. The later consideration is important since, as we will see later, the averaging process carried out in the second approach prevents the reuse of SVM predictions across a lattice time frame. This procedure is more computationally expensive and significantly deteriorates the real time performance. Hence, three SVM classifiers are trained for each phone, each of them trained with the speech features from a different state.

$$p_{frames/states}(ph \mid x_1^T) = \frac{1}{3}\sum_{s=1}^{3}\frac{1}{T_s}\sum_{t=1}^{T_s} p(ph_s \mid x_t) \quad (4)$$

2.2. Syllable classification

As expressed in (5), syllable posterior probabilities are calculated by averaging phone posterior probabilities.

$$p(syl \mid x_1^T) = \frac{1}{N}\sum_{i=1}^{N} p(ph_i \mid x_{T_{i-1}+1}^{T_i}) \quad (5)$$

By using the three phonetic classifiers presented in section 2.1 as the phone posterior probability, we define the following posterior syllable probabilities: $p_{frames}(syl|x_1^T)$, $p_{segments}(syl|x_1^T)$ and $p_{frames/states}(syl|x_1^T)$, that can be used as syllable classifiers.

3. EXPERIMENTS FOR SYLLABLE CONFIDENCE ANOTATION

Experiments were carried out to evaluate the accuracy of the phonetic and syllable classifiers proposed.

3.1. Speech material

We present experimental results on the CU Read and Summarized Story Corpus [8]. We have selected speech belonging to first and second graders (a total of 171 and 57 speakers respectively) and partitioned it into a training set containing 9 hours of audio and a test set of about 2 hours of audio.

3.2. Training and parameter selection

For every speech segment present in the training set, 39-dimensional feature vectors, consisting of 12 Mel Frequency Cepstral Coefficients and energy plus first and second order derivatives, have been extracted. The children's speech corpora available is tagged at the word level only so phone boundaries are obtained using a Viterbi-based phonetic alignment against the transcriptions.

SVM classifiers are well suited for two-class separation tasks, however for n-class (n>2) separation tasks, like building a phonetic classifier, n SVM classifiers need to be trained. In this case we have selected a "one vs. all" approach in which up to three SVM classifiers [9] are trained for each of the 55 phonetic symbols used. For the training of each SVM, half of the data points (positive samples) belong to the actual class while the rest belong to the remaining classes (negative samples).

A radial basis function (RBF) kernel is used for which the parameters C (cost) and γ are estimated over the training set with a "grid-search" process using 5-fold cross validation.

3.3. Phonetic classification

The first experiment conducted evaluates the classification accuracy of the three phonetic classifiers proposed. For evaluating the classifiers we created a test set that contained 500 positive examples and a balanced set of 500 negative examples for each of the 55 phones used. In the test set, each instance is assigned a phone label (half of which are correct labels). For each of the classification algorithms, 55 classifiers (corresponding to one-vs-rest classifiers for each phone) were trained. For each instance, the classifier corresponding to the phone label for the instance is used to assign a probability of the phone given the data. This score is compared to previously trained thresholds (phone-dependent) to classify the phone occurrence as belonging to the phone class or not. Figure 1 shows the classification accuracy for each classification algorithm and every phone class.

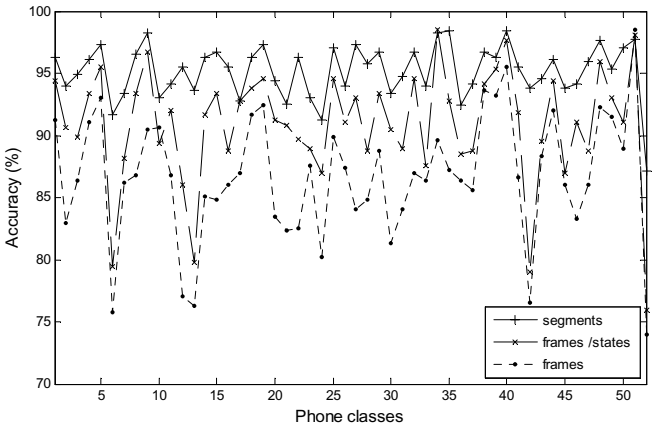


Figure 1. Phonetic classification accuracy for the three classifiers proposed

In table 1 we summarize the average classification results across the whole phonetic symbol set for the proposed classifiers. As expected, the segment-based approach yields the best classification accuracy, followed by the frame-based approach that makes use of state alignment information. An interesting detail observed is that the trained threshold used in the decision making process varies greatly between phones. This suggests that score distribution information for each phone could be incorporated in the syllable classifier for improved accuracy.

Approach	Average accuracy (%)
$P_{frames}(ph\|X)$	86.67
$P_{segmentss}(ph\|X)$	95.20
$P_{frames/states}(ph\|X)$	90.87

Table 1. Average classification accuracy across the phonetic symbol set

3.4. Best single path confidence annotation

In this experiment we compare the classification accuracy of the three syllable classifiers proposed in section 2.2. The experiment consists of running a decoding process using Sonic [10]. The single best scoring path is then annotated with posterior syllable probability estimates. In this experiment, the confidence annotation uses only the state alignment information of the best single path and no lattice information or language model probability is used. The Figure 3 shows a comparison of the syllable classifiers using Detection-Error Tradeoff curves that contain a plot of the false acceptance rate over the false rejection rate.

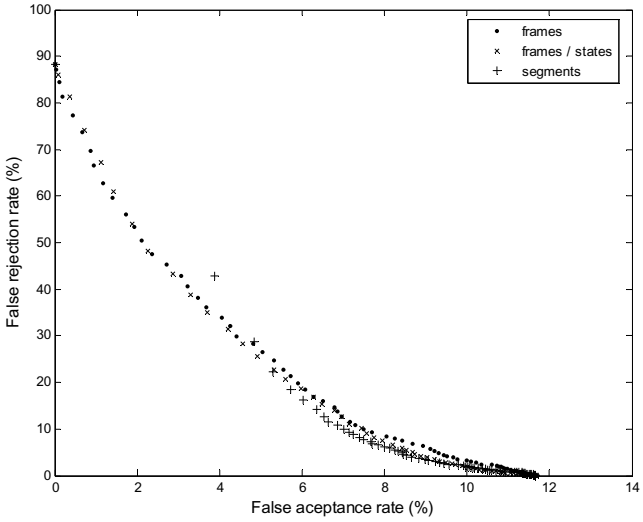


Figure 2. DET curves for lattice based posterior probabilities.

Surprisingly, despite the observed difference in classification performance of the three phonetic classifiers used as a basis for the confidence annotation, the curves depicted are very similar. This can be observed especially in the most relevant area of the graph, i.e. for FRR < 10. However, the segments-based classifier yields a slightly better accuracy for syllable confidence annotation.

4. COMPUTATION OF SYLLABLE POSTERIOR PROBABILITIES OVER SYLLABLE LATTICES

In this section we discuss the mechanism for incorporating the SVM based phone posterior estimates presented in section 2 in the computation of posterior probabilities over syllable lattices.

Initially we describe briefly the typical computation procedure of syllable posterior probabilities over syllable lattices. The posterior probability $p([syl;s,e]|X)$ for a syllable can be calculated as defined in (6) by summing up the posterior probabilities of all paths in the lattice of length M which contain the hypothesis $[syl;s,e]$. $[syl;s,e]$ is the syllable starting at time s and ending at time e, and $X=\{x_1,\ldots,x_T\}$ is the acoustic observation sequence against which it is aligned.

$$p([syl;s,e]\,|\,x_1^T) = \sum_{\substack{[syl;s,e]_1^M:\\ \exists n\in\{1,\ldots,M\}:\\ [syl_n;s_n,e_n]=[syl;s,e]}} \frac{\prod_{m=1}^{M} p(x_{s_m}^{e_m}\,|\,syl_m)^{\alpha}\, p(syl_m\,|\,syl_1^{m-1})^{\beta}}{p(x_1^T)} \tag{6}$$

Typically these posterior probabilities are calculated very efficiently over syllable graphs using the forward-backward algorithm as described, in the case of words, in [3] and [7]. This algorithm considers edges in the graph as HMM-like states, where emission probabilities are the HMM acoustic models scores and transition probabilities between links are obtained from the language model used.

We have proposed an alternative computation procedure (7) where the HMM acoustic scores for each syllable are substituted by the posterior syllable probabilities produced by the SVM syllable classifiers defined in section 2. We realize that this is replacing a quantity that represents P(observations | syllable) with a quantity that is a direct estimation of the posterior probability P(syllable | observations). We believe that these posterior syllable probabilities, given the equality assumption for the prior class probability made in the construction of the SVM classifiers that produce them, can still be effectively combined with language model probabilities in the computation of posterior probabilities over syllable graphs.

$$p([syl;s,e]\,|\,x_1^T) = \sum_{\substack{[syl;s,e]_1^M:\\ \exists n\in\{1,\ldots,M\}:\\ [syl_n;s_n,e_n]=[syl;s,e]}} \frac{\prod_{m=1}^{M} p([syl_m;s_m,e_m]\,|\,x_{s_m}^{e_m})^{\alpha}\, p(syl_m\,|\,syl_1^{m-1})^{\beta}}{p(x_1^T)} \tag{7}$$

In (6) and (7) α represents the acoustic score scaling factor while β represents the language model probability scaling factor. These parameters are necessary to compensate the different dynamic range of acoustic and language model scores, and need to be estimated over a cross-validation set independent from the test set. However, previous work [3] has demonstrated that posterior probabilities calculated as (6) or (7) do not produce satisfactory results. The reason is that the fixed starting and ending time frames of a hypothesis syllable strongly determine the paths involved in the calculation of the forward-backward probabilities. Usually, syllable hypotheses with similar starting and ending time frames represent the same syllable; it therefore makes sense to consider the summation of the posterior probabilities of these syllables as a confidence measure. For this reason we have used a confidence measure (8) proposed in [3] for which the posterior probability accumulation process is carried out over all the time frames of the hypotheses under consideration. After the accumulation process is done, the highest probability value is selected as a measure of confidence.

$$C([syl;s,e]) = \max_{e_{\max}\in\{s,\ldots,e\}} \sum_{[syl;s',e']:s'\le e_{\max}\le e'} p([syl;s',e']\,|\,x_1^T) \tag{8}$$

By substituting in (7) the three posterior syllable probabilities defined in section 2.2 and applying the probability accumulation process defined in (8) we define the following respective confidence measures: $C_{SVMframes}[syl;s,e]$, $C_{SVMsegments}[syl;s,e]$ and $C_{SVMframes/states}[syl;s,e]$. The performance of these confidence measures will be evaluated in section 5 against a baseline confidence measure computed combining expressions (6) and (8) and referenced in the following as $C_{HMM}[syl;s,e]$.

5. EXPERIMENTS FOR LATTICE BASED POSTERIOR PROBABILITIES

In this experiment we compare the performance of the three SVM-based confidence measures proposed in section 4 against an HMM-based one, also described in section 4, which constitutes the baseline. All these confidence measures make use of lattice information so all of them are applied after an initial step of syllable lattices generation using the SONIC continuous speech recognizer. For all of them the scaling factors α and β has been trained over a development set different than the test set. The experimental set-up is the same as described in section 3.

In addition to the use of DET curves, the metric selected to evaluate these confidence measures is the classification error rate (CER) defined as the number of incorrectly assigned tags (which comprises false acceptations and false rejections) divided by the total number of recognized syllables. In figure 3 we show the DET curves of the confidence measures $C_{SVMframes}[syl;s,e]$ and $C_{HMM}[syl;s,e]$ (the baseline). The reason for not depicting the other two SVM-based confidence measures ($C_{SVMsegments}[syl;s,e]$ and $C_{SVMframes/states}[syl;s,e]$) is that their respective DET curves almost completely overlap with the $C_{SVMframes}[syl;s,e]$ in the graph so cant be distinguished. However, the best confidence error rates (CER) for all of them are shown in Table 2.

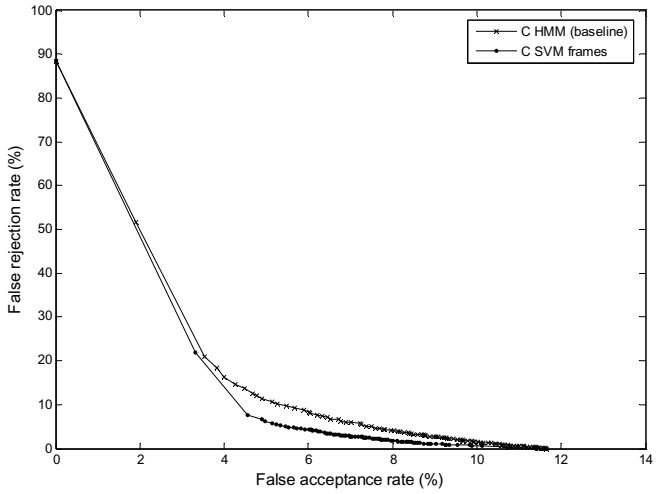


Figure 3. Detection-error tradeoff curves for lattice based posterior probabilities.

Confidence Measure	CER	Relative error reduction (%)
$C_{HMM}[syl;s,e]$ (baseline)	11.43	
$C_{SVMframes}[syl;s,e]$	9.68	15.16
$C_{SVMsegments}[syl;s,e]$	9.63	15.60
$C_{SVMframes/states}[syl;s,e]$	9.35	18.05

Table 2. Confidence error rates and relative error reduction respect to the baseline for the different confidence measures proposed.

As can be seen in Figure 3, the SVM-based confidence measures clearly outperform the HMM-based one used as baseline. In particular, the CER of the SVM-based approaches is at least 15% better than the baseline. Another interesting point is that, despite the considerable differences in classification accuracy observed in the phonetic classifiers (see Table 1) in which these confidence measures rely, their CER is very similar. Considering this similarity, in the context of a real world application, the $C_{SVMframes}[syl;s,e]$ is the most interesting one because the SVM-predictions at the frame level can be shared during the calculation of posterior class probabilities of overlapping phones in the lattice. Note that in the $C_{SVMframes}[syl;s,e]$ confidence measure computation is also possible to share a good number of SVM-predictions (the amount of predictions reused strongly depends on the lattice density). However for the computation of $C_{SVMsegments}[syl;s,e]$, due to the averaging process necessary for creating the composite vectors, no SVM-predictions can be reused so the real time performance deteriorates significantly when the lattice density is relatively high.

6. CONCLUSIONS AND FUTURE WORK

An effective way to incorporate SVM-based posterior probabilities in the computation of posterior probabilities over syllable graphs has been introduced. The new confidence measures presented clearly outperform existing ones in the experiments carried out. Moreover, these confidence measures can be used not only for rejection tasks but for lattice rescoring. Future work will be focused to the use of these confidence measures not only for building a syllable rejection module as part of our children's speech reading tracker but for increasing the information available in the algorithm we are currently using for aligning syllable lattices against multiple pronunciations graphs of syllables.

7. REFERENCES

[1] D. Bolanos, W. Ward, S. Van Vuuren, J. Garrido, "Syllable Lattices as a Basis for a Children's Speech Reading Tracker", in *InterSpeech*, Antwerp, Belgium 2007.

[2] Applications of Support Vector Machines to Speech Recognition A. Ganapathiraju, J. E. Hamaker, J. Picone, *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2348-2355, 2004

[3] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, "Confidence Measures for Large Vocabulary Continuous Speech Recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 9, pp. 288–298, March 2001.

[4] T. Kemp and T. Schaaf, "Estimating confidence using word latices," in *Proc. 5th Eur. Conf. Speech, Communication, Technology 1997*, Rhodes, Greece, Sept. 1997, pp. 827–830.

[5] Vapnick, V. T*he Nature of statistical Learning Theory.* Springer-Verlag, New York, 1995.

[6] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. Cambridge, MA: MIT Press, 1999.

[7] K. Hacioglu and W. Ward, "A Concept Graph Based Confidence Measure", in *International Conference of Acoustics, Speech, and Signal Processing*, Orlando-Florida, USA, 2002

[8] R. Cole and B. Pellom. University of Colorado read and summarized story corpus. Technical Report TR-CSLR-2006-03, University of Colorado, 2006.

[9] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines, 2001.

[10] B. Pellom, "Sonic: The University of Colorado Continuous Speech Recognizer", Technical Report TR-CSLR-2001-01, CSLR, University of Colorado, March 2001.

SESIÓN ORAL 4
TRADUCCIÓN AUTOMÁTICA

DERIVING BENEFIT FROM A GENERALIZED SYNTAX-BASED REORDERING

*Maxim Khalilov, José A.R. Fonollosa**

TALP Research Center
Universitat Politècnica de Catalunya
Campus Nord UPC, 08034,
Barcelona, Spain

Mark Dras

Centre for Language Technology
Macquarie University
North Ryde NSW 2109,
Sydney, Australia

RESUMEN

In this study we describe a syntax-based word reordering technique for n-gram-based statistical machine translation (SMT). The proposed distortion model operates with generalized unlexicalized rules and aims to order source language words so that translation is close to monotonic, simplifying the translation process. In the final step, we apply a translation units blending strategy, combining bilingual tuples extracted from the parallel corpora with monotone and reordered source parts.

Experiments are reported on the BTEC corpus from tourist domain for the Arabic-English translation task, the proposed tuples blending technique significantly outperformes the monotone system.

1. INTRODUCTION

The word disparity problem between source and target languages is a crucial point for many modern SMT systems. Several researchers [1, 2] consider the reordering model to hold great scope for translation quality improvement, and even as a bottleneck bounding further SMT progress. At the same time, there is a controversy about whether a statistical system can benefit from syntactic information, expressed in form of Part-of-Speech (POS) tags, shallow or dependency parse trees.

Though, the word class-based reordering patterns are part of Och's Alignment Template system [1], the classical phrase-based approach does not entirely solve the reordering problem. This problem leads to particularly bad translation when dealing with languages having distinct word orders and linguistic typology. An example of such language pair is Arabic and English: apart from a difference in verbal morphology and the presence of enclitics, they have distinct language topology schemes (VSO for Arabic and SVO for English). Where a monotone translation approach in many cases is not able to deal with such a reordering disparity, a constituent tree structure can be used.

*This work has been funded by the Spanish Government under grant TEC2006-13964-C03 (AVIVAVOZ project) and by Generalitat de Catalunya under project TECNOPARLA.

There have already been some efforts to solve this problem both in purely statistical way or involving additional informational sources. The state-of-the-art phrase-based SMT system Moses[1] implements a distance based distortion model [3] as does a word alignment-based MSD (Monotone, Swap and Discontinuous) reordering model as shown in [4].

A linguistically motivated reordering model employing a monotonic search graph extension was proposed in [5]. In [2] another method of word reordering for N-gram-based MT systems was introduced: a monotone sequence of source words is translated into the reordered sequence using the well established mechanism of SMT.

A set of hand-crafted reordering rules demonstrated a significant improvement for German to English translation as shown in [6]. In [7] the authors present a hybrid system for French-English translation, based on the automatically deriving rewrite patterns extraction from a parse tree and phrase alignments. Inspired by this idea we intend to apply a subtree target-to-source mapping as was done in [8], where a two-side subtree transfer was introduced as a part of a syntax-driven SMT. Afterwards, the translation task, realized by a n-gram-based system is reformulated to translate from the reordered source language, that lead to a mutual word order monotonization, shorter translation units and improved translation.

The rest of the paper is organized as follows: Section 2 outlines the n-gram-based SMT system. Section 3 introduces the syntax-based reordering. In Section 4 we present the results and contrast them with an alternative reordering techniques and Section 5 presents the conclusions.

2. NGRAM-BASED SMT

The n-gram-based approach regards translation as a stochastic process maximizing the joint probability $p(f, e)$, leading to a decomposition based on bilingual n-grams, which we call *tuples*, that are extracted from a word-to-word alignment (performed with GIZA++ tool[2]). Tuples are extracted according to the following constraints [9]:

[1] www.statmt.org/moses/
[2] http://code.google.com/p/giza-pp/

- a monotonic segmentation of each bilingual sentence pair is produced
- no word in a tuple is aligned to words outside of it
- no smaller tuples can be extracted without violating the previous constraints

Figure 1 shows an example of tuple monotonic extraction (*regular* technique resulting in one tuple), contrasted with the *unfolding* technique (resulting in three tuples), that allow producing a different bilingual n-gram language model with reordered source words.

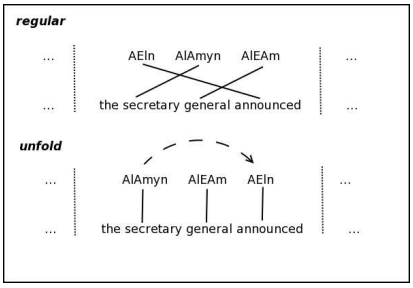


Figura 1. *Example of tuples extraction.*

The Ngram-based translation system implements a log-linear model in which a foreign language sentence $f_1^J = f_1, f_2, ..., f_J$ is translated into another language $e_1^I = e_1, e_2, ..., e_I$ by searching for the translation hypothesis $\hat{e}_1^I$ maximizing a log-linear combination of several feature models [9].

A *translation model (TM)* approximates the joint probability between source and target languages capturing bilingual context, as shown in equation 1:

$$p(S,T) = \prod_{k=1}^{K} p((\tilde{s},\tilde{t})_k | (\tilde{s},\tilde{t})_{k-N+1}, ..., (\tilde{s},\tilde{t})_{k-1}) \quad (1)$$

where s refers to source, t to target, and $(\tilde{s},\tilde{t})_k$ to the k^{th} tuple of a given bilingual sentence pair segmented in K tuples.

The rest of the system models are: a *target language model*, a *POS target language model*, a *word bonus model*, a *source-to-target lexicon model* and a *target-to-source lexicon model*. For more details refer to [9].

We used the MARIE beam-search decoder [10] allowing for efficient pruning of the search space, threshold pruning, histogram pruning and hypothesis recombination. Given the development set and references, the log-linear combination of weights was adjusted using a simplex optimization method (with the optimization criteria of the highest BLEU score) and an n-best re-ranking.

3. SYNTAX-BASED REORDERING

In this study we simulate a situation when the reordering system has access to both the source and target language shallow parsers using word alignment intersection as a 'bridge' between two languages. We used the Stanford Parser as a parsing engine[3] [11] and the Arabic and English Penn Treebank sets (26 POS/23 constituent categories for Arabic Treebank and 48 POS and 14 syntactic tags for English Treebank).

Syntax-based reordering as described in this paper operates with a Context-Free Grammar (CFG), where each branch of the parse tree is represented as follows:

$$X \rightarrow \langle N, T, R, S \rangle \quad (2)$$

where N refers to a set of constituents and POS tags, T is a set of terminals (lexicon), R stands for a mapping from N to $(T \bigcup N)^*$ of the form $N_i \rightarrow \gamma$ (γ is a sequence of terminals and non-terminals) and S is the start variable.

Reordering patterns are expressed in the form *NP@0 VP@1* $\rightarrow$ *VP@1 NP@0 p1*, that means that a sequence of constituents *NP@0 VP@1* should be reordered like *VP@1 NP@0* with probability `p1`. Note that here the number of constituents indicates the order of their appearance in the source part of the pattern.

3.1. Rules extraction

The reordering rule extraction procedure consists of the following steps:

Step 1 align the monotone corpus and find the intersection of src-to-trg and trg-to-src word alignments (construct the projection matrix P);

Step 2 parse the source and the target parts of the parallel corpus;

Step 3 convert the parse trees to the CFG form;

Step 4 extract reordering patterns from the parallel non-isomorphic CFG-trees basing on the word alignment intersection and considering POS and constituents equally;

Step 5 estimate and normalize the number of reordering pattern instances.

Figure 2 shows an example of the rule extraction procedure (Step 4) for a parallel sentence

Arabic: *h*A hW fndq +k*
English: this is your hotel

Given two parse trees and word alignment intersection expressed in form of projection matrix

$$P = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

[3]Generally speaking, the source and targets formal grammars, as well as the parsing mechanisms can differ.

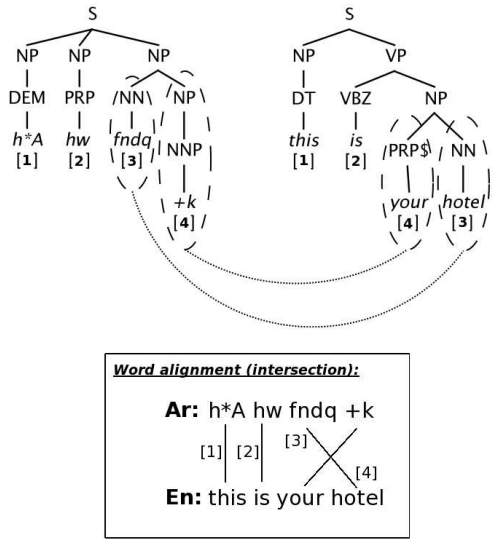


Figura 2. *Rules extraction step.*

the directly extracted reordering rule is *NN@0 NP@1* → *NP@1 NN@0* and since the "NP" node leads to the leaf *"+k"* through the "NNP" POS tag, one more unlexicalized rule can be induced: *NN@0 NNP@1* → *NNP@1 NN@0*. It is worth noticing that the left side of the reordering pattern is always monotone and the right side can be monotone or reordered.

If a word that is aligned in only one direction (source to target or target to source) appears in the branch that is considered as a candidate to be involved into a reordering pattern, it does not exert influence on the alignment projection matrix.

3.2. Organizing reordering rules

Once the list of reordering patterns is extracted, they are organized following the strategy similar to the one proposed in [7] for generalized rules. All the rules that appear less than k times are directly discarded (in experiments we used the threshold $k = 3$). A probability of alternative patterns is estimated basing on absolute counting of their appearance in the training corpus and the most probable rules are stored.

Ambiguous rules are pruned out according to the higher probability principle, for example, for the pair of patterns *NP@0 VP@1 ->VP@1 NP@0 p1*, *VP@0 NP@1 ->NP@1 VP@0 p2*, leading to the recurring contradiction, one rule will be removed depending on the ratio $p2/p1$).

Finally, the reordering table (analogous to the "r-table" as stated in [8]) is a set of POS- and constituent-based patterns allowing for reordering and monotone distortion.

3.3. Source-side monotonization

Rules application is performed as a bottom-up parse tree traversal applying the longest possible rule, i.e. among a set of nested rules, the rule with a longest left-side covering is selected (e.g. in case of *NN JJ RB* sequence appearance and two reordering rules presence *NN@0 JJ@1 ->...* and *NN@0 JJ@1 RB@2 ->...*, the former pattern will be applied).

Figure 3 shows the example of the reordered source-side parse tree with the applied pattern *NN@0 NNP@1 ->NNP@1 NN@0*. The resulting Arabic sentence is

```
h*A hW +k fndq
```

that more closely matches the order of the target language and reflects *possessive pronoun - noun* typical English word order.

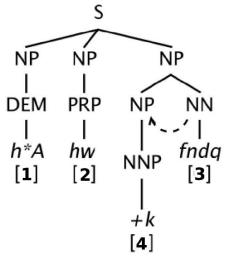


Figura 3. *Reordered source-side parse tree.*

3.4. Tuples blending

In terms of this study, we operate exclusively with generalized (i.e. unlexicalized) reordering rules, that along with improved translation units, cause errors induced by a certain number of grammatical exceptions which can be easily found in any language. Therefore, after the corpus with reordered source part is aligned, two sets of tuples are extracted basing on the reordered and monotone alignment matrices. In the final stage of the translation model construction, the bilingual units from these sets are combined following the criterion of maximizing the number of tuples at the sentence level. This technique entails more tuples involvement into TM contruction that provides better bilingual generalization (shorter translation units have higher probability of appearance in the translanting corpus than the longer ones).

4. EXPERIMENTAL SETTINGS, RESULTS AND COMPARISON WITH UNFOLDING METHOD

The experiments were performed on the BTEC'08 corpus from the tourist domain. A basic corpus statistics can be found in table 1.

The BLEU score obtained on the development set (489 lines, $3{,}7K$ running words and 6 reference translations)

	Arabic	English
Sentences	23.7 K	23.7 K
Words	166.0 K	183.9 K
Average sentence length	7.75	6.99
Vocabulary	10.8 K	6.8 K

Tabla 1. Basic statistics of the BTEC training corpus.

as the final point of the simplex optimization procedure and the translation results done on the test set (500 lines, $4{,}1K$ running words and 16 references) are summarized in table 2. We consider four translation systems: *monotone* and *reordered* configurations that correspond to the systems involving the parallel corpora with monotone and reordered source parts, respectively; a *blending* model as described in subsection 3.4; and the alternative *UC* method, that include the unfold algorithm of tuples extraction and constrained distance-based distortion model used on the decoding step (as described in [12]).

	dev BLEU	test BLEU	# tuples
Monotone	40.55	43.78	135.855
Reordered	41.05	45.15	143.934
Blending	43.20	47.92	170.572
UC	43.61	47.46	163.755

Tabla 2. Summary of the experimental results.

For the tuples *blending* configuration, about 40 % of the tuples came from the system with reordered source part. Curiously, more tuples were generated by this system than by *unfolded* algorithm (the number of bilingual units generated by the former system is the maximum theoretical possible with invariable alignment). We explain this phenomena by several "noisy" tuples generated by the reordered system under conditions of a lack of training material.

In terms of BLEU score, the unfolded and the combined reordered-monotone system demonstrate comparable performance significantly outperforming both the monotone and the syntactically reordered SMT systems.

5. CONCLUSIONS AND FUTURE WORK

The proposed syntactically motivated reordering coupled with the bilingual units blending method shows competitive performance comparing with an alternative reordering method on the small Arabic-English corpus preserving potential power of fully or partially lexicalized reordering rules using. However, more profound analysis of generated bilingual units and their impact on the translation quality is needed and will be done in the near future.

6. BIBLIOGRAFÍA

[1] F. J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, y D. Radev, "A Smorgasbord of Features for Statistical Machine Translation," in *Proceedings of HLT/NAACL04*, 2004, pp. 161–168.

[2] M. R. Costa-jussà y J. A. R. Fonollosa, "Statistical machine reordering," in *Proceedings of the HLT/EMNLP 2006)*, 2006.

[3] Ph. Koehn, F. J. Och, y D. Marcu, "Statistical phrase-based machine translation," in *Proceedings of the HLT-NAACL 2003*, 2003, pp. 48–54.

[4] C. Tillmann y T. Zhang, "A localized prediction model for statistical machine translation," in *Proceedings of the 43rd Annual Meeting on ACL 2005*, 2005, pp. 557–564.

[5] J. M. Crego y J. B. Mariño, "Improving statistical MT by coupling reordering and decoding," *Machine Translation*, vol. 20(3), pp. 199–215, 2007.

[6] M. Collins, Ph. Koehn, y I. Kučerová, "Clause restructuring for statistical machine translation," in *Proceedings of the 43rd Annual Meeting on ACL 2005*, 2005, pp. 531–540.

[7] F. Xia y M. McCord, "Improving a statistical mt system with automatically learned rewrite patterns," in *Proceedings of the COLING 2004*, 2004.

[8] K. Imamura, H. Okuma, y E. Sumita, "Practical approach to syntax-based statistical machine translation," in *Proceedings of the 10th Machine Translation Summit (MT Summit X)*, 2005, pp. 267–274.

[9] J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. A. Fonollosa, y M. R. Costa-juss, "N-gram based machine translation," *Computational Linguistics*, vol. 32, no. 4, pp. 527–549, 2006.

[10] J. M. Crego, J. B. Mariño, y A. de Gispert, "An ngram-based statistical machine translation decoder," in *Proceedings of INTERSPEECH05*, 2005.

[11] D. Klein y C. Manning, "Accurate unlexicalized parsing," in *Proceedings of the 41st Annual Meeting of the ACL 2003*, 2003, pp. 423–430.

[12] M. R. Costa-jussà, J. M. Crego, A. de Gispert, P. Lambert, M. Khalilov, J. A. Fonollosa, J. B. Mariño, y R. E. Banchs, "TALP phrase-based system and TALP system combination for IWSLT 2006," in *Proceedings of the IWSLT 2006*, 2006, pp. 123–129.

INCORPORACIÓN DE INFORMACIÓN SINTÁCTICO-SEMÁNTICA EN LA TRADUCCIÓN DE VOZ A LENGUA DE SIGNOS

B. Gallo, R. San-Segundo, J.M. Lucas, R. Barra, F. Fernández, L.F. D'Haro
Grupo de Tecnología del Habla. Universidad Politécnica de Madrid.
E.T.S.I. Telecomunicación. Ciudad Universitaria SN 28040 Madrid
lapiz@die.upm.es

RESUMEN

Este artículo presenta un conjunto de experimentos para evaluar la mejora obtenida cuando se incorpora información sintáctico-semántica en la traducción estadística de voz a lengua de signos. La traducción se realiza utilizando dos alternativas tecnológicas: la primera basada en modelos de subsecuencias de palabras y la segunda basada en traductores de estados finitos ("FST"). En cuanto a la evaluación de dichos resultados, se utilizan varias métricas, como WER (tasa de error de palabras), BLEU y NIST. Las pruebas realizadas incluyen experimentos con las frases de referencia en castellano y Lengua de Signos y con frases obtenidas del reconocedor de voz. Para evaluar la mejora obtenida se muestran los resultados con y sin información sintáctico-semántica. Los mejores resultados se obtuvieron con la solución de traductores de estados finitos con unas tasas de error de 26,06% para las frases de referencia y de 33,01% para las salidas del reconocedor cuando se incorpora información sintáctico-semántica.

1. INTRODUCCIÓN

Con la realización de este trabajo se pretende evaluar la incorporación de información sintáctico-semántica en una plataforma de traducción capaz de transformar, en base a un conjunto de modelos probabilísticos, frases de castellano a Lengua de Signos Española (LSE). Con estos experimentos se pretende mejorar una herramienta de traducción muy útil para las personas sordas puesto que el coste de un intérprete signante (que conoce la Lengua de Signos) es muy elevado. A la vez supone una aportación importante para las personas sordas prelocutivas (aquellas que se quedaron sordas antes de poder hablar), ya que su capacidad de comprensión del castellano escrito es muy inferior al resto.

Este artículo se centra en la mejora del módulo de traducción basado en métodos estadísticos. Estos métodos son un paradigma de traducción automática donde se generan traducciones en base a modelos estadísticos y de teoría de la información cuyos parámetros se obtienen del análisis de corpus de textos bilingües (documentos que constituyen la base de datos con pares de frases castellano-LSE). Con estos experimentos se pretende mejorar los resultados anteriormente presentados en [1].

Para la realización de los experimentos se dispuso de unos textos en castellano y en LSE con frases típicas que un funcionario de la administración pronuncia al atender a una persona en el servicio de solicitud o renovación del DNI.

2. ARQUITECTURA DEL SISTEMA

El sistema completo está formado por tres módulos: el del reconocedor de voz, el módulo de traducción estadística y finalmente, la representación de los signos mediante un avatar. Se muestra a continuación el diagrama del sistema:

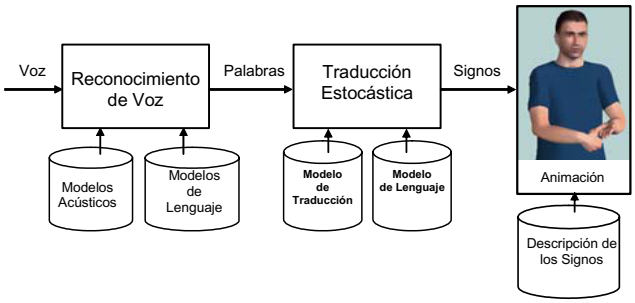


Figura 1. Arquitectura completa del sistema

El reconocedor del habla realiza la conversión del lenguaje natural (habla continua) a una secuencia de palabras basándose en un modelo del lenguaje y varios modelos acústicos.

En segundo lugar, el módulo de traducción estadística consiste en un algoritmo de búsqueda dinámica que utiliza un modelo estadístico para obtener la mejor secuencia de signos resultado de la traducción de una secuencia de palabras obtenidas del reconocedor de voz. Este modelo integra información de dos tipos de probabilidades: la probabilidad de traducción, que recoge información sobre qué palabras se traducen por qué signos y la probabilidad de secuencia de signos, que aporta información sobre qué secuencias de signos son más probables en la LSE.

El último módulo corresponde al avatar en 3D, que se encarga de la representación de los signos

provenientes de la traducción estadística. El avatar utilizado es "VGuido" del proyecto eSIGN [2].

3. TRADUCCIÓN BASADA EN SUBSECUENCIAS DE PALABRAS

La traducción estadística basada en modelos de subsecuencias (o subfrases) requiere la obtención de un modelo de traducción a partir del alineamiento entre las palabras de las lenguas origen y destino utilizando un corpus paralelo. Después del alineamiento de palabras se extraen y puntúan las subsecuencias de palabras que formarán el modelo de traducción. Además, es necesario generar un modelo de lenguaje de la lengua destino. En la fase de traducción, dada una frase de entrada se obtiene la secuencia de signos, que luego se evalúa para calcular los aciertos y fallos en la traducción. La arquitectura completa se muestra en la figura 2.

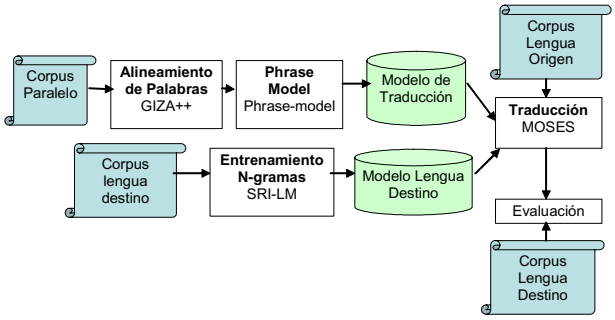


Figura 2. Traducción basada en subsecuencias

3.1 Generación de modelos

En primer lugar debe crearse el Modelo de Lenguaje de la lengua destino y el Modelo de Traducción (a partir de un corpus paralelo tanto en lengua origen (LO) como destino (LD)). El problema de la traducción se centra en conocer la probabilidad p(d|o) de que una cadena o de LO genere una cadena d en LD. Estas probabilidades se calculan utilizando técnicas de estimación de parámetros a partir del corpus paralelo. Aplicando el Teorema de Bayes a p(d|o) esta probabilidad se representa como el producto p(o|d)·p(d), donde el Modelo de Traducción p(o|d) es la probabilidad de que la cadena origen se traduzca por la cadena destino, y el Modelo de Lenguaje p(d) es la probabilidad de ver aquella cadena origen.

Para la creación del Modelo de Lenguaje, se utiliza la herramienta SRILM [3], que realiza la estimación de los modelos de lenguaje tipo N-grama (en la que la probabilidad de una palabra depende de las N anteriores), a partir del corpus de entrenamiento. La generación de los Modelos de Traducción se hace mediante una traducción basada en subfrases. Para esto la herramienta utilizada es el GIZA++ [4], que permite obtener los alineamientos entre palabras de la lengua origen y palabras de la lengua destino, y un módulo de modelos de subfrases a partir de estos alineamientos. Para esto se necesita un corpus paralelo. Los pasos para la generación de los modelos son:

1. Obtención del alineamiento entre palabras: a partir de los dos textos en castellano y LSE se identifican qué palabras de uno se alinean con los signos de LSE. El alineamiento se calcula en ambos sentidos: palabras-signos y signos-palabras.
2. Cálculo de una tabla de traducción léxica: se calcula a partir del alineamiento, obteniendo los valores de w(d|o) y su inversa w(o|d) para todos los pares de palabras.
3. Extracción de subsecuencias de palabras: se recopilan todos los pares de subsecuencias que sean consistentes con el alineamiento.
4. Cálculo de las probabilidades de traducción de cada subsecuencia.

3.2 Traducción

Para realizar el proceso de traducción se combinan los modelos generados en la fase anterior de entrenamiento mediante una combinación lineal de probabilidades cuyos pesos se deben ajustar. Este proceso de ajuste consiste en la ejecución iterativa del traductor Moses [5] sobre un conjunto de validación. En cada iteración se van modificando los pesos con el objetivo de maximizar los resultados de BLEU sobre ese conjunto de validación. Finalmente, y utilizando un nuevo conjunto de test se evalúa finalmente el sistema. No se realiza ningún tipo de reordenamiento previo a la fase de traducción.

4. TRADUCCIÓN BASADA EN TRADUCTORES DE ESTADOS FINITOS

Los traductores de estados finitos ("FST") parten también de un corpus paralelo de entrenamiento y, usando métodos de alineamiento basados en GIZA++, generan un conjunto de cadenas a partir de las cuales se puede inferir una gramática racional. Esta gramática se convierte en un FST caracterizado por su topología y distribuciones de probabilidad, características aprendidas con el programa GIATI [6]. En la figura 3 se muestra la arquitectura de esta solución.

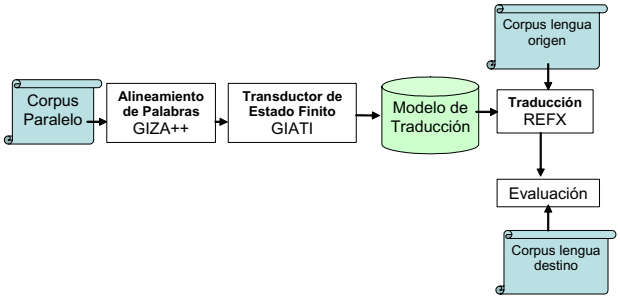


Figura 3. Traducción basada en Traductores de Estados Finitos

Las fases para la generación de los modelos son:

1. Alineamiento con GIZA++ a nivel de palabras. Igual que la primera fase del sistema de traducción basado en subsecuencias.

2. Transformación de pares de entrenamiento a frases. Se construye ahora un corpus extendido a partir de cada uno de los pares de subsecuencias de entrenamiento y sus correspondientes alineamientos obtenidos con GIZA++: se asignarán por tanto palabras de LO a su correspondiente palabra en LD gracias a su alineamiento. Se muestra a continuación un ejemplo de pares castellano / LSE y su alineamiento:

- el denei es obligatorio desde los catorce años # DNI(2) SE-LLAMA(3) OBLIGATORIO(4) DESDE(5) CATORCE(7) PLURAL(6) AÑO(8) EDAD(8)
- el denei es el documento oficial # DNI(2) SE-LLAMA(3) DOCUMENTO(5) OFICIAL(6)

A continuación se forman las palabras extendidas ("extended words", unión de palabras y signos alineados), que representan la traducción propuesta. En este ejemplo:

- (el, λ) (denei, DNI) (es, SE-LLAMA) (obligatorio, OBLIGATORIO) (desde, DESDE) (los, PLURAL), (catorce, CATORCE) (años, AÑO EDAD)
- (el, λ) (denei, DNI) (es, SE-LLAMA) (el, λ) (documento, DOCUMENTO) (oficial, OFICIAL)

3. Inferencia de un Gramática Estocástica y posteriormente de un traductor de estados finitos. Se obtiene un FST a partir de las frases con las palabras extendidas. Las probabilidades de saltos entre nodos de un FST se computan por las cuentas correspondientes en el conjunto de entrenamiento de palabras extendidas. Se ilustra este proceso en la siguiente figura, donde los nodos grises indican que la frase de salida puede terminar en ese punto

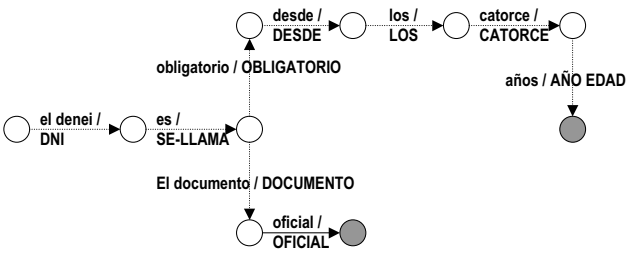


Figura 4. FST para el ejemplo anterior

5. INCORPORACIÓN DE INFORMACIÓN SINTÁTICO–SEMÁNTICA A LA TRADUCCIÓN

5.1. Traducción basada en subfrases

La incorporación de información adicional a las subfrases utilizadas para la traducción constituye una valiosa información en fases de pre o post-procesado de textos [7], ya que la traducción basada en subfrases se limita al "mapeo" de estos pequeños pedazos de texto sin incluir una información lingüística específica (morfológica, sintáctica o semántica). Con la hipótesis de que en la traducción automática se hace un uso pobre de la información morfológica que proporciona una palabra en sí misma, se plantea añadir unas categorías a las palabras para mejorar la tasa de traducción.

De acuerdo con esto, se han añadido una serie de datos (o factores) sobre las palabras. Esta información ha sido:

- Información sintáctico-semántica: una categoría que ofrezca cierta información sintáctico-semántica de la palabra. Para esta información se han aprovechado las categorías utilizadas en el sistema basado en reglas presentado en [1].
- Partes de la Oración ("Part of Speech"): una información sintáctica sobre la palabra. Se pueden clasificar las palabras, por ejemplo, en nombres, artículos, adjetivos, pronombres, verbos, adverbios, preposiciones, conjunciones, intersecciones, posesivos, demostrativos y conjunciones.
- Información adicional sobre el género o número de las palabras, el tiempo de los verbos y las características de los adverbios, por ejemplo.

Se ha realizado una labor manual para la categorización de las frases que componen la base de datos. Con este proceso de categorización, las palabras no son vistas únicamente como las unidades fundamentales del texto, sino como un vector de factores que representa varios niveles de anotación, como se observa en la siguiente figura:

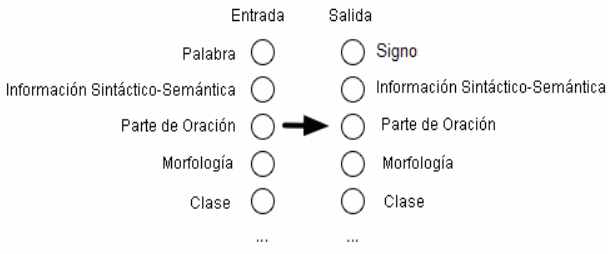


Figura 5. Diferentes tipos de factores

La plataforma de traducción basada en subsecuencias de palabras permite el entrenamiento de varios modelos considerando diferentes factores. Esos modelos se combinan para realizar la traducción. Esta combinación se hace mediante linealmente con unos pesos que se entrenan en la fase de validación.

5.2. Traducción basada en FSTs

En el caso de la traducción basada en FST se ha realizado una categorización previa del texto de entrada para luego aprender los modelos de traducción con los textos categorizados. En este caso hemos utilizado una

única categoría por palabra pero sólo para la lengua origen (castellano). Dicha categoría ha sido el primero de los factores comentados en el apartado anterior. El esquema de la arquitectura se muestra en la figura 6.

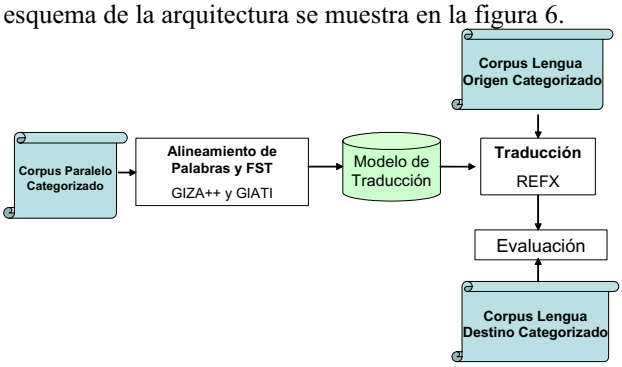


Figura 6. Utilizando textos categorizados

6. RESULTADOS Y DISCUSIÓN

La base de datos utilizada para los experimentos consiste en un corpus paralelo que contiene 414 frases típicas que diría un funcionario en una comisaría. El conjunto de frases se dividió aleatoriamente en tres grupos: entrenamiento (conteniendo el 70% de las frases), evaluación (15%) y test (15%). En relación con los experimentos de reconocimiento de voz se consideró únicamente el experimento 2 comentado en [1]. En el experimento 2 es aquel en el que el modelo de lenguaje se genera a partir del conjunto de entrenamiento, mientras que el vocabulario (540 palabrass) incluye todas las palabras (entrenamiento y test). De esta forma se evita el problema de las OOVs (Out of Vocabulary words). Los resultados de reconocimiento obtenidos en este caso fueron WER (Word Error Rate) = 15,84, I (ins.) = 1,19%, B (borr.) = 5,93%, S (sus.) = 8,72%.

Sin información sintáctico-semántica				
		WER	**BLEU**	**NIST**
Traducción basada en subsecuencias	**RAH**	37,46	0,4939	6,474
	Ref	31,75	0,5469	6,865
Traducción basada en FST	**RAH**	33,42	0,5235	6,834
	Ref	28,21	0,5905	7,350
Con información sintáctico-semántica				
		WER	**BLEU**	**NIST**
Traducción basada en subsecuencias	**RAH**	37,13	0,5124	6,606
	Ref	31,54	0,5581	7,006
Traducción basada en FST	**RAH**	33,01	0,5311	6,943
	Ref	26,06	0,6071	7,664

Tabla 1. Resultados de traducción con y sin información sintáctico semántica

En la tabla 1 se incluyen los experimentos de traducción considerando las dos situaciones: incluyendo o no la información sintáctico-semántica descrita anteriormente. Se muestran los resultados de traducción tanto para las frases de referencia (Ref) como la salida del reconocedor (RAH). Para evaluar la calidad de la traducción se utiliza la WER (Tasa de Error a la salida de la traducción), BLEU [8] y NIST [9]. Mientras WER es una medida negativa (cuando mejor es el sistema menor es esta medida), BLEU y NIST son medidas positivas (aumentan con los mejores sistemas de traducción).

Como se puede observar, con esta base de datos (del dominio de frases del DNI/pasaporte), la traducción estadística basada en FST ofrece mejores resultados que la solución tecnológica basada en subfrases. Se observa también que al incorporar la información sintáctico-semántica es posible mejorar los resultados (menor WER y mayor BLEU o NIST). Si bien es cierto que los resultados mejoran, las diferencias son muy pequeñas. El único caso en el que la mejora es importante es para el caso de traducción basada en FST de las frases de referencia. Finalmente se puede concluir que el mejor sistema es el de la traducción basada en FST incorporando información sintáctico semántica.

8. AGRADECIMIENTOS

Este trabajo ha sido posible gracias a la financiación de los siguientes proyectos: EDECAN (MEC Ref: TIN2005-08660-C04), ROBONAUTA (MEC Ref: DPI2007-66846-C02-02) y ANETO (UPM-DGUI-CAM. Ref: CCG07-UPM/TIC-1823) y al trabajo en colaboración con la Fundación CNSE.

9. BIBLIOGRAFÍA

[1] R. San-Segundo, A. Pérez, D. Ortiz, L. F. D'Haro, M. I. Torres, F. Casacuberta. "Evaluation of Alternatives on Speech to Sign Language Translation". Interspeech 2007. Antwerp, Bélgica. (ISSN:1990-9772)

[2] http://www.sign-lang.unihamburg.de/eSIGN.

[3] Stolcke A. "SRILM – An Extensible Language Modelling Toolkit". Interspeech.

[4] Koehn P., Och J., Marcu D. "Statistical Phrase-Based Translation". Human Language Technology Conference '03 (HLTNAACL 2003), Edmonton, Canada, pp. 127-133, May 2003.

[5] http://www.statmt.org/moses

[6] Casacuberta F., E. Vidal. "Machine Translation with Inferred Stochastic Finite-State Transducers". Comp. Linguistics, V30, n2, 2005-225.

[7] Koehn P., Hoang H. "Factored Translation Models". Proc. of the 2007 Annual Meeting of the ACL, pp. 868-876, Prague, Junio 2007.

[8] Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. "BLEU: a method for automatic evaluation of machine translation" in ACL-2002: 40th Annual meeting of the Association for Computational Linguistics pp. 311-318.

[9] http://www.nist.gov/speech/tests/mt/

N-II: TRADUCTOR AUTOMÁTICO ESTADÍSTICO BASADO EN NGRAMAS

Marta R. Costa-jussà, Mireia Farrús, Marc Poch, Adolfo Hernández y José B. Mariño

Centro de Investigación TALP-UPC,
Campus Nord, 08034 Barcelona
{mruiz,mfarrus,mpoch,adolfohh,canton}@gps.tsc.upc.edu

RESUMEN

Esta comunicación describe el desarrollo del traductor estadístico N-II entre catalán y castellano. Para mejorar la calidad del sistema, se llevó a cabo un riguroso análisis lingüístico. Este ha permitido plantear soluciones estadísticas y basadas en reglas que afrontan con éxito los errores más comunes de la traducción puramente estadística.

1. INTRODUCCIÓN

La traducción de voz a voz se hace a partir de la concatenación de tres sistemas: reconocimiento de voz, traducción de texto y síntesi de voz. La web N-II[1] es un ejemplo del segundo sistema y proporciona una traducción automática entre castellano y catalán en ambas direcciones siguiendo una aproximación estadística. La traducción automática estadística se basa en el hecho de que cada oración e en una lengua destino es una posible traducción de una oración f en una lengua fuente. La principal diferencia entre dos posibles traducciones de una oración dada es la probabilidad asignada a cada una que se tiene que aprender de un texto bilingüe. Por lo tanto, la traducción de una oración fuente *f* se puede formular como la búsqueda de la oración destino *e* que maximiza la probabilidad de traducción $P(e|f)$.

Dentro de los sistemas estadísticos, el N-II utiliza una aproximación basada en un modelo de lenguaje de unidades bilingües. Este sistema ha participado en varias evaluaciones de prestigio internacional obteniendo resultados competitivos, a título de ejemplo ver [2].

Para estimar los parámetros del modelo, la aproximación estadística (y como tal, la basada en Ngramas) requiere corpus bilingües paralelos (formados por pares de oraciones que se traducen mutuamente). Concretamente, para entrenar el traductor N-II, hemos utilizado el corpus paralelo del diario *El Periódico* que contiene 1.7 millones de oraciones.

El resultado del traductor estadístico ha obtenido resultados BLEU superiores al 80 % cuando se testea con un texto del mismo dominio que el utilizado en el entrenamiento. Sin embargo, las traducciones generan diversos errores que hay que tener en cuenta y rectificar para aumentar la calidad del sistema.

En esta comunicación se presenta un análisis preliminar de los errores encontrados más frecuentes, y se proponen diferentes técnicas para resolverlos, como la incorporación de reglas o información morfológica adicional. Finalmente, también se hace una breve relación de casos problemáticos que han quedado por resolver.

Así pues, la sección 2 presenta el sistema básico del N-II. La sección 3 presenta el análisis lingüístico de los errores y la sección 3 la soluciones aplicadas que se han dividido en dos tipos: las que utilizan reglas basadas en información gramatical, y las que optan por un procesado directo del texto. Finalmente, la sección 5 presenta las conclusiones y el trabajo futuro.

2. SISTEMA BÁSICO DE TRADUCCIÓN

El modelo de traducción puede entenderse como un modelo de lenguaje de unidades bilingües (llamadas tuplas). Dichas tuplas definen una segmentación monótona de los pares de oraciones utilizadas en el entrenamiento del sistema (f_1^J, e_1^I), en K unidades $(t_1, ..., t_K)$. En la extracción de las unidades bilingües, cada par de oraciones da lugar a una secuencia de tuplas que solo depende de los alineamientos internos entre las palabras de la oración.

El modelo de traducción se ha implementado utilizando un modelo de lenguaje (bilingüe) basado en n-gramas de tuplas [1]. En la traducción de una oración de entrada, el decodificador debe encontrar la secuencia de tuplas asociada a una segmentación de la oración de entrada que produzca probabilidad máxima. En general, tal probabilidad máxima se calcula como combinación lineal de modelos.

En la traducción del catalán-castellano, dado que son un par de lenguas muy paralelas, la utilización de un único modelo (el de traducción) ya permite obtener un traductor estadístico competente. Hay que tener en cuenta que este modelo de traducción incluye el modelo de lenguaje de destino. En caso de utilizar un corpus monolingüe adicional motivaría incorporar un modelo adicional de destino. El sistema de búsqueda utilizado se ha desarrollado en la

Este trabajo ha sido subvencionado por el Gobierno Español mediante el proyecto coordinado AVIVAVOZ (TEC2006-13694-C03) y el Govern de la Generalitat de Catalunya mediante el proyecto TecnoParla. Asimismo, agradecemos la colaboración de Yesika Laplaza y Carlos Alberto Henriquez y los comentarios de los revisores.

[1] http://www.n-ii.org/

UPC (MARIE[2]).

3. ANÁLISIS LINGÜÍSTICO DE ERRORES

El traductor estadístico N-II presentaba, en un análisis preliminar, una serie de errores, en una o ambas direcciones de traducción, que describimos muy brevemente a continuación.

En primer lugar, los errores encontrados al traducir del castellano al catalán fueron los siguiente:

Obligación El traductor generaba la traducción literal de *tener que* como **tenir que*, en lugar de *haver de*.

Omisión de la preposición *de* La preposición *de* se omite al traducir el verbo *deber*.

Solo Es un término que corresponde a tres categorías gramaticales diferentes: adverbio, adjetivo y nombre. Según la categoría le corresponde una traducción diferente al catalán, y presenta una gran confusión en el traductor, especialmente entre el adjetivo y el adverbio.

Apóstrofe No se cumplen las reglas de apostrofación del catalán para los artículos *el* y *la* y la preposición *de* delante de vocales.

Ele geminada *(l.l)* Aunque debería escribirse siempre con punto volado, es muy habitual encontrar la ele geminada con punto normal, hecho que causa traducciones incorrectas.

En segundo lugar, los errores encontrados al traducir del catalán al castellano se resumen en la siguiente lista:

Preposiciones *a* y *en* Estas preposiciones tienen usos muy delimitados que no se corresponden con una traducción literal correcta al castellano.

Posesivos Adjetivos y pronombres posesivos tienen la misma forma en catalán, hecho que crea ambigüedad a la hora de traducir al castellano, que utiliza formas diferentes.

Perquè Esta conjunción tiene traducciones distintas al castellano en según si introduce una oración subordinada de causa (*porque*) o de finalidad (*para que*).

Soler Las formas conjugadas *sol* y *sols* del verbo soler pueden confundirse con adjetivos.

Conjunciones *y* y *o* Estas dos conjunciones deben transformarse en *e* y *u* cuando preceden palabras que empiezan por *y* y *o*, respectivamente.

Omisión de la preposición *a* A diferencia del catalán, el castellano utiliza habitualmente la preposición *a* delante del objeto directo. Al no encontrarse en la lengua fuente, tampoco aparece en la lengua origen.

Finalmente, los errores encontrados en ambas direcciones fueron los siguientes:

Concordancia de género Una palabra femenina (masculina) en castellano se puede corresponder con una palabra masculina (femenina) en catalán (p.ej. *la señal - el senyal*).

[2]http://gps-tsc.upc.es/veu/soft/soft/marie/

Números Hay números que no aparecen en el corpus de entrenamiento, por lo que no se genera ninguna traducción.

Horas Las expresiones de las horas en castellano y en catalán son, formalmente, diferentes. Por consiguiente, la traducción, en muchos casos, no es literal. La diferencia principal es la utilización de los cuartos: mientras el castellano se expresa mediante los cuartos que *pasan* de una determinada hora, en catalán se habla de los cuartos que se *acercan* a la hora siguiente: *Las cuatro y cuarto* se traduce por *Un quart de cinc*.

Clíticos Con frecuencia, el traductor omite los pronombres personales adheridos al verbo. En otras ocasiones, aunque la traducción de los pronombres personales sea la correcta, el error se encuentra, a menuda, en una combinación incorrecta del pronombre con el verbo en cuestión.

Palabras desconocidas Hay palabras que el corpus contiene únicamente al inicio de oración; por consiguiente, estas palabras solo se encuentran en mayúscula, lo que implica que la misma palabra escrita en minúsculas aparezca como desconocida.

sectionSoluciones aplicadas

Para la solución de algunos de los problemas descritos en la sección anterior se han aplicado dos tipos de técnicas: las técnicas basadas en la utilización de la categoría gramatical de las palabras, y las técnicas basadas en la corrección mediante un procesado directo del texto. Para la evaluación de las soluciones se ha realizado un análisis humano. En la Tabla 1 se muestran algunos ejemplos en los cuales dado un enunciado (O) se compara su correspondiente traducción antes (T1) y después (T2) de utilizar las técnicas descritas.

3.1. Reglas que utilizan la categoría gramatical

Las categorías gramaticales se han incorporado con éxito en traducción estadística para tratar problemas como el reordenamiento [4] y el análisis automático de los errores [5]. El objetivo es adjuntar la categoría gramatical (*tag*) correspondiente a la palabra a tratar, de manera que el modelo estadístico sea capaz de distinguir las palabras en función de su categoría y aprender el contexto.

3.1.1. Desambiguación de la homonimia

A menudo encontramos dos palabras iguales en la lengua origen que no lo son en la lengua destino y que causan traducciones incorrectas. Si las palabras son homónimas y se diferencian por su categoría gramatical, ésta podrá utilizarse para desambiguarlas.

En el caso del *solo*, se diseñan unas reglas que nos identifiquen, en los casos dudosos, si es un adverbio o un adjetivo. Se aplican las reglas en la lengua origen y se adjunta el *tag* a la palabra. Así pues, una oración de la

lengua fuente como *Venía solo.* se modifica a *Venía solo_<ADJ>.*), de manera que el modelo estadístico será capaz de distinguir ambos casos.

Un proceso similar se realiza para los posesivos del catalán: se han diseñado unas reglas que permiten etiquetar la palabra como adjetivo o pronombre posesivo y las etiquetas se incorporan posteriormente a la lengua origen. En el caso del *soler*, en lugar de generar unas reglas para detectar que *sol/sols* son verbos, se ha adjuntado el *tag* correspondiente que proporciona el Freeling [3].

3.1.2. Categorización

En una frase a traducir pueden aparecer números que no existen en el corpus de entrenamiento y, entonces, estas palabras son desconocidas y no se traducen. Para evitar este problema, hemos planteado unas reglas que detectan los números en la lengua origen, los codifican y los generan en la lengua destino. Para detectar los números hay que tener en cuenta su estructura (p.ej. palabra compuesta, con o sin guión) y, como dificultad añadida, se tiene que considerar que los números pueden tener género. Se define una codificación concreta para que en el momento de la generación se sea coherente con el número que se ha detectado. No se han categorizado los números: *un/una*, *dos/dues*, *nou* y *deu* por ser palabras que no son siempre números.

Por otro lado, la expresión de las horas en catalán y en castellano es distinta (como se ha explicado en 3) y el hecho de que el corpus contenga escasos ejemplos de horas puede generar errores en la traducción. Se ha optado por el mismo planteamiento que con los números: se detectan las horas, se codifican y se generan. Para detectar las horas primero hay que identificar su estructura, teniendo en cuenta que hay varias estructuras posibles para una misma hora. Asimismo, cara hora puede tener un contexto que se modifica en traducción. Por ejemplo: *Són dos quarts de dues* se traduce por *Es la una y media*. En este caso, también se modifica la forma verbal (pasa de 3a persona del plural a 3a persona del singular). Así pues, en este ejemplo, la frase entera se detecta toda como una hora. Se codifica de manera que se mantenga la información necesaria para poder generar la hora coherentemente.

3.1.3. Clíticos

En la lengua fuente, los clíticos se detectan y se separan del verbo utilizando el Freeling. Tras la traducción, deben juntarse de nuevo con el verbo. Este proceso de combinación se trata con unas reglas que tienen en cuenta dos factores; en primer lugar, las reglas de acentuación en castellano, ya que la posición de la sílaba tónica cambia al adherir un pronombre enclítico al verbo: *vende* + ***lo*** → *vénde**lo***. En segundo lugar, las reglas de combinación de los pronombres en catalán[3] que, a diferencia del castellano, se escriben con guión o apóstrofe según el caso, y no se alteran las reglas de acentuación: *seguir* + ***lo*** → *seguir-**lo***; *compra* + ***el*** → *compra'**l***; y ***el*** + *aixecava* → ***l'***$^{\prime}$*aixecava*.

3.1.4. Apóstrofe

La apostrofación en catalán sigue, en general, una regla básica: se apostrofan los artículos *el*, *la* y la preposición *de* cuando preceden palabras que empiezan por vocal o *h* muda: *el arbre* → *l*$^{\prime}$*arbre*; *la hora* → *l'hora*; y *de eines* → *d*$^{\prime}$*eines*.

A esta regla se le aplican excepciones[4]:

- No se apostrofan delante de palabras que empiezan por *i* o *u* semiconsonánticas: *el uombat*, *la hiena*, *de iogurt*.
- No se apostrofa el artículo femenino delante de palabras que empiezan por *i* o *u* átonas (incluyendo la *h* muda): *la universitat*, *la Irene*.
- No se apostrofan ni el artículo femenino ni la preposición delante del prefijo negativo *a*: *la anormalitat*, *de asimètric*.
- No se apostrofan *la una* (hora), *la ira*, *la host* y los nombres de letra (*la e*, *la hac*, *la erra*, etc.).

3.1.5. Tratamiento de mayúsculas a inicio de oración

Para minorizar el problema genérico (al que se enfrenta cualquier traductor basado en corpus) de las palabras desconocidas, se ha propuesto una técnica que utiliza información morfológica. Concretamente, se trata de pasar a minúsculas todas aquellas palabras a inicio de oración a excepción de nombres propios, nombres comunes y adjetivos, ya que estas palabras son susceptibles de ser un nombre propio. De esta forma, las palabras que solo estaban en mayúscula en el corpus de entrenamiento y, por lo tanto, en minúscula eran desconocidas, tienen traducción.

3.1.6. Tratamiento de las concordancias

Afrontamos la concordancia de género mediante la utilización de un modelo de *tags* de la lengua destino. Esto permite beneficiar aquellas secuencias de palabras que mantienen coherencia en género, por ejemplo: será más probable una secuencia tal y como *pilota verda* que *pilota verde* porque el modelo de *tags* ha visto más veces un nombre femenino seguido de un adjetivo femenino que un nombre femenino seguido de un adjetivo masculino. El modelo de tags podrá ayudar en la medida que el modelo de traducción, es decir, las tuplas lo permitan. Por ejemplo, la traducción de *senyal blanc* continúa siendo *señal blanco* porque en el corpus no existe la tupla *blanc#blanca*.

Asimismo, como existe una tendencia a omitir palabras utilizamos una bonificación de palabras.

[3] www.cpnl.cat/media/upload/pdf/cnlortografia0305_editora_grup_30_19.pdf

[4] http://www.uoc.edu/serveilinguistic/criteris/ortografia/apostrof.html

posesivos	(O) Els meus amics no són **els teus**.
	(T1) Mis amigos no están ***tus**.
	(T2) Mis amigos no son **los tuyos**.
solo	(O) Era **solo** un niño.
	(T1) Era ***sol** un nen.
	(T2) Només era un nen.
sol/sols	(O) La Creu Roja **sol** disposar de quatre.
	(T1) La Cruz Roja ***solo** disponer de cuatro.
	(T2) La Cruz Roja **suele** disponer de cuatro.
ele	(O) S'ha reformat a **Brussel.les**.
geminada	(T1) Se ha reformado en ***Bruselas. las**.
	(T2) Se ha reformado en **Bruselas**.
obligación	(O) Nos lo **tenemos que** creer.
	(T1) Ens ho ***tenim que** creure.
	(T2) Ens ho **hem de** creure.
y/o	(O) Com a Blanes **o** Olot.
	(T1) Como Blanes ***o** Olot.
	(T2) Como Blanes **u** Olot.
clíticos	(O) No quiero ver**te** más por aquí.
	(T1) No vull veure **et** més per aquí.
	(T2) No vull veure'**t** més per aquí.
apóstrofe	(O) **La acepta** hasta el final.
	(T1) ***La accepta** fins al final.
	(T2) **Láccepta** fins al final.
números	(O) Ha aprovat lálliberament de **quatre-cents quaranta-un** presoners.
	(T1) Ha aprobado la liberación de ***quatre-cents quaranta-un** prisoneros.
	(T2) Ha aprobado la liberación de **cuatrocientos cuarenta y un** prisoneros
horas	(O) Són **tres quarts de vuit**.
	(T1) Son ***tres cuartos de ocho**.
	(T2) Son **las ocho menos cuarto**.
mayúsculas	(O) No entenc per què no hi **assisteixes**.
	(T1) No entiendo por qué no ***assisteixes**.
	(T2) No entiendo por qué no **asistes**.

Tabla 1. Ejemplos de correción de errores.

3.2. Procesado directo del texto

Algunos errores precisan de un preprocesado directo en el texto antes o después de realizar la traducción. La ele geminada se ha tratado antes de la traducción, normalizando la escritura del punto utilizado. En otros errores como la obligación *tener que* y las conjunciones *y/o* se han tratado como postprocesado después de realizar la traducción.

4. EVALUACIÓN

Para la evaluación de las técnicas que hemos incorporado usamos el test de *El Periódico* de 2000 oraciones y una sola referencia. La Tabla 2 compara la eficacia en BLEU de todas las mejoras de este artículo juntas frente a un sistema de referencia que no las incorporaba.

	es > ca	ca > es
Sistema de referencia	76.66	76.98
+ mejoras planteadas	82.28	81.74

Tabla 2. Resultados BLEU en ambas direcciones de traducción.

Hemos de tener en cuenta que lo que tiene más significancia estadística en este test es el tratamiento de los clíticos, puesto que en el sistema de referencia no se han tratado de ningún modo. Además, hemos planteado problemas que el usuario de un traductor podía encontrar y que no se suelen encontrar en textos periodísticos.

5. CONCLUSIONES Y TRABAJO FUTURO

En esta comunicación se ha descrito el traductor estadístico N-II. Se ha detallado el resultado de un análisis lingüístico de errores y se han propuesto soluciones basadas en reglas y de carácter estadístico así como pre y postprocesos de corrección de errores. La eficiencia de las soluciones aportadas se ha demostrado con ejemplos significativos y con una evaluación automática.

6. BIBLIOGRAFÍA

[1] Mariño, J.B. , Banchs, R.E. , Crego, J.M. , de Gispert, A. , Lambert, P. , Fonollosa, J.A.R. y Costa-jussà, M.R. *N-gram Based Machine Translation*. Computational Linguistics, 32:4:527–549, 2006.

[2] Lambert, P. , Costa-jussà, M.R. , Crego, J.M., Khalilov, M., Mariño, J.B., Banchs, R.E., R. Fonollosa, J.A. y Schwenk, H. *The TALP Ngram-based SMT System for IWSLT 2007.* En Proc. of the International Workshop on Spoken Language Translation (IWSLT) pp 169-174, Trento, 2007.

[3] Carreras, X. , Chao, I. , Padró, L. , Padró, M., *FreeLing: An Open-Source Suite of Language Analyzers*, En Proc. of the Conference on Language Resources and Evaluation, LREC. Lisboa, 2004.

[4] Crego, J.M. y Mariño, J.B. *Improving SMT by coupling reordering and decoding* En Machine Translation, 20:3:199–215, 2007.

[5] Popović, M., de Gispert, A., Gupta, D., Lambert, P., Ney, H., Mariño, J.B. y Banchs, R. *Morpho-syntactic Information for Automatic Error Analysis of Statistical Machine Translation Output* En Proc. of the HLT/NAACL Workshop on Statistical Machine Translation, pages 1-6, New York, 2006.

STATISTICAL METHODS FOR SPEECH TECHNOLOGIES IN BASQUE LANGUAGE

M. Inés Torres[1]*, Víctor Guijarrubia*[1]*, Raquel Justo*[1]*, Alicia Pérez*[1]*, Francisco Casacuberta*[2]

[1]Dep. Electricity and Electronics. University of the Basque Country
manes.torres@ehu.es
[2]Dep. of Information Systems and Computation. Technical University of Valencia
fcn@dsic.upv.es

ABSTRACT

The overall goal of this work is to build a speech-input decoder and translation application for Basque, Spanish and English, which allows to speak in whatever the aforementioned languages, and according to the identified language it proceeds to obtain its text transcription and translation into the other two languages. This application gathers different technologies such as language identification, recognition and translation, all of them developed on the basis of statistical methods. In addition, it entails a real challenge, as there are scarce resources developed for Basque language. After carrying out an analysis of different approaches, adapted methods for Basque language features have been developed and assessed.

1. INTRODUCTION

Current trends in both automatic speech recognition (ASR) and machine translation go towards the use of statistical methods as they have proved to offer a very competitive performance. In this work, we made use of them but we also proposed new approaches in order to obtain a better adaptation of the classical methods to the specific features of Basque language.

Basque is a pre-Indoeuropean language of unknown origin. It shares official status, along with Spanish, in the Basque Country, nevertheless it is spoken in a few more regions, such as in south west France and in small American communities. It is a minority language, and thus, great efforts are being made in order to enrich the currently scarce linguistic resources. Regarding the morphology, Basque is an extremely inflected language both in nouns and verbs.

Considering the syntactic structure, both languages present a different arrangement of the words within the sentence. Therefore, an appropriate *language model* capable of capturing the specific structure of the sentences in Basque has a great importance when speech recognition is carried out. In this work a category-based language model was used. Category-based models allow to capture relationships related to the structure of the sentences and in addition they have shown to be a good choice to face the issues derived from the resource scarcity, as it is the case.

From a phonetic point of view, the set of Basque phones does not differ much from the Spanish one. The two languages share the same vowel triangle (only five vowels). Nevertheless, Basque includes larger sets of fricative and affricate sounds [1].

In this work, several fields of the natural language processing have been explored and adapted for Basque language: language identification between Basque, Spanish and English; language modeling and categorization; speech translation. Needless to say, the main tool involved in all these fields is the automatic speech recognition system, which has been developed in this group and constitutes the state of the art in what continuous speech recognition systems concerns. The acoustic models employed are continuous Hidden Markov Models and regarding to the language model (LM), a *k-testable in the strict sense* [2] LM is used. This model integrates n-gram models (n ranging from 1 to k) and allows for back-off smoothing.

The following of this paper is organized as follows: section 2 is devoted to describe the task and corpus object of the study. Sections 3, 4 and 5 briefly describe statistical methods, experimental results and specific challenges that have to be faced when Basque language is involved in language identification, categorization and machine translation respectively. Finally, in section 6, a discussion of the overall results and proposals for future work are reported.

2. TASK AND CORPUS

METEUS is a trilingual text and speech corpus in Basque, Spanish and English. It consists of weather forecast reports picked up from the Internet in Spanish and Basque and later translated into English by a professional translator. The main features are shown in Table 1.

This is the first multilingual corpus that joins natural language text and speech in Basque. It seems to be a

This work has been partially supported by the University of the Basque Country under grant GIU07/57, by CYCIT under grant TIN2005-08660-C04-03 and by Consolider Ingenio-2010 program MIPRCV (CSD2007-00018).

suitable choice for comparison purposes between different languages, and above all, for statistical speech translation, a vaguely explored field in Basque language.

With regard to the speech test, it is a training-independent set that consists of 500 different sentences. Each sentence has been recorded for at least 3 speakers, getting as a result a total of 1800 utterances by 36 speakers for each language. Notice that since the speech sub-test is training independent (instead of being a randomly selected subset), it is suitable as a benchmark to evaluate the systems under the worst situation. Therefore, the results obtained with this speech test are pessimistic, and thus, appropriate in order to establish the lower threshold of the system.

		Basque	**Spanish**	**English**
Training	Sentences	14,615		
	Different sentences	7,523	7,198	6,634
	Words	187,195	191,156	195,575
	Vocabulary	1,135	702	498
	Average Length	12.8	13.0	13.3
Test	Sentences	500		
	Words	8,274	8,706	9,150
	Average Length	16.5	17.4	18.3
	Perplexity (3grams)	6.7	4.8	5.8

Table 1. Main features of METEUS corpus.

The figures of the Table 1 show that there is a great difference in terms of vocabulary for the three languages within the same application (see Table 1). Basque language is a highly inflected language with more than 25 declension cases, whereas English is morphologically simpler. The reliability of the statistics over a smaller number of words with the same amount of training sentences, is likely to be higher, therefore, we expect worse probability distributions to be estimated over the models involving the Basque language.

3. LANGUAGE IDENTIFICATION

Language identification (LID) is a classical problem that is strongly tied to multilingual speech recognition and dialogue systems. The ultimate goal of any LID system is to identify the language being used by an unknown speaker. It has been addressed in the past using a variety of tactics; for instance, those exploiting prosodic cues as rhythm or intonation. Nevertheless, most of them are based on speech recognition approximations: phone decoding approaches, which rely on phone sequences; or large-vocabulary continuous-speech recognition approaches, which operate based on full lexical sequences. A thorough analysis discussing the current state of the LID systems can be consulted here [3].

In general, a LID system is composed of three components: a speech tokenizer that converts the speech into a sequence of tokens; a statistical language model which captures the relationships between the tokens; and a classifier that hypothesizes a language from among the set of languages.

In this paper, we focus on phone decoding approaches. These techniques rely on acoustic phonetic decoders, which find the best sequence of phonetic units depending on the input signal. Some phonotactic models can then be used to analyze these sequences and assign some scores to each language. These phonotactic models can be applied after the decoding process, that is a phone recognition followed by *n*-gram Language Modeling (PRLM), or during the decoding process (PPR) [4]. The language of the utterance is selected to be that with the best score.

The results, in terms of LID accuracy, are summarized in Table 2. In this case, we opted for using a PPR approach, since this yielded the best results [5].

	Basque	**Spanish**	**English**
Accuracies(%)	99.67	99.89	95.33

Table 2. LID accuracies values.

As can be derived, for Spanish and Basque, accuracies of nearly 100% are achieved. For English, the accuracies are also competitive, but slightly lower than those for Spanish and Basque. The acoustic modeling could be the reason for this. Whereas the acoustic models for English are trained using a phonetical transcription based on a dictionary, for Basque and Spanish this transcription is performed using rules. So the HMM sets for Basque and Spanish are better estimated and the acoustic scores are higher. To improve the results, more accurate phonotactic models would be required. Another option could be incorporating different sources of information so that the classifier has more cues to hypothesize the uttered language.

4. CATEGORIZATION AND SPEECH RECOGNITION

Nowadays statistical language models (word n-gram LMs, k-tss models, etc.) are being used in *automatic speech recognition* (ASR) systems. Large amount of training data are required to get a robust estimation of the parameters defining such models. However, there are numerous ASR applications for which the amount of training material available is rather limited. One of the ways to deal with data sparseness is to cluster the vocabulary of the application into a smaller number of categories or classes. Thus, an alternative approach as a class n-gram LM (M_c) [6] could be used. Class n-gram LMs are more compact and generalizes better on unseen events. Nevertheless, relations among the categories of words are only captured, while it is assumed that the inter-word transition probability depends on the word classes. This fact degrades the performance of the ASR system.

In order to avoid the loss of information associated with the use of a class n-gram LM, alternative approaches might be used, e.g. [7]. We propose a different approach that takes advantage of two information sources: words and categories. This approach could be understood as a LM based on categories consisting of segments (or sequences of words) instead of being made up of isolated words. We have employed two different ways of dealing with sequences of words inside the classes. Thus, two different approaches to this kind of LM can be considered : M_{sl} and M_{sw} [8]. These models take into account the relations among the words that take part in the segments of a category.

In this work, we study whether different class-based LMs (M_c, M_{sl} and M_{sw}), integrated into an ASR system, can improve the ASR system performance when experiments over the Basque part of the METEUS corpus are carried out. Let us notice that Basque is a minority language and therefore, few training material is available. Due to this fact, this task is well-suited to study improvements derived from categorization within the language model.

The employed categories were automatically generated by the aid of *mkcls* [9], a free toolkit designed to train word classes based on a maximum-likelihood-criterion. On the other hand, the set of word sequences employed within M_{sl} and M_{sw} models were obtained using also a statistical criterion. Specifically the most frequent n-grams of the corpus were selected as segments. In this sense and in order to avoid rare or unimportant n-grams only segments exceeding a minimum number of occurrences were considered.

Different sets of 300, 400, 500 and 600 categories were generated and the correspoding class-based language models were inferred. The proposed language models were then integrated into the ASR system and evaluated in terms of *word error rate* (WER). The obtained WER results are shown in Table 3 along with that obtained under a classical word-based LM. As can be seen in Table 3, better results are obtained when using word segment based categories (in both M_{sl} and M_{sw} models) than when employing classical class n-gram models (M_c). On the other hand, regarding the experiments carried out with M_{sl} model, a significant drop of WER is observed compared to the M_{sw} model for all of the selected sets of categories. Furthermore, the result obtained with M_{sl} and 500 classes slightly improves the WER value obtained with the word-based LM M_w (a 1.7%).

WER (%)				
stat. cat.	M_{sl}	M_{sw}	M_c	M_w
300	6.17	6.68	6.66	5.91
400	6.01	6.59	6.53	
500	5.81	6.37	6.47	
600	6.02	6.51	6.62	

Table 3. WER results using METEUS, for a classical word n-gram LM (M_w), a classical class n-gram LM M_c and the two proposed category-based LMs containing segments of words (M_{sw} and M_{sl}). Different sets of 300, 400, 500 and 600 statistical classes were employed in all category-based LMs

5. SPEECH TRANSLATION

Stochastic finite state transducers (SFST), thoroughly described in [10], have proved to be useful for both text and speech input machine translation applications [11].

The SFST is characterized by both the topology and the probability distributions. These distinctive features can be automatically learnt from bilingual corpora by efficient algorithms, such as GIATI (Grammar Inference and Alignments for Transducers Inference) [12]. Regarding the topology, a 3-TSS with Witten-Bell smoothing has been selected for all the following experiments.

Once the SFST has been inferred, given a source sentence $\mathbf{s}$, the decoding process can be summarized in equation (1). This expression involves a searching for the most likely target string $\widehat{\mathbf{t}}$, being $d(\mathbf{s},\mathbf{t})$ a path in the SFST, compatible with both the input sentence $\mathbf{s}$ and the output $\mathbf{t}$. Therefore, the searching criterion in the SFST deals with the joint probability of sentence pairs.

$$\widehat{\mathbf{t}} = \arg\max_{\mathbf{t}} P(\mathbf{s},\mathbf{t}) \approx \arg\max_{\mathbf{t}} \max_{d(\mathbf{s},\mathbf{t})} P(d(\mathbf{s},\mathbf{t})) \quad (1)$$

Furthermore, speech input translation aims at looking for the likeliest target language string ($\mathbf{t}$) given the acoustic representation ($\mathbf{x}$) of a source language hidden string ($\mathbf{s}$), as shown in equation (2). Once again, the most likely target is found in terms of Viterbi algorithm.

$$\widehat{\mathbf{t}} = \arg\max_{\mathbf{t}} P(\mathbf{t}|\mathbf{x}) = \arg\max_{\mathbf{t}} \sum_{s} P(\mathbf{s},\mathbf{t}|\mathbf{x}) \quad (2)$$

SFST allows to integrate acoustic models within the network so that speech translation can be carried out at a single decoding step [11]. This is an alternative to the commonly decoupled architecture that makes use of a text-to-text translation system in serial with a speech recognition system. The integrated architecture has shown to get better translation results in many tasks than the decoupled one, in addition, it is rather efficient in what time cost concerns.

In practice, the procedure to build such an integrated architecture from a common ASR is not other than making use of the SFST instead of the LM. In fact, the input projection of a transducer can be seen as an input LM. Regarding the lexical model of the ASR, both the input and the output substrings have to be stored in order to produce both the recognized sequence and its translation.

Experimental results of Table 4 show the performance of the SFST for text and speech input translation. The commonly used automatic evaluation measures have been selected in order to asses the performance of the system:

word error rate (WER) and *bilingual evaluation understudy* (BLEU).

	Basque→Spanish		Basque→English	
	WER	**BLEU**	**WER**	**BLEU**
Text	43.66	48.31	50.01	44.97
Speech	47.87	45.12	54.93	42.19

Table 4. Speech input machine translation results.

Analyzing the translations in detail, it has been shown that the most frequent error sources are the following ones: wrong order, either in local or long range; wrong lexical choices, related to either style or case; wrong punctuation.

There are some specific features of the Basque language to bear in mind in order to improve the translation models. On the one hand, the agglutinant nature, and on the other hand, the long distance reordering issues, since the usual construction for both Spanish and English is as Subject + Verb + Objects, while for Basque is as Subject + Objects + Verb. These difficulties have been proved to be more efficiently tackled in terms of phrases as a translation unit than in term of words [13], at least regarding Spanish to Basque translation.

6. CONCLUSIONS AND FUTURE WORK

Summing up, the following concluding remarks have been reached with regard to the different techniques explored in this work.

Good trilingual language identification accuracies are achieved. The Basque language is almost always properly identified even if it is acoustically similar to Spanish. A combination of different techniques should be explored to improve the LID accuracies.

Regarding category-based LMs for the *speech recognition system*, segment-based categories, taking advantage of two information sources, were needed to obtain improved results of WER. On the other hand, for further work, different criteria could be explored in the categorization and segmentation process, e.g. a linguistic one.

Better translation results could be obtained including more linguistic knowledge in the statistical model. Furthermore, alternative methodologies to exploit both statistical and linguistic knowledge sources should be explored. In future work we aim at studying *factored translation models* combining both running words, lemmas, POS and statistical Tags within a finite-state framework.

7. BIBLIOGRAPHY

[1] V. Guijarrubia, M. I. Torres, and L.J. Rodríguez, "Evaluation of a spoken phonetic database in basque language," in *Proceedings of LREC 2004*, Lisboa, 2004, vol. 6, pp. 2127–2130.

[2] M. I. Torres and A. Varona, "k-tss language models in speech recognition systems," *Computer Speech and Language*, vol. 15, no. 2, pp. 127–149, 2001.

[3] A. F. Martin and A. N. Le, "The current state of language recognition: NIST 2005 evaluation results," in *Proceedings of the IEEE Odyssey 2006, SLR Workshop*, Puerto Rico, 2006.

[4] M. A. Zissman and E. Singer, "Automatic language identification of telephone speech messages using phoneme recognition and n-gram modelling," in *Proceedings of ICASSP-94*, Adelaide, Australia, 1994, vol. 1, pp. 305–308.

[5] V. G. Guijarrubia and M. I. Torres, "Phone-segments based language identification for spanish, basque and english," in *CIARP*, Luis Rueda, Domingo Mery, and Josef Kittler, Eds. 2007, vol. 4756 of *Lecture Notes in Computer Science*, pp. 106–114, Springer.

[6] T. R. Niesler and P. C. Woodland, "A variable-length category-based n-gram language model," in *IEEE ICASSP-96*, Atlanta, GA, 1996, IEEE, vol. I, pp. 164–167.

[7] I. Zitouni, "Backoff hierarchical class n-gram language models: effectiveness to model unseen events in speech recognition," *Computer Speech and Language*, vol. 21, no. 1, pp. 99–104, 2007.

[8] R. Justo and M. I. Torres, "Phrases in category-based language models for spanish and basque ASR," in *Proceedings of Interspeech 2007*, Antwerp, Belgium, August 27-31 2007, pp. 2377–2380.

[9] F. J. Och, "An efficient method for determining bilingual word classes," in *EACL '99*, Bergen, Norway, June 1999, pp. 71–76.

[10] E. Vidal, F. Thollard, F. Casacuberta C. de la Higuera, and R. Carrasco, "Probabilistic finite-state machines - part II," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 7, pp. 1025–1039, 2005.

[11] F. Casacuberta and E. Vidal, "Machine translation with inferred stochastic finite-state transducers," *Computational Linguistics*, vol. 30, no. 2, pp. 205–225, 2004.

[12] F. Casacuberta and E. Vidal, "Learning finite-state models for machine translation," *Machine Learning*, vol. 66, no. 1, pp. 69–91, 2007.

[13] A. Pérez, M. I. Torres, and F. Casacuberta, "Joining linguistic and statistical methods for Spanish-to-Basque speech translation," *Speech Communication*, 2008, doi:10.1016/j.specom.2008.05.016.

TÉCNICAS ESTADÍSTICAS PARA EL FILTRADO DE UN CORPUS BILINGÜE EN TRADUCCIÓN AUTOMÁTICA

Enrique Montolar, Marta R. Costa-Jussà y José A. R. Fonollosa

Universitat Politècnica de Catalunya, UPC
Centro de Investigación TALP,
Campus Nord, 08034 Barcelona
enrique.montolar@gmail.com {mruiz,adrian}@gps.tsc.upc.edu

RESUMEN

Los sistemas de traducción automática estadística están basados en corpus bilingües. La calidad de estos es un factor determinante en la calidad de la traducción. En esta comunicación, presentamos dos filtrados estadísticos que permiten descartar las oraciones del corpus bilingüe que tienen menor probabilidad de ser paralelas. Como resultado, se obtiene una mejora superior a 1 punto BLEU y, al reducir el corpus de entrenamiento, disminuye el coste computacional.

1. INTRODUCCIÓN

La traducción automática estadística se basa en el hecho de que cada oración e en un lenguaje destino es una posible traducción de una oración f en un lenguaje fuente. La principal diferencia entre dos posibles traducciones de una oración dada es la probabilidad asignada a cada una, que se tiene que aprender de un texto bilingüe. Por lo tanto, la traducción de una oración fuente *f* se puede formular como la búsqueda de la oración destino *e* que maximiza la probabilidad de traducción $P(e|f)$.

La aproximación estadística a la traducción automática es una aproximación basada en corpus. Concretamente, se requieren corpus paralelos a nivel de oración.

Un problema habitual de los corpus bilingües es la presencia de oraciones no paralelas que dan lugar a unidades de traducción erróneas. En esta comunicación se presentan dos métodos de filtrado del corpus con el objetivo de eliminar estas oraciones no paralelas.

La sección 2 presenta el sistema básico de traducción. La sección 3 presenta las técnica de filtrado estadístico basadas en el Modelo IBM1 y en el Position Error Rate (PER). La sección 4 muestra los experimentos y la sección 5 presenta las conclusiones.

Este trabajo ha sido subvencionado por el Gobierno Español mediante el proyecto coordinado AVIVAVOZ (TEC2006-13694-C03) y el Govern de la Generalitat de Catalunya mediante el proyecto TecnoParla.

2. SISTEMA DE TRADUCCIÓN

Como sistema de traducción estadística se ha utilizado un sistema basado en n-gramas [1].

El modelo de traducción puede entenderse como un modelo de lenguaje de unidades bilingües (llamadas tuplas). Dichas tuplas, definen una segmentación monótona de los pares de oraciones utilizadas en el entrenamiento del sistema (f_1^J, e_1^I), en K unidades $(t_1, ..., t_K)$.

En la extracción de las unidades bilingües, cada par de oraciones da lugar a una secuencia de tuplas que sólo depende de los alineamientos internos entre las palabras de la oración. La Figura 1 muestra un ejemplo de extracción de tuplas. El modelo de traducción se ha implementado utilizando un modelo de lenguaje (bilingüe) basado en n-gramas [1]:

$$p(e, f) = Pr(t_1^K) = \prod_{k=1}^{K} p(t_k \,|\, t_{k-2}, t_{k-1}) \qquad (1)$$

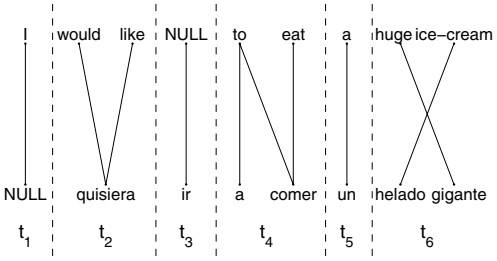


Figura 1. *Extracción de tuplas a partir de un par de oraciones alineadas palabra a palabra.*

En la traducción de una oración de entrada, el decodificador debe encontrar la secuencia de tuplas asociada a una segmentación de la oración de entrada que produzca probabilidad máxima. En general, tal probabilidad máxima se calcula como combinación lineal de modelos utilizados en el sistema de traducción.

El sistema de búsqueda utilizado se ha desarrollado en la UPC: MARIE [1].

[1] *http://gps-tsc.upc.es/veu/soft/soft/marie/*

3. FILTRADO ESTADÍSTICO DE UN CORPUS BILINGÜE

Dado un corpus bilingüe de entrenamiento, nuestro objetivo se centra en analizar dicho corpus y filtrarlo para intentar obtener un nuevo corpus que nos permita mejorar nuestro sistema de traducción. Concretamente, queremos reducir las tuplas erróneas y mejorar el vocabulario bilingüe. Para ello debemos detectar frases dentro del corpus que no sean paralelas. A continuación presentamos dos soluciones basadas en criterios diferentes para detectar y eliminar oraciones que no sean paralelas.

3.1. Planteamiento mediante modelo IBM1

El método propuesto para detectar y posteriormente descartar frases no paralelas, es decir para filtrar el corpus de entrenamiento, estará basado en el modelo de alineamiento IBM1. Recordemos que dicho modelo surge de la necesidad de establecer un alineamiento entre las palabras de un par de oraciones, dados dos textos paralelos a nivel de oración, que son traducciones mutuas del par de lenguas que nos ocupan en cada caso. Los modelos IBM calculan la probabilidad de que dos palabras estén alineadas entre ellas, es decir la probabilidad de que una palabra de la oración origen se corresponda con una palabra de la oración destino. Son modelos basados en palabras, ya que asumen que en el proceso de traducción se establecen relaciones entre palabras individuales de las frases origen y destino.

Así pues podemos establecer la probabilidad de traducción de un par de frases en función de la probabilidad de traducción de las palabras que las componen. Analizando dicha probabilidad para cada par de frases de nuestro corpus, podemos buscar un umbral de probabilidad que nos indicará si las frase son paralelas. Es decir podremos determinar que la probabilidad de que una frase de un texto se corresponda a la frase alineada con esta del otro texto es tan baja, que las dos frases no se corresponden es decir no son paralelas.

La probabilidad de traducción asignada según el modelo IBM1 a cada oración se calcula mediante la expresión:

$$h_{LEX}(e,f) = log\frac{1}{(I+J)^J}\prod_{j=1}^{J}\sum_{i=0}^{I} p_{IBM_1(e_j^n|f_i^n)} \quad (2)$$

Donde e_j^n y f_i^n son la j^{esima} y i^{esima} palabras en la oraciones fuente y destino, con I número de palabras de la fuente y J número de palabras del destino. Así pues $p_{IBM_1(e_j^n|f_i^n)}$ serán las probabilidades de traducción en la dirección fuente-destino $p(e_k/f_k)$ asignada por el modelo IBM1.

Teniendo en cuenta que el modelo IBM-1 es asimétrico, es decir es diferente según el sentido de la traducción, debemos calcular también la probabilidad de traducción en el sentido inverso, este se calcula mediante la expresión:

$$h_{LEX}inv(f,e) = log\frac{1}{(J+I)^I}\prod_{i=1}^{I}\sum_{j=0}^{J} p_{IBM_1(f_i^n|e_j^n)} \quad (3)$$

Donde tenemos $p_{IBM_1(f_j^n|e_i^n)}$ serán las probabilidades de traducción en la dirección $p(f_k/e_k)$ asignada por el modelo IBM1.

Así pues, utilizando las probabilidades léxicas obtenidas del Modelo IBM1, calcularemos la probabilidad de que dos oraciones paralelas en el corpus bilingüe sean traducciones entre ellas.

Al calcular las probabilidades de frases paralelas, debemos tener en cuenta varios factores que influyen en el cálculo de la probabilidad, como la repetición de palabras con una gran probabilidad de coincidencia.

El hecho de que en un par de frases aparezcan las palabras más comunes de cada idioma, nos incrementaría mucho la probabilidad de que fueran paralelas, aunque realmente no tiene porque ser así. Dos frases no coincidentes en absoluto a nivel de significado, pueden estar compuestas de palabras con una gran probabilidad de coincidencia, de no tener en cuenta este factor, una frase de este tipo sería aceptada como válida en el sistema de traducción a pesar de ser errónea.

Para solventar este problema introduciremos el concepto de **stopwords**. Entendemos por **stopwords** aquellas palabras o signos de puntuación que son muy comunes en el texto, como ya hemos dicho su presencia influye aumentando considerablemente la probabilidad de que dos frases sean paralelas. Así pues elaboramos listas de las palabras más comunes para cada idioma y no las consideraremos cuando calculemos la probabilidad IBM1 entre frases.

La idea es poder realizar una comparativa de dos criterios de selección de frases, para así poder analizar que frases se han descartado según cada método y ver con qué método obtenemos mejores resultados.

3.2. Planteamiento mediante el PER

Análogamente al método basado en el modelo IBM1, utilizaremos otra herramienta para la selección de frases basada en el análisis del PER.

El sistema basado en PER consiste simplemente en utilizar el sistema de traducción inicial, con él traduciremos la parte fuente del corpus bilingüe y evaluaremos dicha traducción con la parte destino del corpus bilingüe.

Dada una herramienta que calcula el PER, eliminaremos los pares de líneas cuyo PER sea peor que el del resto. Determinamos el umbral PER entre frases aceptadas y eliminadas de tal manera que eliminemos el mismo número de frases que eliminábamos con el criterio de selección del IBM1.

A priori podemos suponer que el coste computacional de este sistema va a ser considerable, puesto que requiere de la traducción de todo el texto de entrenamiento, cuyo tamaño siempre es extenso, y la posterior evaluación de la traducción línea a línea que también resulta un proceso lento.

4. PROCESO EXPERIMENTAL

Utilizando el criterio IBM1 y el criterio PER haremos un análisis de las probabilidades de frases paralelas y una posterior eliminación de las frases que consideremos que son de peor calidad. Evaluaremos automáticamente mediante las medidas WER, PER, BLEU y NIST.

4.1. Datos

Realizamos el proceso de análisis para el corpus del proyecto TC-STAR [2] de castellano a inglés que utiliza los textos del *European Parliament Plenary Sessions* (EPPS). La Tabla 1 presenta las estadísticas del corpus de entrenamiento. Para evaluar se utilizó el test oficial correspondiente a la segunda evaluación del TC-STAR.

CORPUS	orac.	pal.	vocab.	Lmax	Lmean
train.eng	1,3M	37,0M	109,8k	100	27.3
train.spa		39,5M	147,6k	110	29.1

Tabla 1. *Estadísticas del corpus.*

4.2. Detalles del sistema de traducción

Alineamiento. Mediante la aplicación GIZA++ [2]. se realiza el alineamiento de los textos bilingües paralelos del material de entrenamiento, ejecutándose 4 iteraciones del modelo IBM1, 5 iteraciones del modelo HMM 3 iteraciones del modelo IBM4 y ninguna del modelo IBM3. Se obtiene el alineamiento en las dos direcciones de traducción: tomando alternativamente uno y otro idioma como lenguas fuente. A partir de estos dos alineamientos básicos, se obtienen los alineamientos unión e intersección de los mismos, definidos respectivamente,por los conjuntos unión e intersección de los enlaces establecidos en los alineamientos básicos.

Se eliminan los pares bilingües en el que una de las oraciones contenga más de 50 palabras o en el que el cociente entre el número de palabras de una y otra oración exceda 2.4 (fertilidad superior a 2.4)

Selección de tuplas. Una vez obtenido el alineamiento unión se procede a la segmentación en tuplas del material de entrenamiento. A efectos de simplificar el sistema de traducción, el vocabulario de tuplas se limita a aquellas que tengan una longitud máxima de 15 palabras tanto en el lenguaje fuente como en el lenguaje destino.

[2] *www.tcstar.org*

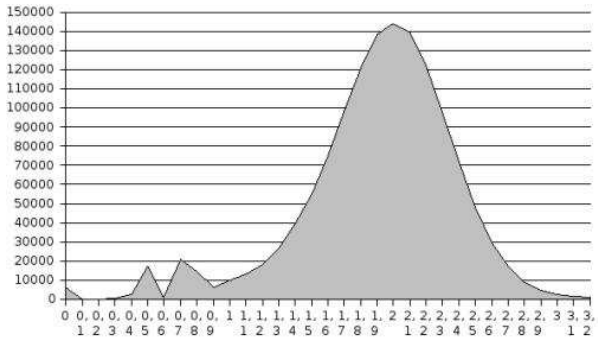


Figura 2. *Modelo IBM1 con Stop Words.*

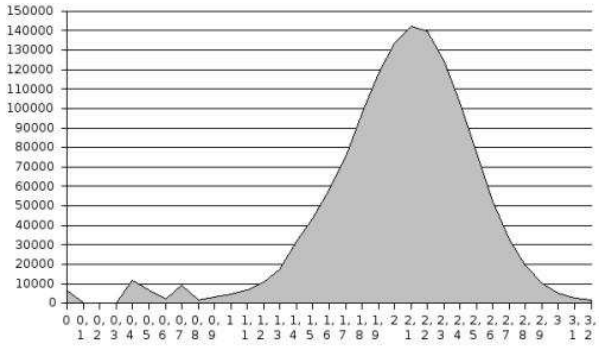


Figura 3. *Modelo IBM1 inverso con Stop Words.*

Estimación del modelo. Para estimar el modelo se utiliza la herramienta SRILM [3]. En este proceso se limita el vocabulario del modelo del lenguaje bilingüe a las tuplas seleccionadas, al que se añade una traducción (tupla) para todas aquellas palabras que no aparezcan solas en ninguna tupla (por lo que no se podrían traducir si en el test apareciesen en un contexto distinto a los existentes en el material de entrenamiento). Estas tuplas de traducción para las palabras ”incrustadas"(“embedded“) son generadas a partir del alineamiento intersección. Como técnica de suavizado se utiliza el método de Kneser-Ney e interpolación lineal (Kneser and Ney, 1995). El modelo generado fue un trigrama (N=3) de tuplas.

4.3. Aplicación del filtrado estadístico

4.3.1. Distribución de probabilidad

A continuación presentamos los resultados de calcular el modelo IBM1 sobre cada una de las oraciones paralelas en el corpus. Las Figuras 2 y 3 muestran el número de oraciones (eje Y) con una determinada probabilidad de ser paralelas (la probabilidad está expresada en logaritmo negativo).

Hemos eliminado la aportación probabilística de las stopwords, en concreto seleccionamos las 30 palabras más comunes en cada idioma.

Corpus	Umbral dir	Umbral inv	Eliminad.
C1	2.51	2.62	12 %
C2	2.7	2.8	5 %

Tabla 2. *Umbrales propuestos y frases eliminadas.*

4.3.2. Umbrales propuestos para la selección de frases.

Dadas las Figuras anteriores, se trata de experimentalmente seleccionar un umbral de probabilidad. En nuestro caso, la probabilidad está expresada en logaritmo negativo, con lo cual, eliminaremos todas las frases que estén por encima del umbral escogido. El hecho de filtrar el corpus original nos da lugar a un nuevo corpus. La Tabla 2 muestra un par de corpus generados (C1 y C2) a partir de ciertos umbrales. Asimismo, se muestra el % de frases eliminadas en cada corpus.

Realizamos la evaluación para ver que resultados obtenemos utilizando los nuevos modelos provenientes de los corpus tratados (ver Tabla 3).

Tipo	Corpus Inicial	C1	C2
BLEU score	43.33	44.52	44.20
NIST score	9.6	9.76	9.72
PER score	31.15	30.98	31.15
WER score	41.41	40.65	40.83

Tabla 3. *Resultados Corpus entrenamiento inicial,corpus C1 y corpus C2*

4.3.3. Criterio selección OR.

Este criterio consiste en admitir una frase como válida si alguna de las probabilidades IBM1, ya sea la directa o la inversa, supera un determinado umbral.

Establecemos un umbral que nos permita eliminar el mismo número de oraciones que en el corpus C1. Ello lo conseguimos estableciendo un umbral tanto directo como inverso de 2.4. Construimos el modelo con el nuevo corpus filtrado (C3) y obtenemos los resultados que se muestran en la Tabla 4.

Tipo	C3	C4
BLEU score	44.50	44.37
NIST score	9.75	9.73
PER score	31.01	31.86
WER score	40.68	40.71

Tabla 4. *Resultados Corpus entrenamiento C3 y C4*

4.3.4. Resultados obtenidos con el sistema PER.

Vamos a utilizar la eliminación de frases mediante el PER con el fin de comprobar los resultados obtenidos mediante el modelo IBM1. Eliminamos el 12 % de las frases mediante este sistema y obtenemos un corpus al que denominaremos C4 que procedemos a evaluar como hemos hecho en cada caso (ver Tabla 4).

Podemos observar que los resultados, aunque mejoran los obtenidos en el sistema inicial, son inferiores a los obtenidos utilizando en el modelo creado a partir del corpus C1, que es el que proporciona mejores resultados.

4.3.5. Análisis de los resultados

En las frases descartadas, vemos la influencia de la longitud de las frases en la evaluación del modelo IBM1, en este caso la media de las palabras por frase del corpus original es de 20, mientras que en las frases descartadas es de 40.

Como ejemplos de frases descartadas podemos citar:

EN línea 70:*We see that the French Government has sent a mediator.*

ES línea 71:*Vemos que el Gobierno francés ha enviado a un mediador.*

donde hay un desplazamiento de línea de un texto a otro. En otros casos encontramos traducciones que en un contexto pueden tener cierto sentido, pero a nivel de corpus de entrenamiento no aportan una información adecuada, por ejemplo encontramos:

- *This is where the main obstacles lie.* se traduce como - *Las consideraciones van especialmente en esta dirección.*

- *To conclude , let me ask what lessons we should be learning.* se traduce como - *Permítanme que finalice mi intervención con una serie de propuestas.*

5. CONCLUSIONES

Se han presentado dos técnicas de filtrado estadístico de corpus bilingües. La técnica basada en el Modelo IBM1 obtiene resultados ligeramente superiores a la técnica basada en el PER. Incorporar la técnica de filtrado estadístico permite reducir el ruido del corpus de entrenamiento y como consecuencia, se reduce el coste computacional y se mejora la calidad del sistema de traducción. En los experimentos que hemos presentado se mejora en más de 1 punto BLEU.

6. BIBLIOGRAFÍA

[1] J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. R. Fonollosa, y M. R. Costa-Jussà, "N-gram-based machine translation," *Computational Linguistics, Association for Computacional Linguistics.*, vol. 32, no. 4, pp. 527–549, 2006.

[2] F.J. Och y H. Ney, "A systematic comparison of various statistical alignment models," vol. 29, no. 1, pp. 19–51, March 2003.

[3] A. Stolcke, "Srilm - an extensible language modeling toolkit," September 2002.

CONFERENCIAS INVITADAS

VOICE CONVERSION: STATE OF THE ART AND PERSPECTIVES

Yannis Styilianou

University of Crete, Grece

RESUMEN

Voice Transformation refers to the various modifications one may apply to the sound produced by a person, speaking or singing. In other words, Voice Transformation aims at the control of non-linguistic information of speech signals such as voice quality and voice individuality. Voice Transformation covers a wide area of research from speech production modeling and understanding to perception of speech, from natural language processing, modeling and control of speaking style, to pattern recognition and statistical signal processing.

Voice Transformation was considered as a hot, novel and fast-growing topic in 1990s having as potential application the concatenated speech synthesis systems where new (virtual or target) voices could be created without requiring to pass through the quite expensive process of developing new voices. By that time, it was widely accepted that Voice Transformation systems were far from providing the required performance. With the recent developments in speech synthesis this need is more pronounced. There is an increasing demand for high quality Voice Transformation methods not only for creating target or virtual voices, but also to model various effects (e.g., Lombard effect), synthesize emotions, to make more natural the dialog systems which use speech synthesis etc.

In this talk I will review the state-of-the-art Voice Transformation methodology showing its limitations in producing good speech quality and its current challenges. Addressing quality issues of current voice transformation algorithms in conjunction with properties of the speech production and speech perception systems I will try to pave the way for more natural Voice Transformation algorithms in the future. Facing the challenges, it will allow Voice Transformation systems to be applied in important and versatile areas of speech technology. Besides speech synthesis, Voice Transformation has other potential applications in areas like entertainment, film, and music industry, toys, chat rooms and games, dialog systems, security and speaker individuality for interpreting telephony, high-end hearing aids, vocal pathology and voice restoration.

Yannis Stylianou Recibió el Diploma en Ingeniería Eléctrica en 1991 y el MSc y PhD en Procesado de Señal en la ENST de Paris, Francia, en 1992 y 1996 respectivamente. Desde 1996 hasta 2001 trabajó en AT&T Lab. Research (NJ, USA). En 2001 ingresó en los laoboratorios Bell (Lucent Technologies) en NJ (USA) (ahora Alcatel-Lucent). Desde 2002 trabaja comoProfesor Asociado en la Universidad de Creta, en el Departamento de Ciencias de la Computación, y como Investigador Asociado en el Laboratorio de Redes de Telecomunicaciones del Instituto de Ciencias de la Computación.

Es miembro del Comité Técnico del IEEE Speech and Language. Es editor asociado del EURASIP Journal on Speech, Audio and Music Processing y de las EURASIP Research Letters in Signal Processing, y vice-chair de la Acción COST 2103: Advandeced Voice Function Assessment". Fué editor asociado para la revista IEEE Signal Processing Letters y estuvo en comité de gestión (MC) de la Action COST 277: Nonlinear Speech Processing". Entre otros proyectos en el FP6, participó en la Red de Excelencia SIMILAR coordinando la tarea de fusion de las modalidades de voz y escritura. Tiene 9 patentes y es miembro de IEEE y de la Technical Chamber of Greece.

EMBODIED CONVERSATIONAL AGENTS IN VERBAL AND NON-VERBAL COMMUNICATION

Björn Granstrom

KTH - Royal Institute of Technology, Sweden

RESUMEN

In face-to-face communication both visual and auditory information play an obvious and significant role. Traditionally in phonetic research the auditory effects of speech production have been the primary object of study. However, when it comes to the non-verbal aspects of speech communication the primary nature of acoustics is not as evident, and understanding the interactions between visual expressions, dialogue functions and the acoustics of the corresponding speech presents a substantial challenge. Some of the visual articulation is for obvious reasons closely related to the speech acoustics (e.g. movements of the lips and jaw), but there is other articulatory movement affecting speech acoustics that is not visible on the outside of the face. On the other hand, many facial gestures used for communicative purposes do not affect the acoustics directly, but might nevertheless be connected on a higher communicative level in which the timing of the gestures could play an important role. The context of much of our research regarding these questions is to be able to create an animated talking agent capable of displaying realistic communicative behavior and suitable for use in conversational spoken language systems.

Useful applications of talking heads include aids for the hearing impaired, educational software, audiovisual human perception experiments, entertainment, and high quality audiovisual text-to-speech synthesis for applications such as news reading. The use of the talking head aims at increasing effectiveness by building on the user's social skills to improve the flow of the dialogue. Visual cues to feedback, turntaking and signaling the system's internal state are key aspects of effective interaction.

The focus of this paper is to present an overview of some of the research involved in the development of audiovisual synthesis to improve the talking head. Some examples of results and applications involving the analysis and modeling of acoustic and visual aspects of verbal and non-verbal communication are presented.

Granström, B., & House, D. (2007). Inside out - Acoustic and visual aspects of verbal and non-verbal communication (Keynote Paper). Proceedings of the 16th International Congress of Phonetic Sciences, Saarbrücken, 11-18.

Granström, B., & House, D. (2007). Modelling and evaluating verbal and non-verbal communication in talking animated interface agents. In Dybkjaer, l., Hemsen, H., & Minker, W. (Eds.), Evaluation of Text and Speech Systems (pp. 65-98). Springer-Verlag Ltd.

Beskow, J., Granström, B., & House, D. (2007). Analysis and synthesis of multimodal verbal and non-verbal interaction for animated interface agents. In Esposito, A., Faundez-Zanuy, M., Keller, E., & Marinaro, M. (Eds.), Verbal and Nonverbal Communication Behaviours (pp. 250-263). Berlin: Springer-Verlag.

Björn Granström joined KTH, The Royal Institute of Technology, in Stockholm, Sweden in 1969, after graduating as MSc in Electrical Engineering. After further studies in Phonetics and General Linguistics at Stockholm University he became Doctor of Science at KTH in 1977 with the thesis "Perception and Synthesis of Speech". In 1987 he replaced Gunnar Fant as Professor in Speech Communication. He has been the director of CTT, The Center for Speech Technology <http://www.speech.kth.se/ctt>, since its start in 1996. Granström has numerous publications in the speech research and technology area. The group is actively participating in many EU projects. His recent research interests include multi-modal communication systems using embodied conversational agents.

APLICACIONES DE LAS TECNOLOGÍAS DEL HABLA EN SISTEMAS CALL Y CAPT

Néstor Becerra Yoma

Laboratorio de Procesamiento y Transmisión de Voz
Departamento de Ingeniería Eléctrica
Universidad de Chile

RESUMEN

Se ha observado últimamente un gran interés en la comunidad internacional por el potencial de tecnologías de voz en las aplicaciones relacionadas con educación tales como CALL (Computer Aided Language Learning). CALL y CAPT (Computer Aided Pronunciation Training), en particular, se pueden considerar como frameworks interesantes para aplicar de modo masivo tecnología del habla. Sistemas CALL ofrecen varias ventajas sobre los métodos convencionales de clases presenciales con profesor: las lecciones se pueden preparar ad-hoc a una clase o a cada estudiante; los estudiantes, a su vez, pueden practicar ejercicios desde sus casas, laboratorio o en cualquier otro lugar en condiciones menos estresantes y embarazosas que en frente del profesor y de otros alumnos; aquellos estudiantes con dificultades de aprendizaje pueden disponer de apoyo extra para estudiar y practicar de modo más interactivo y dinámico que simplemente con un libro; y, finalmente, el problema de baja penetración en varias regiones de profesores debidamente entrenados para enseñar un segundo idioma es alivianado. Además, las tecnologías de voz tienen el potencial de proveer una retro-alimentación adecuada para corregir errores sin la necesidad imperativa de asistencia humana. Esta motivación es de especial importancia para motivar a practicar y aprender. Sin embargo, estas tecnologías sus propias limitaciones y estrategias del tipo "plug-and-play" tienden a fallar en nuestro campo.

En esta charla se discutirá el estado de avance de tecnologías CAPT, y se describirá el diseño y puesta en marcha de un sistema distribuido en Internet para la enseñanza de ingles como segundo idioma en Chile. El sistema usa la tecnología de reconocimiento de voz, basada en HMM, para la evaluación de calidad pronunciación, y para dar respuestas por voz en actividades de comprensión de texto y de asociación de palabras a significados. La evaluación de entonación se implementa separada de la de fonética utilizando estimación de pitch y alineamiento no lineal entre la elocución de test y la de referencia. La plataforma también ofrece actividades de dictado mediante las cuales el alumno debe transcribir palabras y frases. Algoritmos de programación dinámica permiten dar una nota en función del número de errores. Es interesante destacar que la plataforma propuesta puede ser vista como una etapa hacia IALL (Internet Aided Language Learning) una vez que el servicio se ofrece a través de la Internet y todo el procesamiento se realiza de modo centralizado. Finalmente, se presentan resultados prelimares de experimentos de usabilidad realizados con alumnos de un colegio público de Santiago.

Néstor Becerra-Yoma was born in Santiago, Chile, on September 15th, 1964. He received the Ph.D. degree from the University of Edinburgh, UK, and the M.Sc. and B.Sc. degrees from UNICAMP (Campinas State University), Sao Paulo, Brazil, all of them in Electrical Engineering, in 1998, 1993 and 1986, respectively. In 1998 and 1999, he was a post-doc researcher at UNICAMP and a full-time professor at Mackenzie University in Sao Paulo, Brazil. From 2000 to 2002, he was an Assistant Professor at the Department of Electrical Engineering, Universidad de Chile, in Santiago, where he is currently lecturing on telecommunications and speech processing, and working on robust speech recognition/speaker verification, dialogue systems and voice over IP. At the Universidad de Chile he has set up the Speech Processing and Transmission Laboratory (LPTV , Laboratorio de Procesamiento y Transmisión de Voz) to study speech technology applications on the Internet, education and telephone line.

Dr. Becerra-Yoma has been an Associate Professor since 2003 and is the author of 17 international journal articles and 30 conference papers. It is worth highlighting that almost all his publications are co-authored by his students. He has been the PI of two Fondef project (US$ 450.000), three Fondecyt projects (US$ 105.000) and two international cooperation projects. All these projects have been successfully executed, and have already finished or about to end. His results have got plenty of national press coverage in newspapers: deployment in schools of prototypes for English teaching using speech recognition and multimedia technology; and, speech recognition and vocal print technology for applications on telephony. He has supervised five M.Sc. theses and over 25

undergraduate dissertations in the last 7 years. He is also currently supervising 3 PhD students. Dr. Becerra-Yoma coordinated the commission that created the PhD program in Electrical Engineering at Universidad de Chile.

Dr. Becerra-Yoma was co-founder and elected the first chairman of the ISCA (International Speech Communication Association) Special Interest Group on Iberian Languages that already has over 100 members (professors, researchers and students) from Spain, Portugal, USA, Germany and Latin America. He has also been a chairman in several international conferences. He has been continuously invited to review papers in the most important international journal and conferences in speech and signal processing and is one of the guest editors of the special issue on Iberian languages in Speech Communications (Elsevier).

In the framework of his Fondef projects, Dr. Becerra-Yoma has organized several outreach events such as workshops, and ceremonies to officially launch the project and to deploy prototypes in schools. He has supervised the R&D activities that led to pre-competitive and competitive prototypes that are currently being evaluated by industry partners. In 2005, He got a 3rd place out of 20 proposals in a national contest for patenting.

His research interests include speech processing for telephony and education, real time Internet protocols, QoS, and usability evaluation of interfaces. Professor Becerra-Yoma is a member of the Institution of the Electrical and Electronic Engineers, and the International Speech Communication Association.

THIRD-GENERATION CONVERSATIONAL INTERFACES

Giuseppe Riccardi

University of Trento, Italy

RESUMEN

Communicating with machines is becoming pervasive to the point we rely entirely on them to find (vital) information over the web, perform on-line (trans)actions and communicate with people speaking different languages. In the last decade we have seen tremendous research and technology advancement in the speech and text based interfaces. We are now faced with the problem of overcoming their limitations and investigate multimodal input, adaptive interfaces, communicative paradigms and tame task complexity. In this talk we discuss research towards third-generation conversational interfaces.

Prof. Riccardi received his Laurea degree in Electrical Engineering and Master in Information Technology, in 1991, from the University of Padua and CEFRIEL Research Center, respectively. From 1990-1993 he collaborated with Alcatel-Telettra Research Laboratories (Milan, Italy). In 1995 he received his Phd in Electrical Engineering from the Department of Electrical Engineering at the University of Padua, Italy. From 1993-2005, he worked first at AT&T Bell Laboratories and then AT&T Labs-Research where he worked in the Speech and Language Processing Lab. In 2005 joined the faculty of Engineering at University of Trento (Italy) and is affiliated with the interdisciplinary Department of Information and Communication Technology and Center for Mind/Brain Sciences. He is the founder and director of the Adaptive Multimodal Information and Interfaces (AMI2) Lab.

Prof. Riccardi's research on stochastic finite state machines for speech and language processing has been applied to a wide range of domains for task automation. He and his colleagues designed the state-of-the-art AT&T spoken language system ranked first in the 1994 DARPA ATIS evaluation. He pioneered the speech and language research in spontaneous speech for the well-known "How May I Help You?" research program which led to breakthrough speech services. His research on learning finite state automata and transducers has lead to the creation of the first large scale finite state chain decoding for machine translation (Anuvaad).

Prof. Riccardi has co-authored more than 80 papers and 25 patents in the field of speech processing, speech recognition, understanding and machine translation. His current research interests are language modelling and acquisition, language understanding, spoken/multimodal dialog, affective interfaces, machine learning and machine translation.

Prof. Riccardi has been on the scientific committee of EUROSPEECH, INTERSPEECH, ICASSP, NAACL and ACL an EACL. He has co-organized the IEEE ASRU Workshop in 1993, 1999, 2001 and General Chair in 2009. He has been the Guest Editor of the IEEE Special Issue on Speech-to-Speech Machine Translation. He has been a founder and Editorial Board member of the ACM Transactions of Speech and Language. He is elected member of the IEEE SPS Speech Technical Committee (2005-2008). Prof. Riccardi has been member of New York Academy of Science and is senior member of IEEE, ACL, ISCA.

Prof. Riccardi have received many national and international awards and more recently the Marie Curie Research Excellence grant by the European Commission.