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.

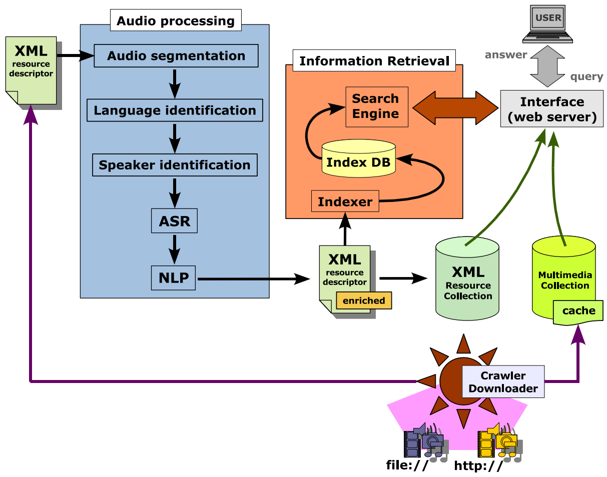


Figura 1. The architecture of the SDR system.

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 multi-words 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:

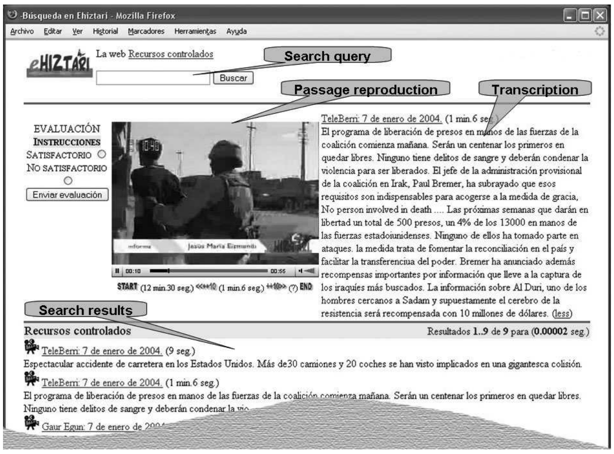
$$S(q,j) = \frac{\sum_{k=1}^{T} w_{qk} w_{jk}}{\sqrt{\sum_{k=1}^{T} w_{qk} \sum_{k=1}^{T} w_{jk}}} \quad , \qquad (1)$$

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



La web Recursos controlados

EVALUACIÓN
INSTRUCCIONES
SATISFACTORIO ○
NO SATISFACTORIO ○

START (12 min.30 seg.) <<++10 (1 min.6 seg.) ++10>> (?) END

TeleBerri: 7 de enero de 2004. (1 min.6 seg.)
El programa de liberación de presos en manos de las fuerzas de la coalición comienza mañana. Serán un centenar los primeros en quedar libres. Ninguno tiene delitos de sangre y deberán condenar la violencia para ser liberados. El jefe de la administración provisional de la coalición en Irak, Paul Bremer, ha subrayado que esos requisitos son indispensables para acogerse a la medida de gracia, No person involved in death .... Las próximas semanas que darán en libertad un total de 500 presos, un 4% de los 13000 en manos de las fuerzas estadoiunidenses. Ninguno de ellos ha tomado parte en ataques. la medida trata de fomentar la reconciliación en el país y facilitar la transferenciua del poder. Bremer ha anunciado además recompensas importantes por información que lleve a la captura de los iraquíes más buscados. La información sobre Al Duri, uno de los hombres cercanos a Sadam y supuestamente el cerebro de la resistencia será recompensada con 10 millones de dólares. (less)

Recursos controlados Resultados **1..9** de **9** para (**0.00002** seg.)

TeleBerri: 7 de enero de 2004. (9 seg.)
Espectacular accidente de carretera en los Estados Unidos. Más de30 camiones y 20 coches se han visto implicados en una gigantesca colisión.

TeleBerri: 7 de enero de 2004. (1 min.6 seg.)
El programa de liberación de presos en manos de las fuerzas de la coalición comienza mañana. Serán un centenar los primeros en quedar libres. Ninguno tiene delitos de sangre y deberán condenar la vio

Gaur Egun: 7 de enero de 200

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.