SPANISH KEYWORD SPOTTING SYSTEM BASED ON FILLER MODELS, PSEUDO N-GRAM LANGUAGE MODEL AND A CONFIDENCE MEASURE

Javier Tejedor and José Colás.
Email: javier.tejedor@uam.es, jose.colas@uam.es

Departamento de Ingeniería Informática, EPS Universidad Autónoma de Madrid. Carretera de Colmenar Viejo, km 15. 28049 Madrid, Spain.

ABSTRACT

In order to organize efficiently lots of hours of audio contents such as meetings, radio news, search for spoken keywords is essential. An approach uses filler models to account for non-keyword intervals. Another approach uses a large vocabulary continuous speech recognition system (LVCSR) which retrieves a word string and then search for the keywords in this string. This approach yields high performance but it requires a lot of training data and costly computation. In this paper we present several filler models and a confidence measure explored in a Spanish keyword spotting system. We will also investigate different weights in the grammar used for the language modelling in the keyword spotting system in order to achieve the best results. The keyword technique used is based on Hidden Markov Model (HMM). Test results are reported on a set of data from the geographic corpus of Albayzin speech data base containing 80 keywords taken from the words which most times occurs in the corpus sentences.
KEYWORDS: Speech recognition, word spotting, filler models, pseudo N-gram, confidence measure.

1. INTRODUCTION

Our task in the keyword system developed is to detect a set of keywords from a speech stream. The main challenge in word spotting techniques is to achieve the highest possible keyword detection rate while minimizing the number of false alarm keywords. That's why it is not sufficient to model only the keywords; models of the background are also required. Most of the wordspotters developed for years were variants of HHM-based, continuous speech recognition systems [1,2,3,4]. In such systems, the non-keyword intervals were represented by a variety of filler models, varying from a few phonetic or syllabic fillers to whole words [5]. Several confidence measures have been proposed by authors in order to minimise the false alarm keywords rate [6,7]. The benefits of incorporating a language model for the transitions between the keywords and the filler models were also evaluated for some of the systems [1,2,4] and were found to be substantial. Normally, the large vocabulary continuous speech recognition (LVCSR) systems with a language model component outperform any other configuration. However, the LVCSR approach to word spotting has two important disadvantages, (1) it is computationally very expensive, and (2) it tends to be domain dependent, requiring knowledge of the full vocabulary, and a large body of training data.

In this paper we describe our investigation into the use of different background or filler models in order to compare them in our Spanish keyword spotting system as well as the use of different language models similar to which are proposed in [8] to achieve the best results. We will also have included a confidence measure in order to minimise the false alarm keywords rate with no decreasing of the keyword recognition rate.

Our paper is organised as follows: Section 2 describes the experimental framework used in our system. Section 3 describes the experiments developed with Albayzin data base. Section 4 describes the discussion about the results achieved. Section 5 describes the conclusions and future work and Section 6 contains the references.

2. EXPERIMENTAL FRAMEWORK

2.1. System Description

Some systems [9] described segment-based wordspotters. Others [10] present a hybrid word / phoneme-based approach for word spotting. In our case, the Spanish keyword spotting system, based on the decoder provided by the HTK tool [11], is developed in two recognition processes. The first one is based on a phonetic decoding achieving the phoneme sequence recognized. This process is necessary in order to include the confidence measure system in our keyword spotting system. The second recognition process is the keyword spotting achieving the list of keywords recognized by the Vierbi algorithm included in the decoding process of HTK tool. The system architecture can be seen in Figure 1. The phonetic decoder process is explained in section 2.7. The keyword spotting process is explained in the section 2.8. The confidence measure system which is

composed by the phoneme performance estimator and the confidence measure module which retrieves the final output is explained in the section 2.9.

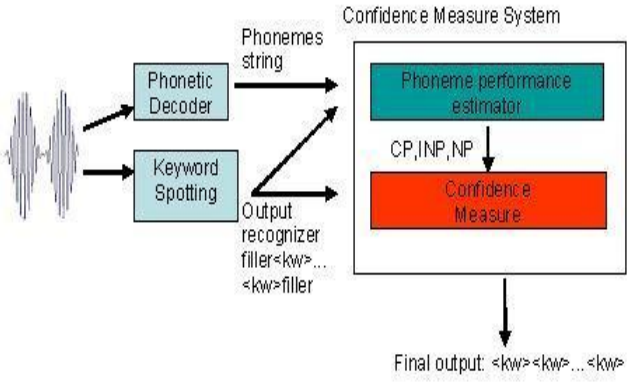

Figure 1. *Keyword Spotting System architecture*

kw denotes a keyword and filler denotes a filler model in our keyword spotting arquitecture.

The recognition network for the wordspotter is shown in Figure 2 for N keywords as well as M filler models.

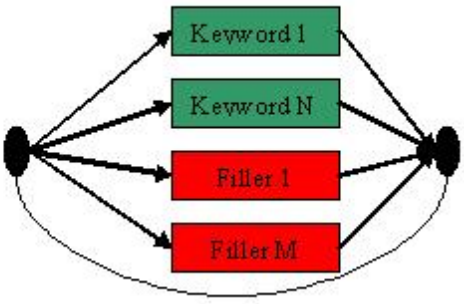

Figure 2. *Recognition network for wordspotting system*

Any transition between keywords and filler models is allowed as well as self transitions for both keywords and fillers. This configuration allows multiple keywords to exist in a single utterance, as well as multiple instances a keyword within the same utterance. For the experiments described in the next section, we used 3, 10, 25 and 49 filler models with 80 keywords.

2.2. Signal Representation and Features

For each of the utterance in the phonetic corpus for training in Albayzin data base as well as each one of the subset of the geographic corpus in the same data base used for the test of the system, 12 Mel-Frequency Cepstral Coefficients (MFCCs) plus energy and their first and second derivates were extracted to characterize the input signal. So a set of 39 MFCCs were used to represent the input signal. The initial signal has 16 Khz and 16 bits. Next, we pre-emphasise the speech signal and Hamming window is taken. The window size is 25 msec and once every 10 msec features are computed for a frame of 25 msec speech samples.

2.3. Training

We have used the phonetic corpus for training in Albayzin data base. In order to build the different filler models we have trained various filler models:

In a first step, 50 Allophones Models (AM) context-independent were trained taking into account the different phonological rules in Spanish language. These models include the initial silence, the short silence and the final silence. They were also used to run the phonetic decoder in the first process of the system.

In a second step, the theoretical set of Phonemes Models (PM) which are 26 and do not cover all the phonological rules in Spanish language were trained. They include the initial silence, the short silence and the final silence.

In a third step, we have clustered all the phonemes (Broad phonetic Models) (BM) in Spanish language according to the following classes were trained: nasals, closed vowels, opened vowels, median closed vowels, deaf fricatives, deaf oclusives, sound oclusives and liquids. The initial silence, the short silence and the final silence are also include in this configuration.

In the last step we have trained one Average Phonemes Model (APM) containing all the phonemes. The initial silence, the short silence and the final silence also belong to this configuration.

Each phoneme is modelled as CHMM (Continuous Hidden Markov Model) and each phoneme model is defined as 3-state left-to-right (no skip path) model, 15 mixtures, each of the two silence models, one for the initial silence and another for the final silence, as 3-state model, 15 mixtures and the short silence, as 1-state model, 15 mixtures (skip path). The phonetic units trained for all these cases were context-independent phones.

2.4. Keyword models

The keyword models were represented by concatenations of phonetic units. They were expanded into a pronunciation network based on their phonetic transcription in Spanish. A grapheme-to-phoneme translator was used to do it. It was also added the short silence to the end of each keyword. In our case it is not necessary to exist this short silence at the end of each keyword pronounced by the speakers due to the short silence has a skip path. The phonetic units used to represent these keyword models were the 50 Allophones Models (AM).

2.5.Filler models

We have investigated four different filler models in our system in order to compare the results achieved with each one. As filler models, we have taken the models resulted from the training of the system. We have compared the AM (Allophones Models) consists

of 49 phonemes models, PM (Phonemes Models) consists of 25 phoneme models, BM (Broad phonetic Models) consists of 10 phonemes models and the APM (Average Phonemes Models) consists of 3 phonemes models. In these filler models we have not included the short silence model which has been included at the end of the keyword as it was explained in the previous section.

2.6.Language Modelling for Keyword Spotting

As it is well-known, filler models in this kind of systems are based on the phoneme network. Keyword spotting systems tend to retrieve the sequence of phonemes instead of the keyword associated to.

To deal with this problem, a pseudo N-gram has been introduced in order to give different weights to the filler models and the keywords. We have investigated different weights combinations (1,2,4,6,8,9,10,11 and 12 represented in X axis in Figure 3 and 4) in order to achieve the best results.

2.7 Phonetic decoder process

First of all, a phonetic decoder based process using the Viterbi algorithm in HTK tool was developed in order to achieve the sequence of phonemes in the set of the test sentences used in the keyword spotting system. This phonetic decoder will allow us to define the confidence measure system explained in the section 2.9. The phonetic decoder will use a phonetic bigram as language model. This grammar was built taking the whole geographic corpus in Albayzin data base. The set of phonemes used in this phase was the 50 Allophones Models (AM) explained in the section 2.3.

2.8.Keyword Spotting process

The Viterbi algorithm proposed in the decoding phase in the HTK tool is used to find the best path through the labelled segment network, with the pronunciation network and the language model serving as constraints. The output is a continuous stream of fillers and keywords. The confidence measure system proposed in the next section will confirm or do not the keyword proposed by the Viterbi algorithm.

2.9. Confidence measure system

Some authors have proposed several confidence measures in order to minimise the false alarms rate in keyword spotting [6,7]. Here, we propose a new measure confidence to achieve the same benefit. We have considered only the keyword retrieved by the decoder as correct if these conditions are true:

1) $CP > INP + F + abs (NP-CP-INP)$.

2) $CP > (NP / 2)$.

where CP is the number of correct phonemes retrieved by the first process (phonetic decoder process) of the system in the samples where the second phase has detected the keyword,

INP is the number of incorrect phonemes of the first phase,

F is a factor which represents the difference allowed between correct and incorrect phonemes,

abs is the absolute value and

NP is th number of the phonemes of the word.

The number of correct and incorrect phonemes in the phonetic decoder process as well as the number of phonemes of each keyword are calculated in the phoneme performance estimator module of our system.

3) If the number of the phonemes of the keyword is less or equal than a constant K1, the phonemes recognized must be equal to the keyword phonetic transcription. It allowed us to deal with short words and classify them better as correct or false alarm keywords.

The steps 1), 2) and 3) are the confidence measure module of our system. It retrieves the keywords that keep these three steps and also eliminates the filler models retrieved by the keyword spotting system to achieve the final output.

A correct phoneme is defined as the phoneme recognized belongs to the keyword recognized between two positions previous the correct position and two positions next to it. An incorrect phoneme is defined as the phoneme recognized does not belong to the keyword recognized between two positions previous the correct position and two positions next to it. We will also have grouped the different phones which represent a same phoneme to not consider as errors the confusability between these phones. In this way, there are four phones for each vowel that are represented by the appropriate vowel, the two phones for the phonemes ‘b’ and ‘d’are also grouped in phoneme ‘b’ and ‘d’, etc.

3. EXPERIMENTS

3.1 Task

All experiments were performed in the geographic Albayzin domain. The task was the detection of 80 keywords pronounced by the speakers in this data base. The keyword consists of mountain, river and city names. They were chosen based on their high frequency of occurrence and the observation that they may constitute a sufficient set for a hypothetical spoken language system that will allow anybody to make searchs in these geographic elements. The training set was composed by 4 speakers who told 200 sentences each one and 160 speakers who told 160 sentences each one. So 4800 sentence phonetically balanced were used to train the phonetic units explained in the section 2.3. The test set was composed by sentences which belong to the geographic corpus in Albayzin data base. 48

speakers told 50 sentences to get the 2400 sentences used to test the keyword spotting system.

3.2 Performance Measures

The performance of the proposed keyword spotting system was measured using these calculations: At first, we present a graphic (Accuracy Keyword Spotting) which shows the percent of correct keywords recognized by the system respect to all the keywords to be recognized. At second we will also present another graphic (FA rate) which shows the percent of false alarms respect to the all the keywords recognized. These two graphics are shown varying the probability of the language modelling explained in the section 2.6. A keyword is considered successfully if it belongs to the sentence of the speaker which is proccessing. All the experiments were run in a Intel Pentium IV-PC 3.00 Ghz, 480 MB RAM.

3.3 Phonetic decoder results

The experiments made in the phonetic decoder process allowed us to build the confidence measure. The vocabulary in this phase consisted of the 50 Allophones Models (AM) trained according to the different phonological rules in Spanish. In this phonetic decoder, a word penalty of 0.0 was inserted and a grammar scale factor of 2.0. These values were chosen due to previous experiences in phoneme recognition with which we achieved a 78% of phonemes recognized correctly. The result of this phase is the sequence of phonemes recognized for the test sentences in the geographic corpus in Albayzin.

3.4 Keyword Spotting with AM and PM as Filler Models

The keyword spotting with the Allophones Models (AM) as filler models used to build the keywords was developed first in order to have a measure to which compare the rest of the filler models investigated. The theoretical set of Phoneme Models (PM) contains less phones due to not all the phonological rules are considered, and each phone represents each phoneme in Spanish. The vocabulary in these experiments contains the 80 keywords to be recognized and the set of 49 AM for AM and 80 keywords and the set of 25 PM for PM, filler models explained in section 2.5. The keywords and the set of AM and PM were expanded into a pronunciation keyword. A final short silence is added at the end of each keyword. The output of this keyword spotting system is a sequence of phonemes and keywords. Two factors control the decision of hypothesizing a keyword instead of hypothesizing the underlying string of phones. The first one is related to the decoder phase of the Viterbi algorithm in HTK and refers to the factors *p* and *s* called as word insertion penalty and grammar scale factor. As in the phonetic decoder these values are set to 0.0 and 2.0 respectively. The second one is related to the different probabilities associated to each arc which represents transitions between phones, transitions between keywords and transitions between keywords and phones. In this way, we have varied this probability according to the explanation in the section 2.6. The AM Keyword Spotting achieved 71.52% as its best keyword recognition rate and 18.82% as false alarms rate whereas the PM Keyword Spotting achieved 78.96% as its best keyword recognition rate and 24.79% as false alarms rate.

3.5 Keyword Spotting with General Filler Models

We have designed two set of general models consisting of 10 phonemes and 3 phonemes called BM and APM as it was explained in the section 2.5. A language model similar to the previous section was used. In the first case, the BM Keyword Spotting used the 80 keywords and a set of 10 classes of phonemes in its vocabulary. In the second case, the APM Keyword Spotting used the 80 keywords and a set of 3 classes (a generic model, the initial silence and the final silence) in its vocabulary. The decision between retrieving a class of phoneme or the generic model depending on the general filler model or a keyword is equals to the previous section. In these two cases, we have achieved the following results: The BM Keyword Spotting achieved 84.33% as its best keyword recognition rate and 41.44% as false alarms rate whereas the APM Keyword Spotting achieved 83.90% as its best keyword recognition rate and 55.72% as false alarms rate. The Accuracy Keyword Spotting rate as well as the False Alarm (FA) rate showing all the results depending on the language modelling used are shown in the Figure 3 and Figure 4 respectively.

3.6 Factors in the Confidence Measure

We have proposed in the section 2.9 a confidence measure in order to minimise the false alarms rate in our system. We have considered three issues in order to consider a keyword is correct. The first one explained in the section 2.9 requires a factor introduced which allows us to be more restrictive when we have to accept a keyword as correct. This factor represents the difference allowed between the number of correct phonemes and the number of incorrect ones retrieved by the phonetic decoder associated to the keyword retrieved by the wordspotter to consider it as correct. Several experiments made before this process allowed us to set this factor to 1 in order to get a good rate between the correct keywords loosed and the false alarm keywords detected due to this factor. The another factor is the constant K1. In our experiments developed, a value 4 for it gave us the best results in the confidence measure system.

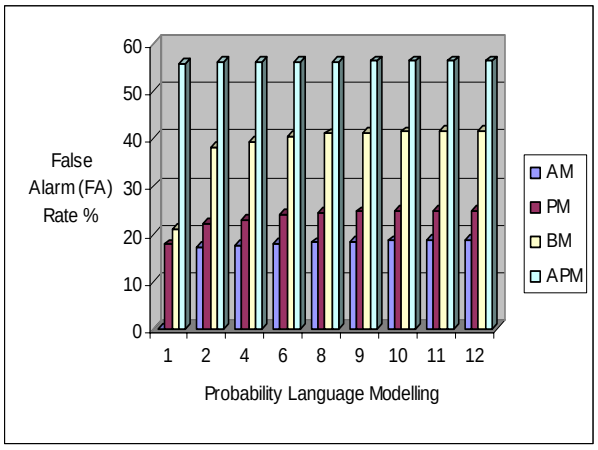

Figure 3. *Accuracy Keyword Spotting Rate*

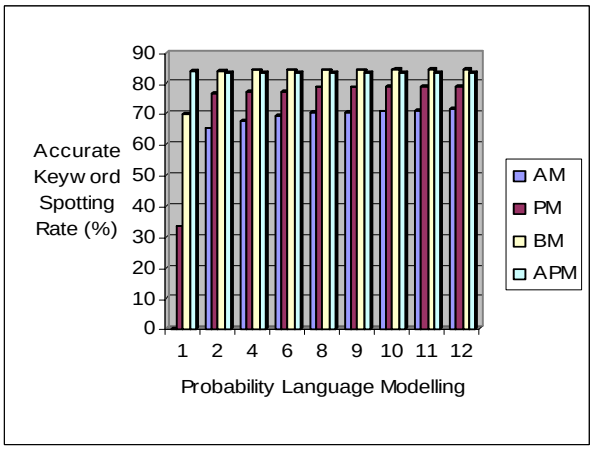

Figure 4. *False Alarms (FA) Rate*

4. DISCUSSION

There is a clearly correlation between the accuracy keyword spotting rate, the percent of correct keywords recognized of all the keywords in the test evaluation, and the false alarms rate, the percent of incorrect keywords recognized of all the keywords recognized in the test evaluation. The more keywords are recognized correctly, the more false alarms the system retrieves. As we can see in figure 3 and table 1, the BM filler model retrieves the best accuracy keyword spotting rate, with the 84% of keywords recognized. This rate is achieved when the Probability Language Modelling (PLM) varies from the range of 2 to 12. PLM in table 2 represents the Probability Language Modelling explained in the section 2.6. As we can see in the figure 4 and table 1, the less false alarms rate we achieve is 17.32 %, achieved with the AM model. In this model the accuracy keyword spotting in this case decreases to 65.18%. Depending on the kind of the global system, for example different audio web searchs, we can choose the best filler model for each situation. In case of the false alarm keywords rate does not important, we would choose the APM filler model due to it retrieves the best accuracy keyword spotting rate. In case of the false alarm keywords rate is important in the system, we would choose the AM model due to it minimise the false alarms rate, despite that retrieving the less accuracy rate. We can see two relevant things in this table. The first one is that when the probability given to the language model is the same for the filler model and for the keyword, the system does not retrieve any keyword for the AM filler model. It is due to the Viterbi algorithm prefers the sequence of phonemes instead of the keyword represented by the concatenation of these phonemes. That's why both the accuracy and the false alarms rate is 0%. The second one is that when we use a Average Phoneme Model (APM) as filler model, the false alarms rate increases a lot. That's why a unique model that represents all the phonemes in the sentences produces that the system prefers a keyword due to the great distance measured by the Viterbi algorithm between the samples in the sentences and the unique model. We can also see that when the language modelling probability to retrieve a keyword increases compared with the filler model one, both the accuracy keyword spotting rate and the false alarms rate also increases. It is due to the system prefers to retrieve a keyword instead of the sequence of phonemes because the probability assigned to it is greater.

PLM	AM Filler Model (Accur / FA)	PM Filler Model (Accur / FA)	BM Filler Model (Accur / FA)	APM Filler Model (Accur / FA)
1	0% / 0%	33,23% / 17,79%	70,18% / 21,05%	83,9% / 55,72%
2	65,18% / 17,32%	76,46% / 22,2%	84,09% / 38,1%	83,23% / 56,15%
4	67,87% / 17,55%	77,43% / 23,07%	84,21% / 39,45%	83,29% / 56,15%
6	69,15% / 17,94%	77,44% / 23,97%	84,21% / 40,37%	83,35% / 56,17%
8	70,37% / 18,21%	78,6% / 24,57%	84,27% / 41,09%	83,29% / 56,23%
9	70,55% / 18,34%	78,6% / 24,66%	84,27% / 41,24%	83,29% / 56,32%
10	70,79% / 18,52%	78,72% / 24,77%	84,33% / 41,44%	83,29% / 56,32%
11	71,22% / 18,72%	78,9% / 24,72%	84,33% / 41,57%	83,23% / 56,34%
12	71,52% / 18,82%	78,96% / 24,79%	84,33% / 41,67%	83,23% / 56,33%

Table 1. *Summary of Accuracy Keyword Spotting and False Alarms Rate for the Filler Models according to the Probability Language Modelling (PLM)*

5. CONCLUSIONS AND FUTURE WORK

There is a correlation between the accurate keyword spotting rate and the false alarms rate in our system due to an increase in the accurate keyword spotting produces an increase in the false alarms rate. A compromise between the accuracy keyword spotting and the false alarms rate depends on the global system where the keyword spotting process is integrated, but an acceptable compromise between the accuracy keyword spotting and the false alarms rate seems to be the PM Filler Model, achieving a 78,96% of accuracy and a 24,79% of false alarms rate.

As feature work we are going to try to reduce the false alarm keywords rate for the filler models investigating other confidence measures.

6.REFERENCES

[1] R. Rose, “Definition of subword acoustic units for wordspotting”, Proc. EUROSPEECH’93, pp, 1049-1052, September 1993.

[2] P. Jeanrenaude, K. Ng, M. Siu, J.R. Rohlicek, and H. Gish, “Phonetic-based word spotter: various configurations and application to event spotting”, Proc. EUROSPEECH’93, pp 1057-1060, September 1993.

[3] E. Lleida, J.B. Marino, J. Salavedra, A. Bonafonte, E. Monte, and A. Martinez, “Out of Vocabulary word modelling and rejection for keyword spotting”, Proc. EUROSPEECH’93, vol. 2, pp. 1265-1268, September 1993.

[4] M. Weintraub, “Keyword-spotting using SRI’s DECIPHER large-vocabulary speech recognition system”, Proc. ICASSP’93, vol. 2, no. 2, pp. 463-466, April, 1993.

[5] H. Cuayáhuitl and B.Serridge, “Out-of-vocabulary Word Modeling and Rejection for Spanish Keyword Spotting Systems”, Proc. MICAI’02, vol. 2313, pp.156, 2002.

[6] T. Schaaf, T. Kemp, “Confidence Measures for Spontaneous Speech Recognition”, Proc. ICASSP’97, vol 2, pp. 887-890, April, 1997.

[7] S. Cox and R. Rose, “Confidence Measures for the Switchboard database”, Proc. ICASSP’96, vol. 1, no. 1, pp. 511-515, May, 1996.

[8] Joo-G. Kim, Ho-Y. Jung and Hyun-Y. Chung, “A Keyword Spotting Approach based on Pseudo N-gram Language Model”, Proc. SPECOM’04, pp.156-159, September 2004.

[9] A. S. Manos and V. W. Zue, “A segment-based wordspotter using phonetic filler models”, Proc. ICASSP’97, vol. 2, no. 2, pp. 899-902, April 1997.

[10] P. Yu, F. Seide, “A hybrid word / phoneme-based approach for improved vocabulary-independent search in spontaneous speech”, Proc. ICSLP’04, vol. 13, no. 5, pp. 635-643, October 2004.

[11] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D, Povey, V. Valtchev and P. Woodland. *HTK Book v.3.2.1*, Microsoft Corporation and Cambridge University Engineering Department, December 2002.