A NEW KEYWORD SPOTTING ALGORITHM WITH PRE-CALCULATED OPTIMAL THRESHOLDS

J. Junkawitsch, L. Neubauer, H. Höge, and G. Ruske

Technical University of Munich, Arcisstr. 21, D-80290 Munich, Germany
Siemens Corp., Otto-Hahn-Ring 6, D-81730 Munich, Germany

ABSTRACT

Keyword spotting is a very forward-looking and promising branch of speech recognition. This paper presents a HMM-based keyword spotting system, which works with a new algorithm.

The first discussion topic is the description of the search algorithm, that needs no representation of the non-keyword parts of the speech signal. For this purpose, the computation of the HMM scores and the Viterbi algorithm had to be modified. The keyword HMMs are not concatenated with other HMMs, so that there is no necessity for filler or garbage models. As a further advantage, this algorithm needs only low computional expense and storage requirement.

The second discussion topic is the determination of a optimal decision threshold for each keyword. In order two decide between the two possibilities “keyword was spoken” and “keyword was not spoken”, the scores of the keywords are compared with keyword specific decision thresholds. This paper introduces a method to fix decision thresholds in advance. Starting with measured phoneme distributions, the score distributions of whole keyword models can be calculated. Furthermore, these keyword distributions form the basis of the computation of decision thresholds.

Tests with spontaneous speech databases yielded 73.9% Figure-Of-Merit when using context-dependent HMMs. The detection rate at 10 fa/kw/h comes to 80%.

1. INTRODUCTION

Keyword spotting becomes a very important branch of speech recognition. Latest research experiences show that it is nearly impossible to design a recognizer that covers all words uttered during the practical application. Therefore the aproach of only detecting keywords within any other words, sounds and noise without modeling these non-keyword parts of the utterance can be considered very promising. In contrary to other keyword spotting systems, which need some out-of-vocabulary representation, we renounce from explicit modeling the background.

The presented speaker-independent keyword word spotting system is able to indicate and classify predefined words within continuous speech. There are no restrictions concerning keyword occurencies: each utterance may contain any number of keywords. This keyword spotter, which is part of the big German speech understanding and translation project Verbmobil, is a phoneme based recognition system, so that all keywords are composed of phoneme Hidden Markov Models (HMM).

The paper presents a new algorithm which uses normalized scores during the Viterbi search. By a special treatment of the HMM scores, in this algorithm neither garbage nor filler nor silence models are necessary to represent the out-of-vocabulary parts of the utterance. Every keyword is allowed to start and to finish anywhere within the spoken sentence at any position. Since this method enables the recombination of different paths with different length within the Viterbi search, no additional computing load occurs. Backtracking is not necessary. The output of the algorithm yields normalized scores indicating the matching of the keywords at distinct time positions.

In order to decide between the two possibilities “keyword was spoken” and “keyword was not spoken” the normalized score of each word must be compared with a word specific threshold. A new technique is presented to calculate optimal values for these thresholds. Starting from the score distributions of the individual phonemes contained in a keyword, the total score distribution of the whole word is computed by length-weighted convolution of the single phoneme distributions. The score distributions are modeled by Gaussians, thus the threshold for each individual keyword can be calculated analytically.

This method enables the determination of the thresholds without the need of ever having spoken the keywords at all! This can be a decisive condition in practical applications where (like in our VERBMOBIL project) not enough training material exists for the individual keywords, whereas the phoneme models can be trained very well. Additionally, this new method allows a-priori estimation of the suitability and usefulness of each keyword in advance (without having spoken these words).

2. SYSTEM OVERVIEW

The keyword spotting system consists of three fundamental modules. The first module is concerned with signal analysis and feature extraction. The speech signal is sampled with a frequency of 16 kHz, and 400 samples at a time are combined to frames with a frame period of 10 ms. A total number of 30 mel-filtered cepstral coefficients and their time derivatives of first and second order are calculated, giving a feature vector with 64 components. This vector is optimized by the use of a linear discriminant analysis. The second module computes the emission probabilities for all states of all

HMMs. The third module contains the recognizer itself. Fig. 1 shows the basic structure of this module.

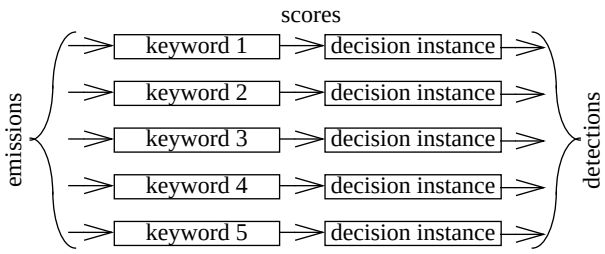

Figure 1: basic structure of the recognition module

The basic unit of the recognizer is a keyword model with a subsequent decision instance. For all keywords such basic units are set up, but no keyword can influence another one because all basic units are completely independent. Each keyword model is realized as a concatenation of phoneme HMMs, but no filler, silence or garbage models are attached at the beginning or the end. The first state of the keyword model has no predecessor state and the last state has no successor state. The search is done applying a modified Viterbi algorithm using specific normalized scores. The decision instance observes the score of the last state of a keyword model and provides classification by comparing local score minima with a decision threshold.

3. MODIFIED VITERBI ALGORITHM

For our purposes the standard Viterbi algorithm and the computation of scores have to be modified. The algorithm described below is based on local scores, which can be interpreted as a distance measure from a certain state to the best state with the highest emission probability. The term LSc_{t,s_j} means the local score of state s_j. This local score is defined as the difference between the negative logarithm of the emission probability $p(\underline{x}_t|s_j)$ of state s_j and the value of the best state at that time. So the best state always has $LSc_{t,s_j} = 0$, while all other states have scores greater than zero.

$$LSc_{t,s_j} = -\log p(\underline{x}_t|s_j) - min_i(-\log p(\underline{x}_t|s_i))$$

Since the first state of a keyword HMM is not concatenated with any predecessor, new search paths are allowed to start at any point in time. These paths don't take over any previous scores and begin with zero. Nevertheless, paths with different starting points are allowed to recombine in a proper way. For this purpose it is necessary to compare paths with different length within the search. This is achieved by using length normalized scores. The total amount of accumulated scores and penalties is devided by the number of states the path has passed. So two variables must be handled for each state s_j and each time instant t: the normalized score NSc_{t,s_i} and the length of the path L_{t,s_i}.

When A_{ij} is the penalty from state s_i to state s_j, the search algorithm can be notated in the following recursive form:

$$NSc_{t,s_i} = min_j\left(NSc_{(t-1),s_j} \cdot \left(1 - \frac{1}{1+L_{(t-1),s_j}}\right) + \frac{1}{1+L_{(t-1),s_j}} \cdot (A_{ij} + LSc_{t,s_j})\right)$$

$$L_{t,s_i} = 1 + L_{(t-1),s_k}$$

At any time instant t a new path may start with

$$L_{t,s_1} = 1 \text{ and } NSc_{t,s_1} = LSc_{t,s_1}$$

These equations can be interpreted as computing all possible paths leading to state s_j and selecting the best one. To calculate the normalized score of such a path, first reduce the weight (i.e. its length) of the predecessor score and then add the weighted local score of state s_j. All weigths depend on the length of the path. When the best predecessor state s_k is determined, its incremented length becomes the length of state s_j.

This algorithm needs low computional expense and storage requirements. So it's sufficient to allocate memory only for one column of the trellis. Tracing back thru the Viterbi matrix is not necessary. No HMMs for the non-keyword parts of the speech signals must be trained. Moreover this algorithm easily enables a time synchronous implementation of a keyword spotting system.

As a result of this calculation scheme, the normalized score of the last state of a keyword model, which serves as input for the decision instance, no longer is a monotonously increasing function in time. The score decreases when the HMM represents the speech signal well and increases when there is only bad matching. So the decision instance has to look for the minima of the score and evaluate them.

4. DECISION THRESHOLDS

The purpose of any decision instance is to determine which class the input belongs to. In the case of the keyword spotter the two possibilities "keyword was spoken" and "keyword was not spoken" have to be decided by observing the score of the last state of the keyword HMM. The decision rule is implemented by using a decision threshold. If the value of a local score minimum remains under this threshold, the occurence of the keyword is indicated, else the keyword is rejected. In contrary to using the same decision threshold for each keyword, experience shows that using individual thresholds improves recognition performance substantially.

In the following a method is presented to fix optimal thresholds for particular keywords automatically. The score of the last state of a HMM is considered to be a random variable. Its conditional probability density functions (cpdf) depend on the corresponding class of the sample. If these cpdf's are known for the two cases "keword was spoken" ω_1 and "keyword was not spoken" ω_0, decision thresholds can be calculated using distinct approaches.

4.1. Computation of cpdf's of keywords

In a phoneme based HMM recognition system a word model is composed of its phoneme models. The normalized score Y of a keyword HMM is interpreted as a random variable. Since each keyword HMM is composed of a sequence of phonemes, this

keyword score can be calculated using phoneme scores X. These phoneme scores are also thought to be random variables. Because of various mean phoneme lengths, weighting coefficients α_i must be used to take this effect into account.

$$Y = \alpha_1 \cdot X_1 + \alpha_2 \cdot X_2 + \ldots, \text{ with } \alpha_i = L_i / L_{total}$$

L_i is the mean duration of phoneme i measured in its number of frames, while L_{total} is the sum of all mean phoneme durations, representing the mean duration of the keyword.

The cpdf of the keyword score Y can be calculated as the convolution of the scaled cpdf's of the involved phonemes:

$$f_Y(y) = \frac{1}{\alpha_1} f_{X_1}\left(\frac{1}{\alpha_1} y\right) \otimes \frac{1}{\alpha_2} f_{X_2}\left(\frac{1}{\alpha_2} y\right) \otimes \ldots$$

In order to reduce computing expense and to get analytic solutions, all cpdf's are approximated by Gaussians $N(\mu, \sigma)$. In this way the keyword score Y is a Gaussian, too. Its mean value and standard deviation can be simply calculated using the means μ_i and the standard deviations σ_i of the phonemes:

$$Y = N((\alpha_1\mu_1 + \alpha_2\mu_2 + \ldots), \sqrt{\alpha_1^2\sigma_1^2 + \alpha_2^2\sigma_2^2 + \ldots})$$

These few parameters μ_i and σ_i must be measured once before calculating the cpdf's of any keywords. Figure 2 shows an example of experimentally measured and Gaussian approximated cpdf's of a phoneme for the conditions "phoneme was spoken" and "phoneme was not spoken". A comparison of calculated and experimentally measured cpdf's of keywords yielded quite good accordance of the means and standard deviations. It is remarkable that in general long keywords have small variances, whereas short keywords have great variances. Thus it is very promising to use word specific threshold especially when working with short and long keywords at the same time. Moreover, the suitability of a keyword may be predicted when the two cpdf's are known, because it is possible to estimate the detection rate and the corresponding number of false alarms.

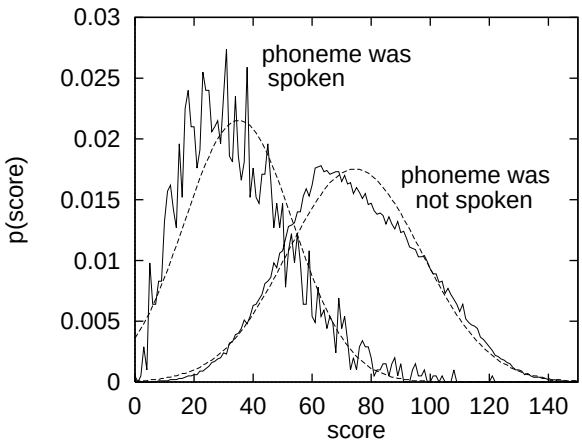

Figure 2: cpdf's for the two classes "phoneme was spoken" and "phoneme was not spoken". The measured densities are approximated by Gaussians.

4.2. Computation of thresholds

There are several methods to fix decision thresholds Y_{th} when the cpdf's for the two classes "keyword was spoken" and "keyword was not spoken" and the a-priori probabilities of the classes are known. One possibility is maximizing the detection rate and simultaneously minimizing the false alarms utilizing statistical hypothesis testing strategies like the Bayes' decision rule, the Neyman-Pearson rule, etc. But best results were achieved by prescribing a fix detection rate α. Since the cpdf is a Gaussian, the equation can be solved and Y_{th} calculated (Φ^{-1} means the inverse Gaussian distribution):

$$\alpha = \int_0^{Y_{th}} p(Y|\omega_1)dY \Rightarrow Y_{th} = \mu + \sigma \cdot \Phi^{-1}(\alpha)$$

5. RESULTS

The keyword spotting system was evaluated with speech data bases dealing with the scheduling of date appointments for meetings. These English dialogues are spontanous speech, spoken from native male and female speakers, containing a lot of silence parts and noise.

Two kinds of HMM were used to test the system: context-independent and context-dependent HMMs. Both were trained on a approximatly 12 hours speech data base consisting of Multicom 94.1, Multicom 94.2, Verbmobil CD 6.0 and Verbmobil CD 8.0.

The tests were done with the Multicom 94.4 corpus. This data base contains 614 utterances of 2.13 hours duration. The list below shows the test vocabulary. When using these 25 keywords, in Multicom 94.4 there are a total number of 941 keyword occurrencies.

> Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, morning, afternoon, tomorrow, evening, lunch, breakfast, meeting, office, schedule, appointment, vacation, calendar, seminar, conference, secretary, together, available, holiday

Figure 3 shows the receiver-operating-characteristics (ROC) for both context-independent and context-dependent HMMs using an equal decision threshold for each keyword. The figure of merit, a commonly used performance score, is defined as the average detection rate from 0 to 10 fa/kw/h (false alarms per keyword per hour). This evaluation yielded a FOM of 58.5% for context-independent and 73.9% for context-dependent HMMs.

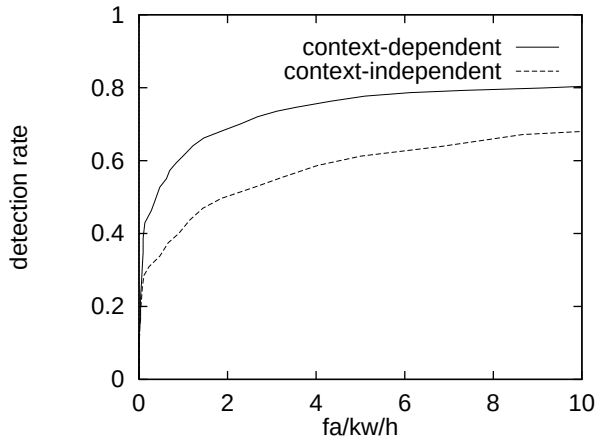

Figure 3: ROC for context-independent and context-dependent HMMs.

Evaluation of recognition performance using word specific thresholds was performed with context-independent HMMs. Table 1 shows some results, prescribing various detection rates α. The experimentally yielded detection rates are better than previously prescribed and all measured operating points lie above the ROC for the context-independent case.

prescribed α	detection rate	fa/kw/h
30%	34.0%	0.319
35%	39.7%	0.750
40%	48.0%	1.41
45%	53.3%	2.53

Table 1: Recognition performance using pre-calculated keyword specific decision thresholds and context-independent HMMs.

6. DISCUSSION

A new keyword spotting algorithm is presented that allows keyword detection without representation of the non-keyword parts of a utterance by silence, garbage or filler models. Every point in time a new Viterbi path is allowed to start. In order to be able to compare paths of different length, specific normalized scores are used within the search. A new approach is introduced, which allows calculation of word specific decision thresholds in advance. Starting with score distributions of phonemes, the cpdf's of keywords can be calculated and then applying different strategies decision thresholds can be fixed. Using only equal-length keywords, word specific decision thresholds don't become very effective. The advantageous application of word specific decision thresholds is the greater the more the vocabulary contains both short keywords and very long keyword-phrases.

In contrary to utilizing Bayes' decision rule for settling the boundaries, fixing a definit detection rate has proven to give best results for practical applications. Tests carried out with the Multicom 94.4 sponaneous continuous speech corpus yielded a FOM of 58.5% for context-independent and 73.9% for context-dependent HMMs. This notable result is attained with low computional and storage expense, since no out-of-vocabulary models are used. Moreover, this keyword spotting system doesn't need any language model. However, it may be promising to apply a specific kind of post-processing in order to avoid keyword overlaps and to consider statistically achieved regularities of the language.

7. ACKNOWLEDGEMENT

This work was funded by the German Federal Ministry for Research and Technology (BMFT) in the framework of the Verbmobil Project under Grant 01 IV 102/C. The responsibility for the contents of this study lies with the authors.

8. REFERENCES

1. Chang, E. I. u. R. P. Lippmann. Figure of merit training for detection and spotting. Advances in Neural Information Processing Systems 6, pp. 1019-1026, 1994.

2. Chigier, B.. Rejection and keyword spotting algorithms for a directory assistance city name recognition application. IEEE ICASSP, II, pp. 93-96, 1992.

3. Fukunaga, K. Introduction to statistical pattern recognition. Academic Press, New York and London, 1972.

4. Hofstetter, E. M. u. R. C. Rose. Techniques for task independent word spotting in continuous speech messages. IEEE ICASSP, II, pp. 101-104, 1992.

5. Marcus, J. N.. A novel algorithm for HMM word spotting, performance evaluation and error analysis. IEEE ICASSP, II, pp. 89-92, 1992.

6. Rohlicek, J. R., P. Jeanrenaud, K. Ng, H. Gish, B. Musicus u. M. Siu. Phonetic training and language modeling for word spotting. IEEE ICASSP, II, pp. 459-462, 1993.

7. Rose, R. C. u. D. B. Paul. A Hidden Markov Model based keyword recognition system. IEEE ICASSP, pp. 129-132, 1990.

8. Rose, R.C.. Keyword detection in conversational speech utterances using hidden Markov model based continous speech recognition. Computer Speech and Language, 9, pp. 309-333, 1995.

9. Sukkar, R. A. u. J. G. Wilpon. A two pass classifier for utterance rejection in keyword spotting. IEEE ICASSP, II, pp. 451-454, 1993.

10. Wilpon, J. G., L. G. Miller u. P. Modi. Improvements and applications for key word recognition using hidden Markov modeling techniques. IEEE ICASSP, pp. 309-312, 1991.