Comparison of Keyword Spotting Approaches for Informal Continuous Speech

Igor Szöke, Petr Schwarz, Pavel Matějka,
Lukáš Burget, Martin Karafiát, Michal Fapšo, Jan Černocký

Department of Computer Graphics and Multimedia
Faculty of Information Technology, Brno University of Technology, Czech Republic
szoke@fit.vutbr.cz

Abstract

This paper describes several approaches to keyword spotting (KWS) for informal continuous speech. We compare acoustic keyword spotting, spotting in word lattices generated by large vocabulary continuous speech recognition and a hybrid approach making use of phoneme lattices generated by a phoneme recognizer. The systems are compared on carefully defined test data extracted from ICSI meeting database. The acoustic and phoneme-lattice based KWS are based on a phoneme recognizer making use of temporal-pattern (TRAP) feature extraction and posterior estimation using neural nets. We show its superiority over traditional HMM/GMM systems. The advantages and drawbacks of different approaches are discussed.

1. Introduction

Keyword spotting (KWS) systems are used for detection of selected words in speech utterances. Searching for various words or terms is needed in spoken document retrieval which is a subset of information retrieval. KWS in spoken speech differs from searching in written text by the ambiguity – we are never able to make an exact "grep", and we have to count on inaccuracies of recognition systems. Therefore, the estimation of *confidence* of the found keyword is of crucial importance. The general scheme of KWS is shown in Figure 2. The confidence of keyword is computed as likelihood ratio (or log-likelihood difference):

$$C_{KW} = L_{left} + L(KW) + L_{right} - L_{bkg}, \qquad (1)$$

where L_{left} stands for the likelihood produced by a filler model preceding the keyword which models the beginning of utterance, $L(KW)$ is the likelihood of the keyword, L_{right} is the likelihood of the right filler model modelling the rest of the utterance. L_{bkg} is the likelihood of the utterance without considering the keyword – this term is needed for the normalization.

The search of keywords and computation of likelihoods can be done in several ways (the advantages and drawbacks are discussed later in respective sections):

- acoustic KWS, where the model of the keyword is composed of phoneme models at the time the keyword is entered.
- "grep" in the output (word string or lattice) of Large vocabulary continuous speech recognition (LVCSR).
- a hybrid approach making use of transcription of speech into discrete units – we use the search in phoneme lattices generated by a phoneme recognizer.

This paper deals with the comparison of these three approaches to KWS and their evaluation on informal continuous speech (recordings of meetings) within the AMI project. It is organized as follows: Section 2 contains the description of likelihood estimation systems for KWS – we have chosen this organization because often, one recognizer is used by more than one of the approaches from the list above. Section 3 describes the evaluation data. Sections 4–6 describe the methods of keyword spotting systems and their results. We conclude in Section 7.

2. Likelihood estimators for KWS

Two HMM/GMM systems were compared as generators of acoustic likelihoods for acoustic KWS (section 4): The first set of HMM/GMM models is trained on 10 h subset of ICSI meetings. The data was parameterized using 13 Mel-frequency cepstral coefficients with Δ and $\Delta\Delta$. Cross-word context-dependent HMMs were trained in standard way using HTK tools. The system is described in more detail in [1]. The set of models is denoted as **ICSI10h**. This system was also used to generate word-lattices for LVCSR-KWS (section 5). Here, we have used a tri-gram language model trained on a blend of CTS (3.5 Mwords), Hub4 (220 M) and ICSI (0.5 M) with trigram probabilities interpolated with weights 0.11, 0.20 and 0.69 respectively. CTS data contains about 277 hours of speech from Switchboard (249 h), Switchboard 2 - Cellular (15 h) and Call Home English (14 h).

Second set of HMM/GMM models used was from AMI-LVCSR system [2][1]. It is a full-fledged LVCSR system trained on conversational telephone speech (CTS) database. Speech is parameterized using 13 perceptual linear prediction (PLP) coefficients with Δ and $\Delta\Delta$. The features are normalized by cepstral mean and variance normalization. Context-dependent (CD) models are trained on CTS data. They are then adapted using MAP adaptation on full 41 h from ICSI meetings (downsampled to 8 kHz). When used in acoustic KWS, these models are denoted as **CTS277h**.

In acoustic KWS, we have also experimented with a phoneme recognizer based on temporal patterns (TRAPs) and neural networks (NN) (see Figure 1). This system [3] makes use of unconventional feature extraction technique based on long temporal trajectories: the temporal context of critical band spectral densities is split into left and right context (LC-RC)

This work was partially supported by EC project Augmented Multiparty Interaction (AMI), No. 506811 and Grant Agency of Czech Republic under project No. 102/05/0278. Jan Černocký was supported by post-doctoral grant of Grant Agency of Czech Republic No. GA102/02/D108.

[1] Developed in a joint effort of University of Sheffi eld (UK), University of Edinburgh (UK), Brno University of Technology (CZ), University of Twente (H) and IDIAP (CH)

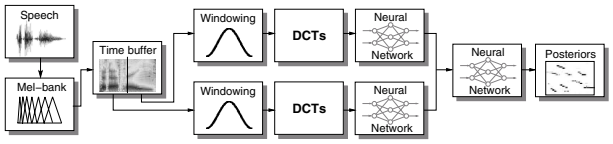

Figure 1: Phoneme recognizer with split temporal context

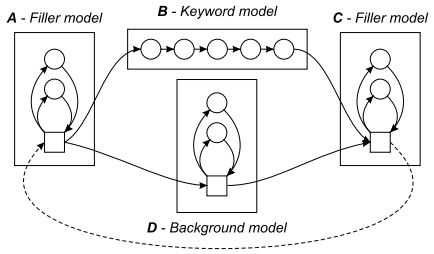

Figure 2: General scheme of keyword spotting.

parts. This allows for more precise modelling of the whole trajectory while limiting the size of the model (number of weights in the NN). Both parts are processed by DCT to de-correlate and reduce dimensionality. The feature vector created by concatenation of vectors over all filter bank energies is fed to NN. Two NNs are trained to generate phoneme posterior probabilities for left- and right-context parts respectively. Third NN functions as a merger and produces final set of phoneme posteriors. In [3], we have shown a substantial improvement for separate modelling of beginning, center and end of a phoneme by the NN. Therefore, all nets produce posteriors of 3-states per phoneme. For the acoustic KWS, these posteriors are transformed to eliminate sharp peaks around 0 and 1 in linear probability distributions [4]. The TRAP-NN system is trained on full train set 41.3 h of ICSI database, its results are denoted as **TRAP-NN40h**.

The TRAP-NN system is also used to generate phoneme lattices needed for experiments in section 6. Here, the phoneme posteriors were transformed to quasi-features and the lattice generation was done using the standard HTK decoder `HVite`. The setting of phoneme insertion penalty and of branching factor for lattice generation is discussed in section 6. No language model was used in the phoneme recognition.

3. Evaluation

Our keyword spotting systems were tested on a large database of informal continuous speech of ICSI meetings [5] (sampled at 16 kHz). Attention was paid to the definition of fair division of data into training/development/test parts with non-overlapping speakers. It was actually necessary to work on speaker turns rather than whole meetings, as they contain many overlapping speakers. We have balanced the ratio of native/nonnative speakers, balanced the ratio of European/Asiatic speakers and moved speakers with small portion of speech or keywords to the training set. The amounts of data in the training, development and test parts are 41.3 h, 18.7 h and 17.2 h respectively. The development part was used for system tuning (phoneme insertion penalty, etc.).

In the definition of keyword set, we have selected the most frequently occurring words (each of them has more than 95 occurrences in each of the sets) but checked, that the phonetic form of a keyword is not a subset of another word nor of word

System	Models	FOM
ICSI10h	tri-phones	**61.88**
CTS277h	tri-phones	**63.66**
TRAP-NN40h	mono-phones	**64.46**

Table 1: The results of different acoustic KWS systems.

transition. The percentage of such cases was evaluated for all candidates and words with high number of such cases were removed. The final list consists of 17 keywords:
`actually, different, doing, first, interesting, little, meeting, people, probably, problem, question, something, stuff, system, talking, those, using.`

Our experiments are evaluated using *Figure-of-Merit* (FOM) [6], which is the average of correct detections per 1, 2, ... 10 false alarms per hour. We can approximately interpret it as the accuracy of KWS provided that there are 5 false alarms per hour.

Obviously, in real scenarios, more specific words than `doing, probably`, etc. will be used. For statistical evaluation using FOM, we however need a set of keywords with many occurrences in the data. We are aware that this set of keywords favorizes LVCSR-KWS as these words are all represented in the recognition vocabulary and their tri-grams in the language model are reliably estimated.

4. Acoustic KWS

In acoustic KWS, the model of keyword is concatenated from phoneme models, we allow also for pronunciation variants (there are 33 variants for the total of 17 keywords). The filler and background models are phoneme loops. As these systems were developed for real-time operation, we have not used the right filler model. The likelihood of the keyword is taken from the last state of keyword model and immediately compared with the likelihood at the output of background model. Figures 3 and 4 show the networks used for acoustic KWS.

Both systems based on HMM/GMM make use of cross-word tri-phone models. Links among tri-phones are context sensitive (e.g. there is link between tri-phones `A-B+C` and `B-C+D` but not between `A-B+C` and `D-E+F`). The first and the last phoneme of keyword is expanded to all context possibilities in tri-phone network (Figure 4). The TRAP-NN system uses only context-independent models (Figure 3), so that the network is much simpler.

The results of acoustic keyword spotting are summarized in Table 1. The best FOM of 64.46% was obtained with the TRAP-NN system. This is a good result, as it confirmed our previous comparison of phoneme-recognition systems [3] showing the superiority TRAP-NN over HMM/GMM. The advantage of this system is also the simplicity of recognition network (mono-phones) and speed – the posteriors can be pre-computed so that the system's real-time factor was lower than 0.02 (more than 50× faster than real-time). On the other hand, when a new keyword is entered, this system must always go through all the data, which (even in case of 0.02×RT) can make the search times prohibitive. The main use of this system is in real-time spotting (meeting assistants, security) and as a post-processor for candidates selected by other techniques.

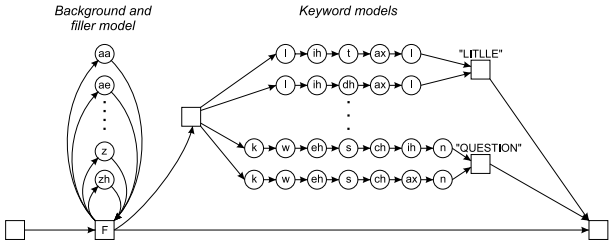

Figure 3: Keywords spotting network using mono-phones.

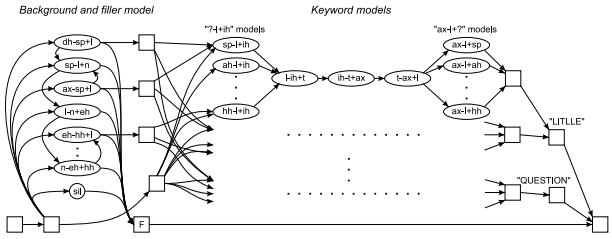

Figure 4: Keyword spotting network using tri-phones.

5. LVCSR KWS

Detecting keywords in the "hard" output of LVCSR gives only poor results (61.2%). Therefore we are using the LVCSR lattices – oriented acyclic graphs, where each node represents a word and each link represents time boundaries of the word at the end of the link (Figure 5). Searching in lattices provides better results, as the lattice holds several hypothesis in parallel.

LVCSR recognizer assigns each word acoustic $L_a^{lvcsr}(KW)$ and language model $L_l^{lvcsr}(KW)$ likelihoods. In case we "grep" for keywords in LVCSR word lattices and use the sum of these two likelihoods as confidence of the hypothesis, we obtain FOM of 61.23%. We are further improving this result by computing confidence of each hypothesis based on log-likelihood ratio (Eq. 1):

$$C^{lvcsr}(KW) = L_{alpha}^{lvcsr}(KW) + L^{lvcsr}(KW) + \\ + L_{beta}^{lvcsr}(KW) - L_{best}^{lvcsr}, \quad (2)$$

where the forward likelihood $L_{alpha}^{lvcsr}(KW)$ is the likelihood of the best path through lattice from the beginning of lattice to the keyword and backward likelihood $L_{beta}^{lvcsr}(KW)$ is computed from the end of lattice to the keyword. These two likelihoods are computed by the standard Viterbi formulas:

$$L_{alpha}^{lvcsr}(N) = L_a^{lvcsr}(N) + L_l^{lvcsr}(N) + \min_{N_P} L_{alpha}^{lvcsr}(N_P) \quad (3)$$

$$L_{beta}^{lvcsr}(N) = L_a^{lvcsr}(N) + L_l^{lvcsr}(N) + \min_{N_F} L_{beta}^{lvcsr}(N_F) \quad (4)$$

where N_F is set of nodes directly following node N (nodes N and N_F are connected by an arc), N_P is set of nodes directly preceding node N. The algorithm is initialized by setting $L_{alpha}^{lvcsr}(first) = 0$ and $L_{beta}^{lvcsr}(last) = 0$. The last likelihood we need in eq. 2: $L_{best}^{lvcsr} = L_{alpha}^{lvcsr}(last) = L_{beta}^{lvcsr}(first)$ is the likelihood of the most probable path through the lattice.

The result of LVCSR KWS with likelihood ratio confidence is 66.95%. We see that this result is better than acoustic KWS. In case we pre-index the LVCSR lattices, the search can be also very fast. The drawback of LVCSR is however the strong dependence on the dictionary – in case we want to search for a word not contained in the dictionary, we have no chance to find it. In the same time, these words can be the ones that contain very useful information (proper names, names of new products etc). LVCSR-based KWS should therefore be combined with a method for searching new and unknown keywords.

6. Phoneme Lattice KWS

This approach overcomes the drawbacks of LVCSR-KWS (dependency on dictionary) and acoustic-KWS (need to process all the data for a new keyword). Phoneme lattices were generated from phoneme posterior probabilities (output of TRAP-NN40h system). Phoneme insertion penalty was set to 0 for lattice generation to eliminate deletion of phonemes. We generated lattices with different branching factors. Generated lattices contain no language model probabilities, phoneme likelihood is only the acoustic likelihood $L_a^{phn}(N)$.

The search of keyword is different from the "grep" applied on LVCSR lattice. We are searching the phonetic form of the keyword, and our searching algorithm can handle insertions (keyword in lattice contains more phonemes than phonetic representation of searched keyword) and/or substitution (keyword in lattice contains different phonemes than phonetic representation of searched keyword) of phonemes. The search is implemented as a Viterbi algorithm, where the keyword confidences are given by:

$$C_{KW}^{phn} = L_{alpha}^{phn}(KW) + L^{phn}(KW) + L_{beta}^{phn}(KW) - L_{best}^{phn}, \quad (5)$$

where $L^{phn}(KW)$ is the likelihood of the keyword. For the sub-strings of K phonemes from the phoneme lattice, this likelihood is expressed as:

$$L^{phn}(KW) = \sum_{i=1}^{K} \left[\varphi(N) L_a^{phn}(N) + (1 - \varphi(N)) L_w(N) \right]. \quad (6)$$

The variable $\varphi(N) = 1$ if the i-th keyword phoneme matches $i-th$ phoneme from the sub-string of lattice. $\varphi(N) = 0$ in case the phoneme is not matching (insertion or substitution). $L_a^{phn}(N)$ is the acoustic likelihood of the i-th phoneme from lattice sub-string (stored in the lattice). In case there is no match, $L_w(N)$ can be understood as a penalization for substitution or insertion error. We have experimented with several ways to determine $L_w(N)$, the best results were obtained by setting $L_w(N)$ to the worst acoustic likelihood in the lattice adjusted to the length of phoneme N.

Partial Viterbi likelihoods $L_{alpha}^{phn}(KW)$ and $L_{beta}^{phn}(KW)$ from Eq. 5 are very similar to LVCSR-KWS. Forward likelihood $L_{alpha}^{phn}(KW)$ is the likelihood of the best path through lattice from the beginning of lattice to the keyword. Backward likelihood $L_{beta}^{phn}(KW)$ is computed from the end of lattice to the keyword. Forward and backward probability are recursively evaluated as:

$$L_{alpha}^{phn}(N) = L_a^{phn}(N) + \min_{N_P} L_{alpha}^{phn}(N_P) \quad (7)$$

$$L_{beta}^{phn}(N) = L_a^{phn}(N) + \min_{N_F} L_{beta}^{phn}(N_F) \quad (8)$$

where N_F is set of nodes directly following node N (nodes N and N_F are connected by arc), N_P is set of nodes directly preceding node N. The first node has $L_{alpha}^{phn}(first) = 1$ and the last node has $L_{beta}^{phn}(last) = 1$.

Finally, the likelihood $L_{best}^{phn} = L_{alpha}^{phn} = L_{beta}^{phn}$ is the likelihood of the most probable path through lattice. Note, that $L_{alpha}^{phn}(N)$, $L_{beta}^{phn}(N)$ and L_{best}^{phn} can be pre-computed and

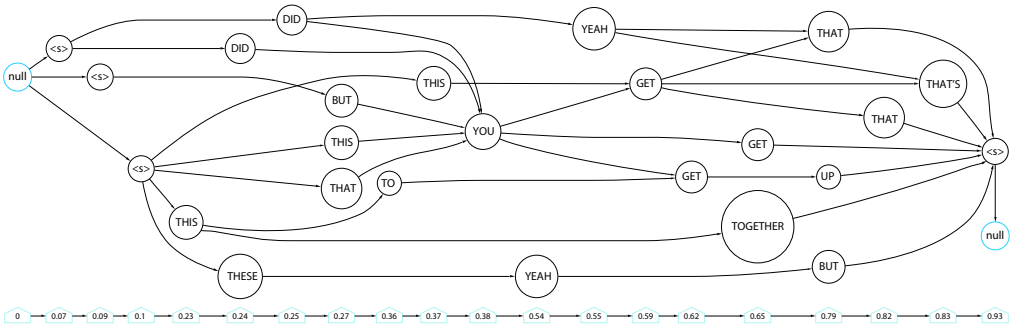

Figure 5: Example of word lattice for utterance "DID YOU GET THAT".

Parameters	Branching factor					
	2	3	4	5	6	7
NoIns NoSub	32.66	43.85	49.91	53.45	55.30	56.33
Ins NoSub	32.86	44.44	50.39	54.09	55.86	56.89
NoIns Sub	36.91	**51.25**	55.66	**57.45**	**58.32**	**58.90**
Ins Sub	**37.03**	51.08	**55.76**	57.20	58.22	58.84

Table 2: Accuracies (FOM) of systems depending on branching factor of lattice

stored with the lattice, so that the evaluation of confidence is very fast.

The results are summarized in Table 2. Searching for the exact phonetic form of keyword in lattice gives FOM of 56.33%. Allowing of substitutions increases the accuracy by $+2.5\%$ to 58.90 for branching factor 7. Generated lattices have sufficiently high density (caused by 0 word insertion penalty), so that allowing of insertions has no significant affect on the accuracy. Our highest branching factor was 7 because the FOM saturates there, only the size of lattices grow up. The best result on phoneme lattice KWS is FOM 58.90%.

The results (see the comparison of the 3 approaches in Table 3) are the worst among the three approaches, but we should take into account, that searching phoneme-lattices is able to combine the advantages of LVCSR and acoustic approaches (no dependence on the dictionary and possibility to search fast). We are aware the penalization used for substitutions was quite rudimentary and plan to use more sophisticated approaches (for example using phoneme confusion matrix to determine the penalization for a mismatch between the phoneme searched and contained in the phoneme lattice). This approach can also be used for a fast search of candidates that can be post-scored by the acoustic KWS.

7. Conclusions and future work

Comparison of accuracies of acoustic, LVCSR and phoneme-lattice KWS is shown in Table 3. All presented techniques are evaluated on informal continuous speech database containing native and non-native English speakers in meeting environment. The set of keywords contains 17 of the most frequent words for statistically reliable evaluation using Figure-of-merit (FOM).

The best accuracy is provided by system using searching in LVCSR word lattices and keyword confidence computation using likelihood ratio. The usefulness of LVCSR-KWS is however limited - the keyword must be contained in the LVCSR's vocabulary. This is not a problem for our keyword set (frequent words selected in order to have statistically reliable results) but can severely impair the performance when searching for really useful information (rare words, proper names, ...). This scenario is on the other hand well handled by the other two approaches. Taking into account the absence of language model, the results obtained by the acoustic KWS are very encouraging. The phoneme-lattice based KWS is not reaching the accuracies of the other two methods, but can be used for a fast pre-selection of candidates. Especially in case of searching archives containing hundreds or thousands hours of speech data, the speed of search is as important as the accuracy.

System	FOM
Acoustic KWS - TRAP-NN40h	**64.46**
Phoneme lattice KWS	**58.90**
LVCSR lattice KWS	**66.95**

Table 3: The results of different keyword spotting systems.

In future, we will work on improving the individual approaches, especially the phoneme-lattice spotter. We will also combine the three approaches in a modular, "Google-like" system, that will be usable for different scenarios (meetings, lectures, security applications).

8. References

[1] I. Szoke, P. Schwarz, L. Burget, M. Karafiát, and J. Černocký, "Phoneme based acoustics keyword spotting in informal continuous speech," in *Proc. RADIOELEKTRONIKA 2005*, Brno, Czech Republic, may 2005.

[2] T. Hain, J. Dines, G. Garau, M. Karafiat, D. Moore, V. Wan, R. Ordelman, and S. Renals, "Transcription of conference room meetings: an investigation," in *submitted to Eurospeech*.

[3] P. Schwarz, P. Matějka, and J. Černocký, "Towards lower error rates in phoneme recognition," in *Proc. TSD 2004*, Brno, Czech Republic, Sept. 2004, number ISBN 87-90834-09-7, pp. 465–472.

[4] I. Szoke, P Schwarz, P. Matějka, L. Burget, M. Karafiát, and J. Černocký, "Phoneme based acoustics keyword spotting in informal continuous speech," in *submitted to TSD 2005*, Karlovy Vary, Czech Republic, 2005.

[5] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI meeting corpus," in *International Conference on Acoustics, Speech, and Signal Processing, 2003. ICASSP-03*, Hong Kong, april 2003.

[6] J. R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden markov modeling for speaker-independent word spotting," in *International Conference on Acoustics, Speech, and Signal Processing, 1989. ICASSP-89*, Glasgow, UK, may 1989, vol. 1.