Efficient decoding and training procedures for utteranceverification in continuous speech recognition

Conference Paper · June 1996

DOI: 10.1109/ICASSP.1996.541144 · Source: IEEE Xplore

2 authors:

Eduardo Lleida
University of Zaragoza

Richard C. Rose
McGill University

EFFICIENT DECODING AND TRAINING PROCEDURES FOR UTTERANCE VERIFICATION IN CONTINUOUS SPEECH RECOGNITION

*E. Lleida *and R. C. Rose*

AT&T Bell Labs, Murray Hill, NJ, USA
lleida@mcps.unizar.es

ABSTRACT

It is often necessary in speech recognition to include a mechanism for verifying decoded utterances in order to account for incorrectly decoded vocabulary words and utterances corresponding to words or sounds that are not included in a prespecified lexicon [1, 2]. This paper describes an utterance verification procedure for hidden Markov model (HMM) based continuous speech recognition that is based on a Likelihood Ratio ($\mathcal{LR}$) criterion. There are two important contributions. The first is a search algorithm which directly optimizes a likelihood ratio criterion. This search algorithm is important because it allows decoding to be performed in speech recognition according to the same measure of confidence that is used in hypothesis testing. The second contribution is a corresponding training procedure for estimating model parameters which also directly optimizes the same likelihood ratio criterion. These techniques are applied to spontaneous spoken queries in the context of a "movie locator" dialog system [3].

1. INTRODUCTION

It is well known that speech recognition performance degrades rapidly when presented with utterances that either contain out-of-vocabulary words or are not well modeled by predefined language models. It is very difficult to anticipate all possible input utterances in the process of configuring an automatic speech recognition (ASR) system. As a result, it is often desirable to accept only the portions of a decoded utterance that have been decoded with sufficient "confidence". This implies the existence of an utterance verification (UV) procedure. The goal of UV is to verify whether a hypothesized word or string of words corresponds to actual occurrences of those words or to word substitutions or insertions, referred to collectively here as false alarms.

There are two ways to incorporate utterance verification into a speech recognizer. The first is as a "post-processor." Given the decoded lexical items and the associated segmentation provided by the speech recognizer, the UV system assigns a confidence measure to each segment. This is refered to here as a *two-pass* recognition/verification strategy. The two–pass procedure has the disadvantage that it must verify a hypothesized lexical item given the possibly errorful segmentation produced by the decoder. The second method for performing UV in speech recognition is to modify the decoding criterion used in the speech recognizer so that the decoded string is that which obtains the highest confidence score with respect to all possible string hypotheses. This is refered to here as a *one-pass* recognition/verification strategy. The principal advantage of the one–pass strategy is that the hypothesis test is not applied to a single string. In performing search by directly optimizing the confidence measure or hypothesis test criterion, an entire ensemble of hypotheses are simultaneously being evaluated.

In this paper, we present search and training procedures which directly optimize a confidence measure based on a likelihood ratio criterion. As a result, both training and recognition are performed under the same optimization criterion as is used for hypothesis testing. Section 2 presents the likelihood ratio based decoding strategy. While a very general search algorithm is presented, a more efficient algorithm taking the form of a modified Viterbi decoder is derived as a special case of this general procedure. Section 3 describes several empirically derived methods for integrating local confidence scores using non–uniform time weighting procedures. Section 4 summarizes the likelihood ratio based training algorithm which is used to adjust the HMM model parameters in a manner which is consistent with the UV procedure. Finally, in Section 5 these techniques are applied to spontaneous spoken utterances from a "movie locator" spoken dialog task [3].

2. LIKELIHOOD RATIO DECODER

This section presents a decoding algorithm which optimizes a likelihood ratio criterion. After introducing utterance verification in terms of a likelihood ratio based hypothesis testing framework, the decoding procedure presented. It is described as a method for obtaining a state sequence that maximizes the ratio of data likelihood with respect to a null hypothesis HMM model and alternate hypothesis HMM model respectively. It is assumed that the input to the speech recognizer is a sequence of feature vectors $Y = \{\vec{y}_1, ..., \vec{y}_T\}$ representing a speech utterance containing both within vocabulary (*target*) and out-of-vocabulary (*imposter*) words. It is also assumed that the output of the recognizer is a single *word* string hypothesis $\mathcal{W} = W_1, ..., W_K$ of length K. Of course, all the arguments given below apply to verifying one of multiple complete or partial string hypotheses produced as part of an N–best list or word lattice as well. As the model parameters for a particular unit are not known a priori, they have to be estimated from the training data assuming a known form of the density $P(Y|\lambda)$.

The likelihood ratio $\mathcal{LR}$ test is designed to determine whether or not a sequence of feature vectors Y were generated by a given family of probability densities, defining the following test:

$\mathcal{H}_0$: null hypothesis, Y generated by target model λ_c

$\mathcal{H}_1$: alternative hypothesis, Y generated by alternative

*This work has been carried out while visiting AT&T Bell Labs supported by a grant from the DGICYT (Dirección General de Investigación Científica y Técnica) of the Spanish government. Currently in Centro Politécnico Superior de Ingenieros, University of Zaragoza, Spain

0-7803-3192-3/96 $5.00©1996 IEEE

model λ_a

$$\mathcal{LR}(Y,\lambda^c,\lambda^a) \;=\; \frac{P(Y|\lambda^c)}{P(Y|\lambda^a)} \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\gtrless}} \tau \qquad (1)$$

where τ is a decision threshold. For utterance verification in ASR, λ^c and λ^a are HMM's corresponding to the *correct* or *target* hypothesis and the *alternative* hypothesis respectively.

As with any hypothesis testing procedure, an utterance verification procedure is evaluated with respect to two types of errors. *Type I errors* correspond to the correctly decoded vocabulary words being rejected by the utterance verification procedure. *Type II errors* or *false alarms* correspond to incorrectly decoded word insertions and substitutions being accepted by the UV process. These two errors are a function of the confidence measure threshold.

In practice, the alternative hypothesis model has two roles in utterance verification. The first is to reduce the effect of sources of variability on the confidence measure. If the probabilities of the null hypothesis model and the alternative hypothesis model are similarly affected by some systematic variation in the observations, then forming the likelihood ratio should cancel the effects of the variability. The second role of the alternate model is more specifically to represent the incorrectly decoded hypotheses that are frequently confused with a given lexical item. In the subword model based CSR system described in Section 5, each subword unit has associated with it a dedicated alternate hypothesis model.

In order to describe a likelihood ratio based decoder, one can simply expand the likelihood ratio in Equation 1 in terms of the underlying HMM parameters

$$\mathcal{LR}(Y,\lambda^c,\lambda^a) = \frac{\sum_{Q^c} \pi^c_{q^c_1} b^c_{q^c_1}(\vec{y}_1)\ldots a^c_{q^c_{T-1},q^c_T} b^c_{q^c_T}(\vec{y}_T)}{\sum_{Q^a} \pi^a_{q^a_1} b^a_{q^a_1}(\vec{y}_1)\ldots a^a_{q^a_{T-1},q^a_T} b^a_{q^a_T}(\vec{y}_T)}. \quad (2)$$

In Equation 2, $Q^c = \{q^c_1,\ldots,q^c_T\}$ and $Q^a = \{q^a_1,\ldots,q^a_T\}$ are the sequences of states for the correct and alternative model respectively.

Assuming that search is performed using the Viterbi algorithm, the complete data likelihood is replaced by $P(Y,\tilde{Q}|\lambda)$ where

$$\tilde{Q} = \arg\max_{Q} P(Y,Q|\lambda) \qquad (3)$$

is the single best state sequence. So the likelihood ratio is computed as

$$\mathcal{LR}(Y,\lambda^c,\lambda^a) = \frac{\pi^c_{\tilde{q}^c_1} b^c_{\tilde{q}^c_1}(\vec{y}_1)\ldots a^c_{\tilde{q}^c_{T-1},\tilde{q}^c_T} b^c_{\tilde{q}^c_T}(\vec{y}_T)}{\pi^a_{\tilde{q}^a_1} b^a_{\tilde{q}^a_1}(\vec{y}_1)\ldots a^a_{\tilde{q}^a_{T-1},\tilde{q}^a_T} b^a_{\tilde{q}^a_T}(\vec{y}_T)}. \qquad (4)$$

Let $q^2_t = (q^c_t, q^a_t)$ be a point in a 3 dimensional space defined by $(\lambda^c, \lambda^a, t)$. We define a path in the 3-D space as $Q_{3D} = \{q^2_1, q^2_2, \ldots, q^2_T\}$. The paths $\tilde{Q}^c$ and $\tilde{Q}^a$, defined by Equation 3, can be seen as the projection onto the two planes (λ^c, t) and (λ^a, t) of a 3-D path, as shown in Figure 1, which maximizes the likelihoods $P(Y,Q^c|\lambda^c)$ and $P(Y,Q^a|\lambda^a)$ separately over a speech segment. In the two-pass procedure, the speech segment is defined by the Viterbi backtrace on the plane (λ^c, t).

Define the best single 3-D path $\tilde{Q}_{3D}$ as the sequence of states which maximizes the likelihood ratio

$$\tilde{Q}_{3D} = \arg\max_{Q_{3D}} \mathcal{LR}(Y,\lambda^c,\lambda^a) \qquad (5)$$

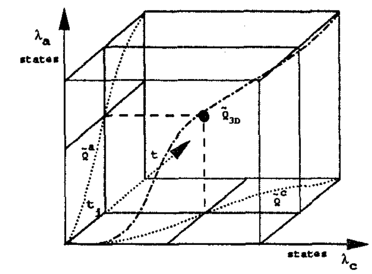

Figure 1: 3-D HMM search space

where $\tilde{Q}_{3D}$ can be found using a 3-D Viterbi search over the target hypothesis model states $1 \le n \le N_c$ and alternate hypothesis states $1 \le m \le N_a$,

$$\begin{aligned}\delta_t(n,m) &= \max_{\substack{1 \le i \le N_c \\ 1 \le j \le N_a}} [\delta_{t-1}(i,j)\frac{a^c_{in}}{a^a_{jm}}]\frac{b^c_n(\vec{y}_t)}{b^a_m(\vec{y}_t)} \quad t = 2,\ldots,T \\ \delta_1(n,m) &= \frac{\pi^c_n b^c_n(\vec{y}_1)}{\pi^a_m b^a_m(\vec{y}_1)}, \end{aligned} \qquad (6)$$

where N_c and N_a are the number of states in the target and alternative models respectively. This new criterion allows information concerning the alternative hypothesis to be introduced directly in the decoder. However, some constraints on the search space must be imposed to obtain more manageable computational complexity. One of many possible constrained search procedures is investigated here.

Suppose that λ^c and λ^a are HMM's with identical topologies, and that the state sequence is constrained so that $q^c_t = q^a_t = q_t$ for each $t = 1,\ldots,T$. In this way, the single optimum state sequence, $\tilde{Q}_{3D} = \tilde{Q}$, is decoded by applying those constraints to Equation 5. As a result, the process of identifying the optimum state sequence can take the form of a modified Viterbi algorithm where the recursion is defined as

$$\begin{aligned}\delta_t(j) &= \max_{1 \le i \le N} [\delta_{t-1}(i)\frac{a^c_{ij}}{a^a_{ij}}]\frac{b^c_j(\vec{y}_t)}{b^a_j(\vec{y}_t)} \quad t = 2,\ldots,T \quad 1 \le j \le N \\ \delta_1(j) &= \frac{\pi^c_i b^c_i(\vec{y}_1)}{\pi^a_i b^a_i(\vec{y}_1)} \quad 1 \le j \le N \end{aligned} \qquad (7)$$

The accumulated path score, $\delta_t(j)$, obtained in the Viterbi algorithm corresponds to a measure of confidence in the path terminating in state j at time t. This procedure implies that recognition and verification can be performed simultaneously as opposed to the two-pass procedure where strings are hypothesized and verified in separate steps. The algorithm of Equation 7 was used here for two purposes. The first was to obtain string hypotheses decoded according to a likelihood ratio criterion. The second purpose was to "rescore" utterance segmentations obtained from a maximum likelihood decoder for the purpose of providing improved confidence measures.

3. WORD LEVEL CONFIDENCE MEASURES

The output of the UV process is a set of word hypotheses with their corresponding confidence measure. There are many different ways that word level confidence measures can be formed by combining likelihood ratio scores. Five different word level measures of confidence for word W_k corresponding to phonetic baseform $U_k = u_{k,1},\ldots,u_{k,N_k}$, where $u_{k,j}$ is the j^{th} subword unit in U_k. Let $\mathcal{LR}_k$ and $\mathcal{LR}_{k,j}$ be the likelihood ratios for the word W_k and the unit $u_{k,j}$ respectively, computed using either the two-pass utterance verification strategy or the one-pass strategy given by Equation 7. The confidence measures are defined as:

$$m_1 = \log \mathcal{LR}_k \qquad m_2 = \frac{1}{N_k}\sum_{i=1}^{N_k} \log \mathcal{LR}_{k,i} \tag{8}$$

$$m_3 = \log\Big(\frac{1}{N_k}\sum_{i=1}^{N_k} (\mathcal{LR}_{k,i})^{-\kappa}\Big)^{1/\kappa} \tag{9}$$

$$m_4 = \frac{1}{N_k}\sum_{i=1}^{N_k} \frac{1}{1+\exp(-\gamma(\log \mathcal{LR}_{k,i} - \tau))} \tag{10}$$

$$m_5 = \frac{1}{\frac{1}{N_k}\sum_{i=1}^{N_k}(1+\exp(-\gamma(\log \mathcal{LR}_{k,i} - \tau)))} \tag{11}$$

where m_2 to m_5 are different non-uniform weightings of the likelihood ratios of the units that compose a given word. We will discuss with more detail the properties of this word confidence measures in the experimental results section.

4. LIKELIHOOD RATIO BASED TRAINING

The training procedure for the target and alternative models is based on the one-pass strategy. The goal of the training procedure is to increase the value of $\mathcal{LR}$ for correctly hypothesized keywords and decrease the value of $\mathcal{LR}$ for false alarms. This is accomplished by applying an iterative discriminative training algorithm.

Let $\{q_1, ..., q_t, ..., q_T\}$ be the sequence of states for the best sentence decoded with the one-pass procedure. Defining the *frame* log-likelihood ratio as

$$r_{ij}(\vec{y}_t) = \log(a^c_{ij} b^c_j(\vec{y}_t)) - \log(a^a_{ij} b^a_j(\vec{y}_t)), \tag{12}$$

the corresponding log-likelihood ratio for the unit u is defined as

$$R_u(Y^u) = \frac{1}{\mathrm{tf_u} - \mathrm{ti_u} + 1}\sum_{t=ti_u}^{t=tf_u} r_{q_{t-1}q_t}(\vec{y}_t) \tag{13}$$

where ti_u and tf_u are the initial and final frame of the speech segment decoded as unit u and $Y^u = \{\vec{y}_{ti_u}, ..., \vec{y}_{tf_u}\}$.

As it was defined earlier, utterance verification has to deal with two types of errors, *Type I error* or false rejection and *Type II error* or false alarm. The purpose of the utterance verification process is to minimize these types of errors, so, we have to define a function related with these errors as objective to be minimized. As the training algorithm is based on a gradient descent approach, the cost function must be continuous, at least, up to the first derivative.

Define the cost function $\mathcal{F}(Y^u, \lambda^c, \lambda^a)$ for a unit u as the sigmoid function [1]

$$\mathcal{F}(Y^u, \lambda^c, \lambda^a) = \frac{1}{1+\exp(-\gamma\delta(u)(R_u(Y^u) - \tau))} \tag{14}$$

$$\delta(u) = \begin{cases} -1 & u \in Correct \\ 1 & u \in Impostor \end{cases}$$

and the average cost function $\mathcal{C}(u, \lambda^c, \lambda^a)$ as

$$\mathcal{C}(u, \lambda^c, \lambda^a) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{F}(Y^u_i, \lambda^c, \lambda^a). \tag{15}$$

where N is the number of occurrences of the unit u in the training set. In this way, the average cost function is a *soft* count of the number of errors *Type I* and *Type II*, assuming that the decision threshold is τ in Equation 14. Effectively, imposters with scores greater than τ (*type II*) and targets with scores lower that τ (*type I*) increase the average cost function count, so, if we minimize this function we can reduce the misclassification between targets and imposters. Note that the cost function is a smooth function with gradient concentrated around τ. So the cost function in the gradient descent algorithm is most heavily influenced by segments with scores close to τ. This means that the imposters with score values which are much greater than and targets with score values that are much less than τ are considered as outliers, and have little effect in updating the model parameters.

Finally, the model parameters are adjusted according to the following formula

$$\Lambda_{n+1} = \Lambda_n - \epsilon \nabla \mathcal{C}(u, \lambda^c, \lambda^a), \tag{16}$$

where n is the n-th iteration of training and ϵ is a learning rate constant.

5. EXPERIMENTAL STUDY

The utterance verification techniques described above were evaluated on spontaneous spoken utterances collected during a trial of a dialog based task over the public switched telephone network. In this task, users generated queries concerning the location and times that movies that were playing in their area [3]. Utterance verification techniques were evaluated in terms of their ability to detect correctly decoded vocabulary words and reject false alarms.

Recognition was performed using a finite state grammar with a lexicon of 570 words. A total of 3025 sentences were used for training acoustic models and 752 utterances were used for testing. The total number of words in the test set was 4864, where 134 of them were out-of-vocabulary. There were 275 sentences in the test set that could not be parsed by the finite state grammar, where 85 of these sentences contained out-of-vocabulary words. The feature set used in recognition included 12 Mel-cepstrum, 12 delta Mel-cepstrum, 12 delta-delta Mel-cepstrum, energy, delta energy and delta-delta energy coefficients. The subword models used in the recognizer consisted of 43 context independent units. The target models, λ^c, were three state continuous density HMM's with a maximum of 32 Gaussian mixtures per state.

Each alternative model was composed of two models, *background* λ^a_{bg}, and *imposter* λ^a_{im}. Hence, the likelihood ratio of Equation 1 becomes

$$\mathcal{LR}(Y, \lambda^c, \lambda^a) = \frac{P(Y \mid \lambda^c)}{\alpha_{bg}P(Y \mid \lambda^a_{bg}) + \alpha_{im}P(Y \mid \lambda^a_{im})}, \tag{17}$$

where α_{bg} and $\alpha_{im} = 1 - \alpha_{bg}$ in Equation 17 can also be estimated in training. The purpose of λ^a_{bg} is to provide a broad representation of the feature space. This broad representation serves to reduce the dynamic range of the likelihood ratio. A single background alternate hypothesis model was shared amongst all "target" HMM models. The purpose of λ^a_{im} is to provide a detailed model of the decision regions associated with each individual HMM. Only the target and imposter models are updated using the discriminative training procedure. Unless it is stated otherwise, the alternate hypothesis models used in the experiments described in this section were parameterized as follows. The single background model, λ_{bg}, consisted of 3 states and 32 mixtures per state. For each subword HMM, an imposter model, λ^a_{im}, consisting of 3 states and 8 mixtures per states was trained from the false alarms decoded during training for the corresponding unit.

For all of the UV procedures that were investigated, it was found that performance was heavily dependent on the manner in which word level confidence measures were computed from unit level likelihood ratios as described in Section 3. A comparison of the five different confidence measures defined in Equations 8 through 11 is given in Figure 2. Figure 2a and 2b display the sum of *type I* and *type II* errors plotted against the threshold that is applied to the word level confidence scores. It is clear from Figure 2a that confidence measures m_2 and m_3, obtained by combining unit level $\mathcal{LR}$ scores, out–perform the word level confidence score, m_1.

Figure 2b illustrates the advantages associated with confidence measures m_4 and m_5. First, the error rates at the optimum threshold settings are very similar, and these error rates are the lowest obtained among the five measures that were evaluated. Second, the dynamic range for the two measures is constrained to the interval between zero and one. Finally, for m_5 in particular, the measure is robust with respect variation in the threshold value. The error rate does not change significantly over a broad range of threshold values. A comparison between histograms of the m_4 confidence scores given in Figure 2c and the m_5 confidence scores in Figure 2d provides insight into this roubustness. While the scores for correctly decoded hypotheses have similar distributions for the two measures, the false alarm scores are more uniformly distributed for measure m_5 than m_4. However, since the form of m_4 is more closely related to the cost function of Equation 14 which is used in the likelihood ratio based training procedure, m_4 will be used in the remaining experiments.

An additional set of experiments was performed to demonstrate the performance of the different decoding procedures. The effect of the likelihood ratio based training procedure on performance was also investigated. There are three different scenarios that are compared. The first scenario is the "one-pass" (OP) procedure where the likelihood ratio decoding algorithm of Equation 7 is used both for generating the recognition hypotheses and for providing the likelihood ratio scores. The second decoding scenario, TPFR, is a "two-pass" procedure where a maximum likelihood decoder is used during search to generate the recognition hypotheses and the $\mathcal{LR}$ decoding algorithm of Equation 7 is used to obtain confidence scores only for that portion of the utterance corresponding a hypothesized word. The third scenario, TPMR, is also a two pass strategy where the confidence measure is obtained by computing the likelihoods $P(Y, Q^c|\lambda^c)$ and $P(Y, Q^a|\lambda^a)$ separately. Figures 3a,b show a comparison between the one-pass (OP) and the two-pass procedures. The curves in Figure 3a represent receiver operating characteristic curves which display the percent of words that were correctly recognized with respect the false alarm rate. The target models and the alternative models were the same in both procedures. Clearly, the one-pass procedure gives better performance than the two-pass procedure over the entire range of confidence threshold settings. Also, the use of the likelihood ratio decoder for "re-scoring" word hypotheses in the two-pass procedure (TPFR) outperforms the TPMR.

Figures 3c,d provide a comparison of two different parameterizations of λ^a. In the first case, $\lambda^a = \lambda^a_{bg}$. In the second case, $P(Y|\lambda^a) = 0.5P(Y|\lambda^a_{bg}) + 0.5P(Y|\lambda^a_{im})$, where an individual λ^a_{im} is assigned to each subword unit. It is clear from the figures that using dedicated imposter models improves the performance of the system. After one iteration of the training procedure, an improvement is obtained in both cases. Finally, Table 1 shows the error rates for the minimum combined error operating point using different alternative model parameterizations ($\alpha_{bg} = \alpha_{im} = 0.5$). It is interesting to note from the table that successive iterations of the training procedure serve to compensate for poor initialization of the alternate hypothesis models.

6. SUMMARY

Decoding and training procedures for HMM utterance verification and recognition have been presented which are based on the optimization of a likelihood ratio based criterion. The decoding algorithm presented allows word hypothesis detection and verification to be performed simultaneously in a "one-pass" procedure. Experimental results have shown that the one-pass recognition and verification procedure consistently obtains better performance than than more traditional "two-pass" procedures.

Acknowledgements

This work would not have been possible without the considerable work of C.D. Mitchell on the basic infrastructure of the speech recognition system used in this work.

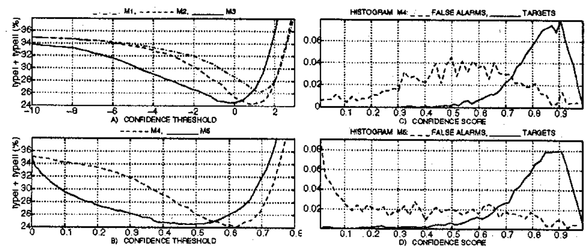

Figure 2: Performance of word level confidence measures

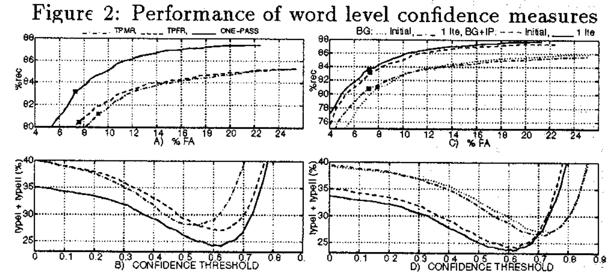

Figure 3: a,b) Comparison of UV scenarios; c,d) Comparison of alternative model definitions

mixtures	λ^a_{bg} : 32, λ^a_{im} :4		λ^a_{bg} : 32, λ^a_{im} :8		λ^a_{bg} : 16, λ^a_{im} :4	
	OP	TPFR	OP	TPFR	OP	TPFR
initial	29.3	28.1	31.5	28.3	43.3	30.8
1 ite	25.9	27.0	26.9	27.4	26.3	27.2
2 ite	25.7	26.8	24.5	27.2	25.6	27.1

Table 1: Type I + type II minimum error rates (%)

7. REFERENCES

[1] R.C. Rose, B.H. Juang, C.H. Lee, "Verifying String Hypothesis in Continuous Speech Recognition", Proc. ICASSP-95.

[2] R.A. Sukkar, C.H. Lee, B.H. Juang,, "A Vocabulary Independent Discriminatively Trained Method for Rejection of Non-Keywords in Subword Based Speech Recognition", EUROSPEECH-95, 1995.

[3] J.J. Wisowaty, "Continuous Speech Interface for a Movie Locator Service", Proc. of the Human Factors and Ergonomics Society, 1995.