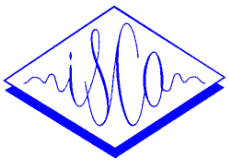

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

Minimum Error Classification Training for HMM-Based Keyword Spotting

Yasuhiro KOMORI and David RAINTON*

ATR Interpreting Telephony Laboratories, Kyoto, 619-02 JAPAN
(komori@cis.canon.co.jp)

Abstract

This paper compares and contrasts the keyword spotting performance of conventional maximum likelihood trained HMMs vs. that of minimum error trained HMMs.

The unique aspect of this work is the use of a new minimum error classification algorithm [1] for training the continuous mixture density HMM components of an HMM-based keyword spotting system. The actual spotting algorithm used was the HMM garbage model approach proposed previously in [2] [3].

Speaker-independent keyword spotting experiments were performed using the ATR Japanese continuous speech database. The reported results show the clear superiority of the minimum error trained HMMs in the chosen keyword spotting application.

10.21437/ICSLP.1992-5

1. Introduction

In speech recognition research today, interest is shifting more and more towards the recognition of real life, spontaneous utterances, produced in as natural an environment as possible. As a consequence, such speech inevitably contains frequent ungrammatical, meaningless and mistaken utterances, which conventional, grammatically constrained recognizers are often unable to cope with. One increasingly popular solution to this problem of ungrammaticallity, is the use of keyword spotting.

In this paper we employ a recently proposed, continuous mixture density HMM based keyword spotting technique [2] [3], consisting of a parallel network of HMM phone based keywords, plus an additional garbage (GB) HMM for modelling non-keyword speech, and a silence HMM. However, unlike conventional approaches, where the HMM network is typically trained using maximum likelihood (ML) methods, here we employ a new minimum error (ME) training algorithm [1]. That is, instead of training the HMMs to maximize the probability of producing the training data, as in the conventional ML-training approach, all the HMMs, including the GB-HMM, are trained to directly minimize the number of keyword spotting errors. Although several instances of keyword spotting using discriminative training have already been described in the literature [4] [5], this paper details the first application of the newly proposed [7] ME-training algorithm in an HMM-based keyword spotting framework.

The keyword spotting performance of conventional ML-trained HMMs vs. that of ME-trained HMMs, was experimentally compared using the ATR, speaker-independent, Japanese continuous speech database.

*Currently with Information Systems Research Center, Canon Inc., Kawasaki, 211 JAPAN

2. Hmm-Based Keyword Spotting

Figure 1 illustrates the keyword spotting method used in this paper. The algorithm employs a very simple HMM-based keyword spotting technique, previously described in [2] [3]. Briefly, the algorithm comprises of a null grammar, time synchronous, Viterbi search decoder, with a parallel network of phone-HMM sequences, representing keyword speech, a silence HMM, and a single GB-HMM representing non-keyword background speech. Any part of an utterance, if not part of a keyword, or silence, is classified as background speech, or "garbage", and is modelled using the GB-HMM.

Crucial to this approach then, is the use of the GB-HMM, which is trained on every phone realization appearing in the training data, irrespective of whether or not it occurs within a keyword. The phone HMMs, on the other hand, are trained simply using those phone realizations occurring within keywords, a simple lexicon being used to map between keywords and phone-HMM sequences.

During recognition, phone models compete with the GB model, a keyword only being spotted when, for a given section of speech, the associated phone-HMM sequence probability is greater than that of the most likely GB-HMM sequence.

3. Minimum Error Training

In ME-training, instead of training the HMMs to maximize the probability of producing the training data, as in the conventional ML approach, we directly train the HMMs to minimize the number of keyword spotting errors. This is achieved by gradient descent minimization of a "loss" function, the minimization of which is directly related to the minimization of the keyword spotting error rate [6] [7] [8] [9] [10].

The classifier structure itself is defined by a set of "HMM classes", *i.e.*

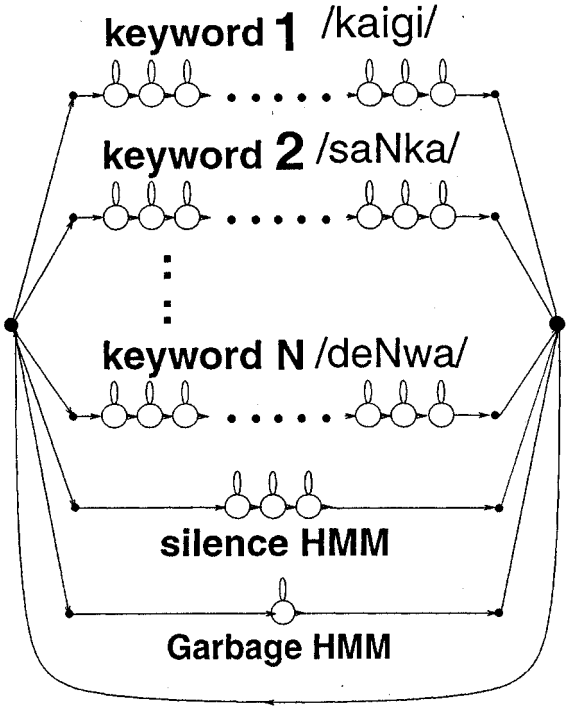

Fig. 1. HMM-Based Keyword Spotting.

$$\lambda = \{\lambda_1, \ldots, \lambda_i, \ldots, \lambda_I\} \tag{1}$$

where I is the number of HMMs, and each λ_i is a vector of HMM parameters completely characterizing the i'th HMM model. Each "HMM class" represents either a phone, or the garbage model. The training data set O, is defined as being comprised of K different "token classes", *i.e.*

$$O = \{O_1, \ldots, O_k, \ldots, O_K\} \tag{2}$$

where O_k is the sequence of training tokens from the k'th token class. In the chosen application, each token class corresponds to either a keyword, or the garbage model. If there are N_k tokens per token class then the k'th token class is represented by the set

$$O_k = \{O_{k,1}, \ldots, O_{k,n}, \ldots, O_{k,N_k}\}. \tag{3}$$

$O_{k,n}$ is the sequence of observation vectors produced by the n'th token in the k'th class.

The mapping between HMM sequences and the token classes is defined by a grammar $\mathcal{G}(\cdot)$, *i.e.*

$$\mathcal{G}(O_k) \mapsto \Lambda_k \tag{4}$$

where Λ_k is the "composite HMM" corresponding to the k'th token class. Each composite HMM Λ_k is a network of individual HMMs from λ representing all possible parses for the k'th token class. In this paper, each composite HMM is a keyword HMM produced from a network of phone HMMs.

The loss itself is defined as [7]

$$L(O, \lambda) = \sum_{k=1}^{K} \left\{ \sum_{n=1}^{N_k} l(d(O_{k,n}, \lambda)) \right\} \tag{5}$$

where $l(\cdot)$ is the individual token loss for the n'th token from the k'th token class. This token loss is in turn defined in terms of the misclassification measure $d(O_{k,n}, \lambda)$, *i.e.*

$$l(d(O_{k,n}, \lambda)) = \frac{1}{1 + \exp\left(-\alpha\left[d(O_{k,n}, \lambda) - \beta\right]\right)} \tag{6}$$

The real valued constants α and β control the slope and mid point of the sigmoid, and are determined experimentally. The misclassification measure $d(O_{k,n}, \lambda)$ is a measure of the penalty incurred when correctly classifying the n'th token in the k'th token class, *i.e.*

$$\begin{aligned} d(O_{k,n}, \lambda) &= -g(O_{k,n}, \Lambda_k) + g(O_{k,n}, \Lambda_c) \\ \text{where} \quad c &= \operatorname*{argmax}_p \; g(O_{k,n}, \Lambda_p) \end{aligned} \tag{7}$$

where $g(O_{k,n}, \Lambda_p)$ is the discriminant function

$$g(O_{k,n}, \Lambda_p) = \ln\left(P(O_{k,n}, \Theta_p | \Lambda_p)\right) \tag{8}$$

and Θ_p is defined as the Viterbi model/state sequence for the p'th composite HMM.

Importantly this definition of loss is first order differentiable with respect to all the HMM parameters, *e.g.* mixture weights, means and variances. Thus the loss can be minimized using standard gradient descent techniques. For the results presented in this paper, the very simple descent algorithm described in [1] was used. Prior to gradient descent the HMMs were initialized using conventional ML-training. In these experiments, we only updated mixture weights and means.

4. Experiments

The aim of this work was to compare and contrast the keyword spotting performance obtained using conventional ML-trained HMMs, with that obtained from HMMs trained using the above ME algorithm.

Speaker-independent keyword spotting performance was evaluated using the ATR Japanese speech database, containing 64 keyword realizations. Both the ME and ML models were trained using speech data obtained from 8 male speakers, testing being conducted on an open 9'th male speaker.

4.1 Acoustic Analysis

All speech data was sampled and digitized at 12kHz. The resulting speech samples were then pre-emphasized, using the filter transfer function $H(z) = 1 - 0.97z^{-1}$, Hamming windowed, and transformed to 12 LPC cepstral coefficients, 12 delta cepstral coefficients (computed over a 60ms window), an energy and a delta energy coefficient. A frame-shift window was used to produce a 26 dimensional observation vector every 5ms

4.2 Hmms

The phone-HMMs, chosen as the basic units for keyword representation, were all 3 state, 3 Gaussian mixture models, each state being connected to just itself and the next state. There were, in total, 47 phones and a silence model with separate modelling of short and long vowels, and in the case of some sounds, different HMMs representing different utterance positions.

For modelling non-keyword back-ground speech a single GB-HMM was used, consisting of a single state, with 64 Gaussian mixtures and a self loop.

No explicit duration control was used in any of the HMMs. Furthermore, no keyword HMM sequence duration restrictions were imposed, and no spotting probability thresholds were employed.

4.3 Training and Testing

Initially, all models (ie. silence-, phone- and GB-HMMs) were trained using standard forward-backward ML re-estimation. The single silence-HMM and the 47 phone-HMMs were trained using the silence/phone hand-labels. The GB-HMM was also trained on the same speech, simply by replacing all phone hand-labels with a single GB label. The training data comprised of 8 male speakers, 1,400 words per speaker, from the 5,240 word ATR speech database.

After initial ML-training, the silence-, phone- and GB-HMM were then re-trained using the ME algorithm. This retraining was performed using continuous speech data, from the same 8 speakers. Testing was performed on an open 9'th male speaker's continuous speech. Each speaker's speech data contained the same 64 keyword realizations, in the same spoken context.

The 64 keyword realizations, appearing in the training and testing date, consisted of the following eight keywords: 1) *kaigi*, 2) *saNka*, 3) *hoNyaku*, 4) *kokusai*, 5) *moushikomi*, 6) *touroku*, 7) *tsuuyaku*, 8) *zdeNwa*.

Table I shows both training and testing conditions.

4.4 Results

The experimental keyword spotting results are shown in Tables II, III and IV. Table II shows the results for the ME-trained phone-HMMs (the GB-HMM was ML-trained). Table III shows the results for the ME-trained GB-HMM (the phone-HMMs were ML-trained). Table IV shows the results for ME-trained phone-HMMs and a ME-trained GB-HMM. The **iter.** indicates the number of iterations of ME-training. The **spots** indicates the number and rate of correctly spotted keywords. The **del.** indicates the number of deletion errors. The **ins.** indicates the number of false alarms. The **f/w/h** indicates the false alarm rate per keyword per hour. The **init.**, in first row in every table, shows the baseline keyword spotting performance obtained using ML-trained phone-HMMs and a ML-trained GB-HMM.

TABLE I Training and Testing Conditions

Condition of ML-training (1,400 words of 8 males)

silence & 47 phones	3-states, 3-mixs	hand-label
garbage	1-state, 64-mixs	replace label

Condition of ME-training (64 keywords of 8 males)

silence & 47 phones	3-states, 3-mixs	context close
garbage	1-state, 64-mixs	25 sentences

Testing by ME-trained phone-HMMs

silence & 47 phones	3-states, 3-mixs	ME-trained
garbage	1-state, 64-mixs	ML-trained

Testing by ME-trained GB-HMM

silence & 47 phones	3-states, 3-mixs	ML-trained
garbage	1-state, 64-mixs	ME-trained

Testing by ME-trained all HMMs

silence & 47 phones	3-states, 3-mixs	ME-trained
garbage	1-state, 64-mixs	ME-trained

4.5 Discussion

When we consider only the number of spotted keywords, the best results were obtained using the ML-trained HMMs, which spotted 59 out of 64 keywords. However, in this instance, the number of false alarms was 195, which is rather a large number.

Considering the balance between spotted keywords and false alarms, the best result is the case of ME-trained phone-HMMs at iter.=25 in Table II. The number of spotted keywords is 57 and the false alarms only 76. Thus, while the number of spotted keywords is slightly reduced, the number of the false alarms has been more than halved.

The trends seen in ME-training, differ depending on the experiment. In the ME-trained phone-HMMs, Table II, over-training is evident. Since, while the number of false alarms always decreases as the number of ME iterations increases, the number of spotted words, increases after the first iteration, until peak performance is reached, and then begins to decrease. In the ME-trained GB-HMM, Table III, performance seems to peak at around 50-55 spotted keywords and 120-140 false alarms. The worse case was when all HMMs were ME-trained, Table IV. Here ME-training drastically decreases the number of false alarms only at the expense of also drastically reducing the number of spotted keywords. This may possibly be due to the fact that the amount of available ME-training data for the phone-HMMs is much less than that for the GB model. The ME-training algorithm may be unable to sensibly cope with such an imbalance.

From these experiments, we conclude that two problems have to be solved before ME-training can be successfully applied to HMM-based keyword spotting:

1) a ME-training stopping iteration must be found,

2) a sensible balance must be found between keyword and non-keyword, training data.

5. Conclusion

In this paper, a new minimum error classification training method is applied to an HMM-based keyword spotting algorithm. Speaker-independent keyword spotting experiments were performed using continuous speech data, the results of which demonstrate the effectiveness of minimum error classification training compared to conventional maximum likelihood training.

Acknowledgements

I would like to thank Dr. Akira Kurematsu, Shigeki Sagayama, Dr. Masahide Sugiyama for their continuous support, and Richard Lengagne for training the maximum likelihood HMMs and also members in ATR for their fruitful discussions.

References

[1] D. Rainton and S. Sagayama, *Minimum Error Classification Training of HMMs -Implemetation Details and Experimental Results*, IEICE Tech. Report, SP91-107, pp.39-46, Jan. 1992.

[2] J.R. Rohlicek, W. Russell, S. Roukos and H. Gish, *Continuous Hidden Markov Modeling for Speaker-Independent Word Spotting*, Proc. of ICASSP'89, pp.627-630. May 1989.

[3] R.C. Rose and D.B. Paul, *A Hidden Markov Model Based Keyword Recognition System*, Proc. of ICASSP'90, pp.129-132. April 1990.

[4] R.C. Rose, *Discriminant Wordspotting Techniques for Rejecting Non-vocabulary Utterances in Unconstrained Speech*, Proc. of ICASSP'92, Vol. 2 pp.105-108. March 1992.

[5] L.T. Niles, L.D. Wilcox and M.A. Bush, *Error-Correcting Training for Phoneme Spotting*, Proc. of ICASSP'92, Vol. 1 pp.425-428. March 1992.

[6] S. Katagiri, C.H. Lee and B.H. Juang, *A generalized probabilistic descent method*, Proc. Acoust. Soc. of Japan, 2-p-6, pp. 141-142, Nagoya, Japan, Sept. 1990

[7] S. Katagiri, C.H. Lee and B.H. Juang, *New discriminative algorithms based on the generalized probabilistic descent method*, Proc. IEEE-SP Workshop on Neural Networks for Signal Processing, Princeton, Sept., 1991.

[8] S. Katagiri, C.H. Lee and B.H. Juang, *Discriminative MultiLayer Feed-Forward Networks*, Proc. IEEE-SP Workshop on Neural Networks for Signal Processing, Princeton, Sept., 1991.

[9] P.C. Chang, S.H. Chen and B.H. Juang, *Discriminative Analysis of Distortion Sequences in Speech Recognition*, Proc. ICASSP91, pp. 549-552.

[10] W. Chou, B.H. Juang and C.H. Lee, *Segmental GPD Training of HMM Based Speech Recognizer* Proc. ICASSP92, pp. 473-476, March 1992.

TABLE II ME-trained phone HMMs

iter.	spots (%)	del.	ins.	f/w/h
init.	59 (92.2)	5	195	62.4
05	52 (81.3)	12	177	56.7
10	53 (82.8)	11	153	49.0
15	54 (84.4)	10	123	39.4
20	56 (87.5)	8	89	28.5
25	**57 (89.1)**	**7**	**76**	**24.3**
30	51 (79.7)	13	55	17.6
35	46 (71.9)	18	37	11.9
40	45 (70.3)	19	37	11.9
45	32 (50.0)	32	14	4.5

TABLE III ME-trained GB-HMM

iter.	spots (%)	del.	ins.	f/w/h
init.	59 (92.2)	5	195	62.4
05	49 (76.6)	15	142	45.5
10	49 (76.6)	15	137	43.9
15	50 (78.1)	14	129	41.3
20	49 (76.6)	15	122	39.1
25	55 (85.9)	9	124	39.7
30	53 (82.8)	11	126	40.3
35	54 (84.4)	10	135	43.2
40	54 (84.4)	10	143	45.8
45	53 (82.8)	11	144	46.1

TABLE IV ME-trained all HMM

iter.	spots (%)	del.	ins.	f/w/h
init.	59 (92.2)	5	195	62.4
05	40 (62.5)	24	125	40.0
10	43 (67.2)	21	103	33.0
15	44 (68.8)	20	64	20.5
20	36 (56.3)	28	39	12.5
25	44 (68.8)	20	34	10.9
30	41 (64.1)	23	25	8.0
35	33 (51.6)	31	21	6.7
40	34 (53.1)	30	23	7.4
45	22 (34.4)	42	11	3.5