DISCRIMINATIVE TRAINING FOR SPEAKER IDENTIFICATION BASED ON MAXIMUM MODEL DISTANCE ALGORITHM

Q.Y. Hong and S. Kwong

Dept of Computer Science, City University of Hong Kong, Hong Kong, China
qyhong@cs.cityu.edu.hk, cssamk@cityu.edu.hk

ABSTRACT

In this paper we apply the Maximum model distance (MMD) training [4] to speaker identification and a new selection strategy of competitive speakers is proposed to it. The traditional ML method only utilizes the utterances for each speaker model, which probably leads to a local optimization solution. By maximizing the dissimilarities among those similar speaker models, MMD could add the discriminative capability into the training procedure and then improve the identification performance. Based on the TIMIT corpus, we designed the word and sentence experiments to evaluate this proposed training approach. The results show that the identification performance can be improved greatly when the training data is limited.

1. INTRODUCTION

Speaker identification is the process of determining from which of the registered speakers a given utterance comes [1]. It is very popular to model the speakers with the Gaussian mixture model (GMM) [2]. It can be viewed as a single-state hidden Markov model (HMM). Generally, the maximum-likelihood (ML) estimation is considered as a good choice of training approach. The standard ML design criterion is to use a training sequence of observations $\mathbf{O}$ to derive the set of model parameters λ, yielding

$$\lambda_{ML} = \arg\max_{\lambda} P(\mathbf{O} \mid \lambda) \tag{1}$$

However this method only considers the likelihood for a single speaker. That is, each model is estimated separately using its labeled training utterances. When there are confusable models or the training data is limited, usually it can only reach a local optimization solution. To compare the likelihood against those similar utterances and maximizes their likelihood differences, another training algorithm named maximum model distance (MMD) [3,4] was developed for speech recognition. In this algorithm, each HMM represents the stochastic characteristics of a class of acoustic signals, and the difference of those stochastic characteristics can be mapped into the dissimilarities of their HMMs. By maximizing the dissimilarities among HMMs, the performance of speech recognizer would be improved.

In this work, this discriminative algorithm is extended to the GMM-based speaker identification. The general theory of MMD is introduced first and the re-estimation formulas of GMM are given. After that, we describe a different selection method of the competitive speaker. Experiments based on TIMIT are then conducted to verify the performance of this proposed method.

2. MAXIMUM MODEL DISTANCE

In the GMM-based system, the acoustic distribution of feature vectors extracted from a speaker λ_v 's speech $\mathbf{O}^v = \{\mathbf{o}_t, 1 \le t \le T_v\}$ is modeled by a Gaussian mixture density, which is defined as

$$P(\mathbf{O}^v \mid \lambda_v) = \prod_{t=1}^{T_v} \sum_{k=1}^{K} c_k N_v(\mathbf{o}_t, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{2}$$

The mixture density is a weighted linear combination of K component Gaussian densities $N_v(\mathbf{o}_t, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, with mean vectors $\boldsymbol{\mu}_k$ and covariance matrices $\boldsymbol{\Sigma}_k$ respectively,

$$N_v(\mathbf{o}_t, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{D/2}\left|\boldsymbol{\Sigma}_k\right|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{o}_t - \boldsymbol{\mu}_k)' \boldsymbol{\Sigma}_k^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_k)\right] \tag{3}$$

where prime denotes vector transpose and D is the dimension of the vector $\mathbf{o}_t$.

Using the log probability, a model distance measure $D(\lambda_v, \Lambda)$ between model λ_v and the whole model set Λ with M speakers is defined as

$$D(\lambda_v, \Lambda) = \frac{1}{T_v}\left\{\log P(\mathbf{O}^v \mid \lambda_v) - \log\left[\frac{1}{M-1}\sum_{\theta=1,\theta\neq v}^{M}\left(P(\mathbf{O}^v \mid \lambda_\theta)\right)^{\eta}\right]^{\frac{1}{\eta}}\right\} \tag{4}$$

where η is a positive number. When η approach ∞, the term in the bracket becomes $\max_{\theta=1,\theta\neq v} P(\mathbf{O}^v \mid \lambda_\theta)$, i.e. only the top competitor is considered.

The maximum model distance criterion is to find the entire model set Λ such that the model distance is maximized:

$$(\Lambda)_{MMD} = \arg\max_{\Lambda} \sum_{v=1}^{M} D(\lambda_v, \Lambda) \tag{5}$$

When searching the classifier parameter Λ, one could realize different weight distributions among the competitors of λ_v by varying the value of η:

$$D(\Lambda) = \sum_{v=1}^{M} D(\lambda_v, \Lambda) = \sum_{v=1}^{M} \frac{1}{T_v} \left\{ \log P(\mathbf{O}^v \mid \lambda_v) - \log \left[\frac{1}{M-1} \sum_{\theta=1,\theta\neq v}^{M} \left(P(O^v \mid \lambda_\theta) \right)^\eta \right]^{\frac{1}{\eta}} \right\}$$
$$= \sum_{v=1}^{M} \frac{1}{T_v} \log P(\mathbf{O}^v \mid \lambda_v) - \frac{1}{\eta} \sum_{v=1}^{M} \frac{1}{T_v} \times \log \left[\sum_{\theta=1,\theta\neq v}^{M} \left(P(\mathbf{O}^v \mid \lambda_\theta) \right)^\eta \right] - \frac{1}{\eta} \sum_{v=1}^{M} \frac{1}{T_v} \log \frac{1}{M-1} \tag{6}$$

Since $D(\Lambda)$ is a smooth and differentiable function in terms of the model parameter set Λ. Traditional optimization procedures like the gradient scheme could be used to find the optimal solution of Equation 6. The parameter adjustment rule is

$$\widetilde{\Lambda}_{n+1} = \Lambda_n + \varepsilon_n U_n \nabla D(\Lambda) \mid_{\Lambda=\Lambda_n} \tag{7}$$

where $\widetilde{\Lambda}$ is used to distinguish from Λ, which satisfies the stochastic constraints on the GMM model parameters, i.e. $\sum_{i=1}^{K} c_i = 1$. ε_n is a small positive number that satisfies certain stochastic convergence constraints. U_n could be an identity matrix or a properly designed positive-definite matrix, $\nabla D(\Lambda)$ is the gradient vector of the target function with respect to the parameter set Λ. From Equation 6, we get

$$\frac{\partial D(\Lambda)}{\partial \lambda_v} = \frac{1}{T_v P(\mathbf{O}^v \mid \lambda_v)} \frac{\partial P(\mathbf{O}^v \mid \lambda_v)}{\partial \lambda_v} - \sum_{\theta=1,\theta\neq v}^{M} \frac{\Pi_\theta^v}{T_\theta P(\mathbf{O}^\theta \mid \lambda_v)} \frac{\partial P(\mathbf{O}^\theta \mid \lambda_v)}{\partial \lambda_v} \tag{8}$$

where

$\Pi_\theta^v = \dfrac{P^\eta(\mathbf{O}^\theta \mid \lambda_v)}{\sum_{m=1,m\neq\theta}^{M} P^\eta(\mathbf{O}^\theta \mid \lambda_m)}$ is the relative similarity measure between speaker model λ_v and λ_θ against all competitors of λ_θ in $\mathbf{O}^\theta$.

3. RE-ESTIMATION FORMULA

Let $\gamma_t^{vc}(k)$ be the probability at time t with the kth mixture component accounting for $\mathbf{o}_t^{vc}$ for λ_v. It can be defined as follows

$$\gamma_t^{vc}(k) = \frac{N_v(\mathbf{o}_t^{vc}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k=1}^{K} N_v(\mathbf{o}_t^{vc}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \tag{9}$$

Optimizing $D(\Lambda)$ with Lagrange multiplier rule, we can get the recursive re-estimation formulae of λ_v's parameter for multiple observation sequences as follows

$$c_k^v = \frac{\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k)}{\sum_{k=1}^{K} \left[\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k) \right]} \tag{10a}$$

$$\boldsymbol{\mu}_k^v = \frac{\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) \mathbf{o}_t^{vc} - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k) \mathbf{o}_t^{\theta c}}{\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k)} \tag{10b}$$

$$\boldsymbol{\Sigma}_k^v = \frac{\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) (\mathbf{o}_t^{vc} - \boldsymbol{\mu}_k^v)(\mathbf{o}_t^{vc} - \boldsymbol{\mu}_k^v)' - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k) (\mathbf{o}_t^{\theta c} - \boldsymbol{\mu}_k^\theta)(\mathbf{o}_t^{\theta c} - \boldsymbol{\mu}_k^\theta)'}{\frac{1}{C_v} \sum_{c=1}^{C_v} \sum_{t=1}^{T_c^v} \gamma_t^{vc}(k) - \sum_{\theta=1,\theta\neq v}^{M} \frac{1}{C_\theta} \sum_{c=1}^{C_\theta} \Pi_{\theta c}^v \sum_{t=1}^{T_c^\theta} \gamma_t^{\theta c}(k)} \tag{10c}$$

where C_v is the number of observation sequences of speaker v, T_c^v is the length of the cth observation sequence for the speaker model λ_v.

4. COMPETITIVE SPEAKER

Through the adjusting of the relative similarity measure $\Pi_{\theta c}^v$, the contribution of different competitive speaker can be weighted automatically. To reduce the computation complexity, we usually use a threshold to select the most competitive speakers

$$\log P(\mathbf{O}^v \mid \lambda_\theta) - \log P(\mathbf{O}^v \mid \lambda_v) > THR \tag{11}$$

where THR is the pre-fixed threshold value to be decided experimentally. Although this method has been proved to be effective for speech recognition, it has some problems in the area of speaker identification:

- Applying the same threshold to all speakers sometimes resulted in a small portion of speakers having too many competitive speakers. It may cause the re-estimation of model parameters deteriorated especially for the case of short training utterances.
- It is more likely that sentences are used as the training data and all of them can be identified correctly. This makes it more difficult to set a suitable threshold for all the speakers. If THR is not low enough, MMD will converge to the ML method and lose its discriminative capability.

To resolve these problems, we propose to select only the top competitive by calculating the log probability shown as follows

$$\hat{\theta} = \arg\max_{\theta} P(\mathbf{O} \mid \lambda_\theta) \tag{12}$$

with $\hat{\theta}$ being the identified speaker that attains the highest probability score among all competing speakers. In the training procedure, only the statistical accumulators of this top competitive speaker will be calculated. Actually, when using the sentences as the training data, it is found that in most cases the weight of the top competitive speaker is nearly 1.0 and the other speakers can be neglected.

5. EXPERIMENTAL RESULTS

Experiments were conducted based on the TIMIT corpus. TIMIT contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect regions. Each speaker includes 2 SA sentences, which are the same across all speakers, 5 SX sentences and 3 SI sentences. Both the train and test sections include 8 dialect regions labeled from DR1 to DR8.

Speech utterances from TIMIT were parameterized with *mel* frequency cepstral coefficients (MFCC). The signal was pre-emphasized using a coefficient of 0.97 and analyzed with a frame rate of 10 ms. Each frame of speech was windowed with a 25ms Hamming window and represented by a 36 dimensional feature vector, which consists of 12 MFCCs with the first and second differentials appended. The feature extraction was conducted with the HTK toolkit [8]. In the following experiments, two kinds of training utterances were used to verify the performance of the MMD training.

5.1 Short Training Utterance

In the first experiment, we used isolated words as short training utterance to demonstrate the effectiveness of the proposed speaker selection strategy. Using isolated words rather than sentences assured that the model distance $D(\Lambda)$ could be calculated according to Equation 6. Otherwise the calculation might be out of precision. This experiment also illustrated the relation between $D(\Lambda)$ and the identification accuracy.

There are 38 speakers in DR1 of the train section. From sentences SA1 and SA2 of each speaker, we extracted 20 words which included "*all, an, ask, carry, dark, don't, greasy, had, in, like, me, oily, rag, she, suit, that, wash, water, year, your*". That is, each speaker had 20 uttered words for the training. Each GMM model consisted of 8 mixtures. Model parameters were initialized with the *K*-means algorithm.

In the MMD training, the value of η and competitive threshold was set as 1.0 and 5.0 respectively. During the training procedure, it was very likely that the competitive speaker had a higher log probability than the speaker to be trained. This might cause the final statistical accumulators become negative and made the parameters of GMM fail to be re-estimated. To avoid this problem, one alternative solution is to add up the weight of the trained speaker. In this experiment, we set it as the value of speaker number. The performances of MMD using the threshold and top competitive method are compared as follows

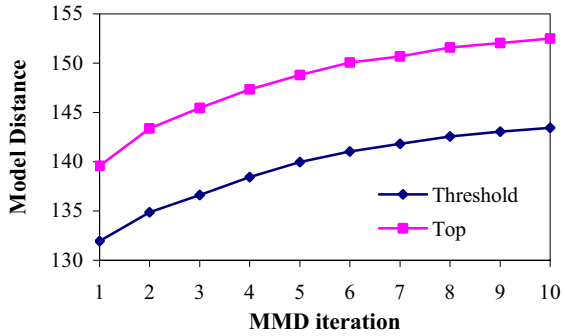

Figure 1 Comparison of the model distance $D(\Lambda)$

With the number of iterations increased, the value of $D(\Lambda)$ was increased monotonously. The top competitive method always has higher value of $D(\Lambda)$ than the threshold method. This indicates that the GMMs trained by the top competitive method have better discriminative capability than the GMMs trained by the threshold method. Their identification performances for the training set are further compared as follows

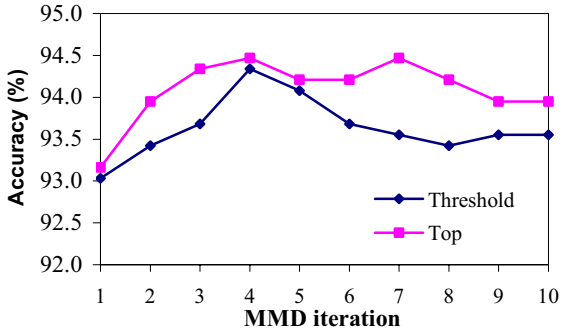

Figure 2 Comparison of identification accuracy

We can see that the top competitive method also has higher identification accuracy. In the fourth iteration, both methods reach the best performance. Other data were tested and had similar results. Thus we usually get the training result within five MMD iterations in practice. And only the top competitive method is applied. Using the other eight sentences of each speaker as the test data, the performance of MMD is compared with the ML training as follows

Method	Training Set	Test Set
ML	93.03%	84.87%
MMD	94.47%	87.17%

Table 1 Identification results of 38 speakers

5.2 Long Training Utterance

It is preferred to use longer training data like sentences for speaker identification, since they include more speaker information. In the second experiment, we still used 38 speakers from DR1 and each GMM model consisted of 16 mixtures. The number of training sentences was ranged from 2 to 8 and the rest sentences of the same speaker were used as the test data. Since the identification results for training set were always 100%, we only compare the results for test set in Table 2.

For the case of 7 and 8 training sentences, it is seen that given enough training data, there is no difference between ML and MMD. However for the other cases, MMD has better performance than ML. The improvement is more obvious when the training data is limited.

Train/Test Sentences	Training Method	
	ML	MMD
2/8	91.12%	94.41%
3/7	96.99%	98.87%
4/6	99.12%	99.56%
5/5	97.89%	100%
6/4	98.03%	100%
7/3	99.12%	99.12%
8/2	100%	100%

Table 2 Results of different training sentences

For the case of 2 training sentences, the accuracy of ML is 91.12%, while MMD can reach 94.41%. The identification performance is improved by 3.61%. For 3 training sentences, the results of ML and MMD are 96.99% and 98.87% respectively. That is, the error rate is reduced from 3.01% to 1.13% through discriminative training.

5.3 Dialect Performance

This experiment was designed to verify the robustness of MMD. We merged the speakers from train and test section in the same region. The mixture number was fixed as 16 and there were 3 training sentences for each speaker model. The test results of ML and MMD are summarized as follows

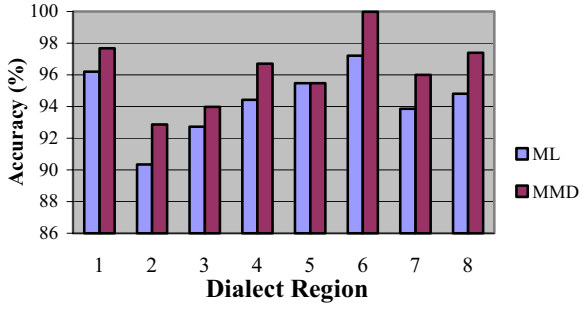

Figure 3 Performance comparisons for DR1-8 in TIMIT

Except the fifth dialect DR5, MMD always has better performance than ML. In DR6, the accuracy of MMD is 99.97%, which is significantly better than the 97.21% of ML. Finally, we used the total 630 speakers in TIMIT to make a further comparison. The identification accuracy of ML is 84.38%, while MMD can reach 88.57%. The performance is improved by 4.97%.

6. CONCLUSION

This paper has described a discriminative training method for GMM-based speaker identification. We proposed to apply the MMD to speaker identification problem and further proposed a novel competitive speaker selection strategy to it. Experimental results based on the words extracted from TIMIT shows that this selection strategy has a better performance than the threshold method used by the original MMD. And the sentences and dialect experiments have demonstrated that our training approach is very attractive for the limited training data compared with the traditional ML method.

7. ACKNOWLEDGMENT

This work is supported by City University Strategic Grant 7001416.

8. REFERENCES

[1] S. Furui, "An Overview of Speaker Recognition Technology," *Automatic Speech and Speaker Recognition*. Edited by Lee, C., Soong, F., Paliwal, K. Kluwer Academic Press, 1996.

[2] D.A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication* 17 (1995) 91-108.

[3] S. Kwong, Q.H. He, K.F. Man, and K.S. Tang, "A maximum model distance approach for HMM-based speech recognition," *Pattern Recognition* 31 (3) (1998) 219-229.

[4] Q.H. He, S. Kwong, K.F. Man, and K.S. Tang, "Improved maximum model distance for HMM training," *Pattern Recognition*, Vol. 33, 2000, pp.1749-1758.

[5] C.S. Liu, C.H. Lee, W. Chou, B.H. Juang, and A.E. Rosenberg, "A study on minimum error discriminative training for speaker recognition," *Journal of Acoust. Soc. Am.* 97(1), 637-648, January 1995.

[6] C. Martin del Alamo, F.J. Caminero Gil, C. de la Torre Munilla, and L. Hernanadez Gomez, "Discriminative training of GMM for speaker identifcation," in *Proc. ICASSP*, vol. 1, pp. 157-160, 1996.

[7] O. Siohan, A.E. Rosenberg, and S. Parthasarathy, "Speaker identification using minimum classification error training," in *Proc. ICASSP*, vol. 1, pp. 109-112, 1998.

[8] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK book (for htk version 3.0)*. (htk.eng.cam.ac.uk/prot-docs/HTKBook/htkbook.html), July 2000.