DISCRIMINATIVE TRAINING OF GMM USING A MODIFIED EM ALGORITHM FOR SPEAKER RECOGNITION

Konstantin P. Markov *Seiichi Nakagawa*
markov@slp.tutics.tut.ac.jp *nakagawa@tutics.tut.ac.jp*

Dept. of Information and Computer Sciences, Toyohashi Univ. of Technology,
1-1 Hibarigaoka, Tempaku-chou, Toyohashi-shi, Aichi-ken, 441 JAPAN

ABSTRACT

In this paper, we present a new discriminative training method for Gaussian Mixture Models (GMM) and its application for the text-independent speaker recognition. The objective of this method is to maximize the frame level normalized likelihoods of the training data. That is why we call it the Maximum Normalized Likelihood Estimation (MNLE). In contrast to other discriminative algorithms, the objective function is optimized using a modified Expectation-Maximization (EM) algorithm which greatly simplifies the training procedure. The evaluation experiments using both clean and telephone speech showed improvement of the recognition rates compared to the Maximum Likelihood Estimation (MLE) trained speaker models, especially when the mismatch between the training and testing conditions is significant.

1. INTRODUCTION

In the resect years, the discriminative training methods have attracted many researchers attention because they help in improving the performance of the speech recognition systems. It has been shown that methods, such as Minimum Classification Error (MCE) and Maximum Mutual Information (MMI), are also effective in the GMM based speaker recognition systems [1, 2]. In general, the discriminative learning outperforms the standard ML estimation when the parametric distribution function (usually Gaussian) of the models is inconsistent with the actual data distribution and when the amount of training data is limited and does not allow reliable parameter estimation.

MCE method uses classification errors on the training data directly in its objective function. However, with clean speech where nearly 100% recognition rate can be achieved on the training data, its advantage is greatly reduced since no misclassification occurs. The objective of the MMI method, in the other hand, is to maximize the class a posteriori probability, which is not directly connected with the classification accuracy, but with big number of reference speakers it becomes computationally expensive. In both methods, usually, the Generalized Probabilistic Descent (GPD) algorithm is used for optimization of the objective function. Although it is complicated and requires adjustment of several free parameters, its effectiveness has been proven. However, as pointed in [1], when applied for speaker recognition it has some drawbacks and heuristic corrections are necessary for achieving good performance.

We have developed a new discriminative training algorithm, which widens the separation between the most competitive classes or, in other words, between acoustically most close speakers. The objective function to be maximized uses the likelihood ratio between the target speaker model and its "cohort" of competing models taken at frame level [3]. For the optimization, we use the standard Expectation Maximization (EM) algorithm with some modifications. This allowed us to come up with a simple and tractable re-estimation procedure. Evaluation experiments using clean speech database showed that this algorithm is effective especially when there is mismatch between the model, i.e. normal distribution, and data distributions.

2. MNLE ALGORITHM

2.1. MMI and MNLE Objectives

Given N classes (speakers) and training data X_n for each class, the MMI objective is to maximize the class a posteriori probability. Generally, it is given by:

$$\mathcal{F}_{MMI}(\Lambda) = \sum_{n=1}^{N} \log \frac{p(X_n|\lambda_n)}{\sum_{i=1}^{N} p(X_i|\lambda_i)} \qquad (1)$$

We can also define a frame level MMI objective as:

$$\mathcal{F}_{MMI_f}(\Lambda) = \sum_{n=1}^{N} \sum_{t=1}^{T_n} \log \frac{p(x_{nt}|\lambda_n)}{\sum_{i=1}^{N} p(x_{it}|\lambda_i)} \qquad (2)$$

which is used in [2] and optimized via GPD. Both these objectives require the likelihoods from all speakers to be known in the learning process.

The objective function of the Maximum Normalized Likelihood Estimation (MNLE) algorithm is defined as follows:

$$\mathcal{F}_{MNLE}(\Lambda) = \sum_{n=1}^{N} \sum_{t=1}^{T_n} \log \frac{p(x_{nt}|\lambda_n)}{\frac{1}{B}\sum_{b=1}^{B} p(x_{nt}|\lambda_{nb})} \qquad (3)$$

which is similar to $\mathcal{F}_{MMI_f}$ with the difference that instead the a posteriori probability a likelihood normalization is used. $\lambda_{nb}, b = 1, \ldots, B$ are the normalization background speaker models for speaker n. The choice of this background models greatly influences the performance of the MNLE. We have all reasons to believe that the target speaker is most often misclassified with some of its acoustically most close speakers. Therefore, choosing those speakers as background speakers will ensure that maximizing the objective function will result in better separation between them and the target speaker.

2.2. Learning Algorithm

Equation (**3**) can be rewritten as:

$$\mathcal{F}_{MNLE}(\Lambda) = \sum_{n=1}^{N}\sum_{t=1}^{T_n} \log p(x_{nt}|\lambda_n) - \sum_{n=1}^{N}\sum_{t=1}^{T_n} \log(\frac{1}{B}\sum_{b=1}^{B} p(x_{nt}|\lambda_{nb})) \quad (4)$$

$$= \mathcal{F}_{MLE}(\Lambda) - \mathcal{F}_D(\Lambda) \quad (5)$$

where $\mathcal{F}_{MLE}(\Lambda)$ is, actually, the objective of the conventional MLE training and $\mathcal{F}_D(\Lambda)$ is a correction term responsible for the discriminative nature of the algorithm.

The objective function can also be decomposed into an individual objective functions $\mathcal{F}_n(\Lambda)$ for each speaker as:

$$\mathcal{F}_{MNLE}(\Lambda) = \sum_{n=1}^{N} \mathcal{F}_n(\Lambda) \quad (6)$$

The Q function of the EM algorithm is defined as follows:

$$Q(\Lambda|\overline{\Lambda}) = \sum_{t=1}^{T}\sum_{y_t} \frac{f(x_t, y_t|\Lambda)}{f(x_t|\Lambda)} \log f(x_t, y_t|\overline{\Lambda}) \quad (7)$$

where y_t is the unobservable data, which specifies some pdf and $\overline{\Lambda}$ denotes the new parameter set. Applying this formula to GMM with M mixtures of Gaussian densities we have:

$$Q(\Lambda|\overline{\Lambda}) = \sum_{n=1}^{N}\sum_{t=1}^{T_n}\sum_{j=1}^{M} \frac{f(x_{nt}, \omega_{nj}|\Lambda)}{f(x_{nt}|\Lambda)} \log f(x_{nt}, \omega_{nj}|\overline{\Lambda}) \quad (8)$$

where ω_{nj} specifies the n^{th} GMM's j^{th} mixture. Now, based on the normalized likelihood formulation, for $f()$'s we have:

$$f(x_{nt}, |\Lambda) = \frac{p(x_{nt}|\lambda_n)}{\frac{1}{B}\sum_{b=1}^{B} p(x_{nt}|\lambda_{nb})} \quad (9)$$

$$f(x_{nt}, \omega_{nj}|\Lambda) = \frac{c_{nj} b_{nj}(x_{nt})}{\frac{1}{B}\sum_{b=1}^{B} p(x_{nt}|\lambda_{nb})} \quad (10)$$

$$f(x_{nt}, \omega_{nj}|\overline{\Lambda}) = \frac{\overline{c}_{nj} \overline{b}_{nj}(x_{nt})}{\frac{1}{B}\sum_{b=1}^{B} p(x_{nt}|\overline{\lambda}_{nb})} \quad (11)$$

where $b()$ denotes a Gaussian pdf and c_{nj} is the mixture weight. Inserting the Eqs.(**9**), (**10**) and (**11**) in Eq.(**8**) we get the final formula for the Q function. In Eq.(**11**) the new model parameters of the background speakers are required. But, they may not be available since the re-estimation proceeds model by model. However, they can be approximated by the same parameters obtained in the previous iteration.

After taking the derivative of the Eq.(**8**) with respect to each model's parameter, the following re-estimation equations can be derived (detailed derivation can be found in [4]):

$$\overline{c}_{nj} = \frac{\sum_{t=1}^{T_n} p_{njt} - \sum_{n'}\sum_{t=1}^{T_{n'}} p^b_{njn't}}{T_n - \sum_{n'}\sum_{t=1}^{T_{n'}} P_{n't}} \quad (12)$$

$$\overline{\mu}_{nj} = \frac{\sum_{t=1}^{T_n} p_{njt} x_{nt} - \sum_{n'}\sum_{t=1}^{T_{n'}} p^b_{njn't} x_{n't}}{\sum_{t=1}^{T_n} p_{njt} - \sum_{n'}\sum_{t=1}^{T_{n'}} p^b_{njn't}} \quad (13)$$

$$\overline{\Sigma}_{n,j} = \frac{\sum_{t=1}^{T_n} p_{njt} A_{njt} - \sum_{n'}\sum_{t=1}^{T_{n'}} p^b_{njn't} A_{njn't}}{\sum_{t=1}^{T_n} p_{njt} - \sum_{n'}\sum_{t=1}^{T_{n'}} p^b_{njn't}} \quad (14)$$

where:

$$P_{n't} = \frac{p(x_{n't}|\lambda_n)}{\sum_{b=1}^{B} p(x_{n't}|\lambda_{n'b})} \quad (15)$$

$$p_{njt} = \frac{c_{nj} b_{nj}(x_{nt})}{p(x_{nt}|\lambda_n)} \quad (16)$$

$$p^b_{njn't} = \frac{c_{nj} b_{nj}(x_{n't})}{\sum_{b=1}^{B} p(x_{n't}|\lambda_{n',b})} \quad (17)$$

$$A_{njt} = (x_{nt} - \mu_{nj})(x_{nt} - \mu_{nj})^t \quad (18)$$

$$A_{njn't} = (x_{n't} - \mu_{nj})(x_{n't} - \mu_{nj})^t \quad (19)$$

In all these equations "i" denotes the parameters or training vectors from those models for which the target model n has been acting as a background model. It is easy to recognize that the first terms in the re-estimation formulas are the same as in standard MLE re-estimation equations, but now a similar correction terms are subtracted from them. For example, the $p^b_{njn't}$ (Eq.(**17**)) can be viewed as a posteriori probability of $x_{n't}$ with regard to all mixtures from the background models. It corresponds to the MLE posteriori probability p_{njt} (Eq.(**16**)). Thus, the Eqs.(**12**)-(**14**) can be expressed in the following generalized form:

$$\theta^{i+1} = \frac{MLE^i_{numer} - COR^i_{numer}}{MLE^i_{denom} - COR^i_{denom}} \quad (20)$$

In practice, when we deal with limited training data, there is a danger that the Eqs.(**12**)-(**14**) may become negative and, therefore, the correction terms must be scaled down. Another complication is that because of the denominator term in Eq.(**11**), the monotonic increase of the Q function is no longer guaranteed. Indeed, in our preliminary experiments, the value of the objective function was increasing with some fluctuations. Eq.(**4**) suggests that maximization of the objective function $\mathcal{F}(\Lambda)$ can be done by first maximizing the $\mathcal{F}_{MLE}(\Lambda)$, i.e. performing ML estimation, and then using obtained ML parameters as a starting point to proceed with the training. Therefore, the re-estimation can be performed using:

$$\theta^{i+1} = \frac{MLE^0_{numer} - \epsilon\sum_{j=1}^{i} COR^j_{numer}}{MLE^0_{denom} - \epsilon\sum_{j=1}^{i} COR^j_{denom}} \quad (21)$$

where ϵ is the scaling parameter which corresponds to the learning step of the GPD algorithm.

In the standard MLE training, the algorithm converges when the increase of the objective function (the likelihood of the training data) is zero or below some small threshold. In the case of MNLE training, the objective function

(see Eq.(**6**)) will have the maximum when all individual objective functions $\mathcal{F}_n(\Lambda)$ are also maximized. As we have explained in [4], because of the correction terms, the $\mathcal{F}_n(\Lambda)$ has the maximum after which it starts to decrease. In order to prevent this and to have a clear convergence criterion, the following algorithm is used:

Step 1 Begin with ML trained models. Mark all the models as to be re-estimated.

Step 2 Compute the new parameters of the models marked to be re-estimated.

Step 3 Check each speaker individual objective function $\mathcal{F}_n(\Lambda)$ and if there a maximum has been achieved mark the model not to be re-estimated further.

Step 4 Repeat Step 2 and Step 3 until all models are not marked to be re-estimated.

This approach guarantees the monotonic increase of the overall objective function. An example is shown in Fig.**1** where the value of the $\mathcal{F}_{MNLE}(\Lambda)$ is plotted with respect to the iteration number.

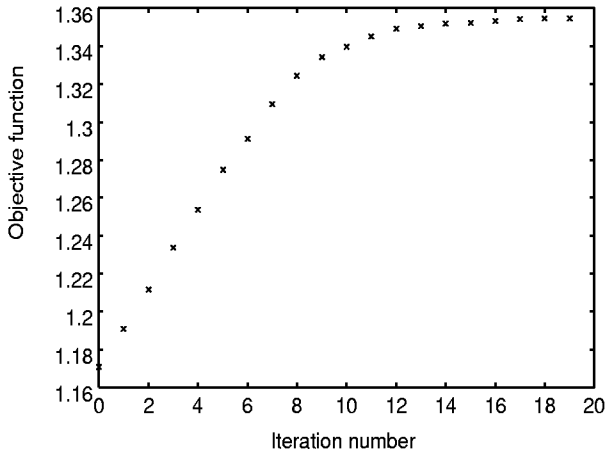

Figure 1: MNLE objective function value vs. iteration number.

3. EXPERIMENTS

3.1. Databases

For the experiments with clean speech, we used NTT database consisting of recordings of 22 male and 13 female speakers, collected in 5 sessions over 10 months in a sound proof room. For training, 5 equal and 5 different sentences uttered at normal speed for each speaker from one session were used. Five other sentences uttered at normal, fast and slow speeds from the other four sessions were used as text-independent test data. The input speech was sampled at 12 kHz. 10 mel-cepstrum coefficients were calculated by the 14th order LPC analysis at every 8 ms with a window of 21.33 ms. In addition, 10 regressive (Δ) coefficients, were obtained and treated in the experiments as a separate feature stream. Each session's data were also mean normalized (CMN) and silence parts were removed.

The second database we used is NTIMIT corpus. It provides speech identical to TIMIT database, except that the speech is degraded through carbon microphones and real telephone line conditions. In the experiments, only the test portion (168 speakers) of the database was used. The training data consist of one SA, five SX and two SI sentences. The remaining one SA and one SI sentences are individually used as tests. Since the noise varies from transmission over different telephone lines, eliminating the silence (filled with noise) is important for good performance. This was done using an adaptive voice activity detector based on the ESPS package [5] which estimates the noise level at the beginning of each utterance and then extracts only the speech segments. Then, the speech data were transformed into 10 mel filter bank cepstral coefficients - MFCC at every 10 ms with window of 30 ms. We used 22 mel-filters covering only the telephone bandwidth (300-3400 Hz). Further, 10 (Δ) coefficients were calculated and treated as a separate feature stream. Cepstral mean normalization is not applied because we found that it degrades the performance as also pointed in [6].

3.2. NTT Database Results

In the evaluation experiments, we used GMM with different number of mixtures and full or diagonal covariance matrices. The number of background (cohort) speakers was set to 5. In the MNL training procedure, a fixed learning step was used. The results presented in Table **1** were obtained using the standard maximum likelihood decision. Models trained using the conventional ML estimation algorithm serve as a baseline for comparison. In addition, we trained the same GMMs using 20 iterations of the MCE/GPD algorithm. However, we achieved only minor improvements with the MCE training for the slow and fast tests. This shows that the MCE objective is not effective with clean speech where the identification rate with the training data is 100%, i.e. no misclassification occurs.

Table 1: Speaker identification rates (%).

Num. of mixtures	Cov. matrix	Training alg.		
		MNLE	MCE/GPD	MLE
Normal speed test				
4	full	96.0	94.1	94.1
8	full	97.0	97.0	97.0
32	diagonal	96.3	95.9	95.9
64	diagonal	96.0	95.9	95.9
Fast speed test				
4	full	93.6	91.0	91.0
8	full	94.4	94.1	94.0
32	diagonal	92.1	91.8	91.7
64	diagonal	92.7	92.6	92.6
Slow speed test				
4	full	92.4	90.9	90.9
8	full	93.4	93.0	92.7
32	diagonal	93.0	91.8	91.6
64	diagonal	93.4	92.3	92.0

As shown in the results, the MNL trained models outperform both the baseline and MCE trained model in all cases except one. In this case, 8 mixture GMM and normal speed test, the maximum identification rate is achieved, which suggests that this model best matches with the data distribution and, thus, the effect of the discriminative training is reduced to zero. This is justified by the fact, that the same model shows an improvement with slow or fast test data which introduce a significant distribution mismatch. The bigger improvement achieved for the 4 mixture and 32 mixture models compared to 8 mixture and 64 mixture models can be explained with the bigger mismatch in these models.

Table **2** presents the speaker verification equal error rates for the three training algorithms. Again, the MNLE gives the best results improving the performance even in the 8 mixture GMM and normal speed test case.

Table 2: Speaker verification equal error rates (%).

Num. of mixtures	Cov. matrix	Training alg.		
		MNLE	MCE/GPD	MLE
Normal speed test				
4	full	1.36	1.64	1.64
8	full	1.12	1.23	1.18
32	diagonal	1.15	1.29	1.29
64	diagonal	0.88	1.05	1.07
Fast speed test				
4	full	2.03	2.24	2.26
8	full	1.23	1.41	1.43
32	diagonal	2.21	2.80	2.88
64	diagonal	1.84	2.61	2.66
Slow speed test				
4	full	2.55	2.92	2.96
8	full	1.76	1.97	2.06
32	diagonal	2.18	2.31	2.36
64	diagonal	1.93	2.49	2.57

3.3. NTIMIT Database Results

In the experiments with NTIMIT database, we trained the GMMs using both the MLE and MNLE methods. The number of background speakers for the MNLE training was set 10. The speaker identification rates using a maximum likelihood test are shown in Table **3**. As we can see, in all cases, significant improvements were achieved when the models were trained using MNLE. Although the overall results are not high, which suggest severe mismatch between the training and testing conditions, the MNLE training shows to be effective.

Speaker identification on NTIMIT database is a challenging problem and the few published results are quite different for similar train/test data divisions. Thus, an identification rate of 60.7% for all the 630 speakers was reported in [6] and only 26.7% in [7]. Such variations can be contributed to the different front-ends and features, rather than to the different modeling approaches.

Table 3: Speaker identification rates (%).

Num. of mixtures	Cov. matrix	Training alg.	
		MNLE	MLE
4	full	38.1	35.1
8	full	37.5	33.3
32	diagonal	41.7	37.8
64	diagonal	38.7	36.3

4. CONCLUSIONS

We introduced a new discriminative training method for GMM which, in contrast to other discriminative methods, is based on the well known EM algorithm. Evaluation experiments on clean speech database showed that it is effective and outperforms the MCE method. With the telephone speech, where the mismatch between the model distribution and actual data distribution is significant, the MNLE method was also effective and showed even bigger improvements of the system performance.

The future work will be focused on the refinements of this algorithm, such as applying variable and/or class dependent learning step.

REFERENCES

1. C. Alamo, F. Gil, C. Muninlla, and H. Gomez, "Discriminative training of GMM for speaker identification," in *Proc. ICASSP'96*, pp. 89–92, 1996.
2. H. Li, J. Haton, and Y. Gong, "On MMI learning of Gaussian mixture for speaker models," in *Proc. EUROSPEECH'95*, vol. 1, pp. 363–366, 1995.
3. K. P. Markov and S. Nakagawa, "Frame level likelihood normalization for text-independent speaker identification using Gaussian Mixture Models," in *Proc. ICSLP*, pp. 1764–1767, 1996.
4. K. P. Markov and S. Nakagawa, "Discriminative training of GMM using frame level likelihood normalization for speaker recognition," Tech. Rep. SP97-18, IEICE, June 1997.
5. Entropic Research Laboratory, Inc., *ESPS Programs*, 1996.
6. D. A. Reynolds, "Large population speaker identification using clean and telephone speech," *IEEE Signal Processing Letters*, vol. 2, pp. 46–48, Mar. 1995.
7. F. Bimbot, I. Magrin-Chagnolleau, and L. Mathan, "Second-order statistical measures for text-independent speaker identification," *Speech Communication*, vol. 17, pp. 177–192, Aug. 1995.