A DISCRIMATIVE TRAINING ALGORITHM FOR GAUSSIAN MIXTURE SPEAKER MODELS

Jialong He, Li Liu, and Günther Palm
Abteilung Neuroinformatik
University of ULM, GERMANY
jialong@neuro.informatik.uni-ulm.de

ABSTRACT

The Gaussian mixture speaker model (GMM) is usually trained with the expectation-maximization (EM) algorithm to maximize the likelihood (ML) of observation data from an individual class. The GMM trained based the ML criterion has weak discriminative power when used as a classifier. In this paper, a discriminative training procedure is proposed to fine-tune the parameters in the GMMs. The goal of the training is to reduce the number of misclassified vector groups. Since a vector group can be thought as derived from a short sentence, this training procedure optimize the speaker identification performance more directly. Even though the algorithm itself is based on an heuristic idea, it works fine for many practical problems. Besides, the training speed is very fast. In an evaluation experiment with the YOHO database, when each speaker is modeled with 8 mixtures, the identification rate increases from 83.8% to 92.4% after applying this discriminative training algorithm.

1 INTRODUCTION

The Gaussian mixture speaker model is a probabilistic model by which the distribution of data is modeled as a linear combination of several multivariate Gaussian functions. The GMM is usually trained with the EM algorithm to maximize the likelihood (ML) of the observation data from an individual class [1]. However, the model trained based on this criterion lacks discriminative power when used as a classifier. One method that has been tried to remedy this problem is to maximize mutual information (MMI). The MMI criterion directly increases the *a posteriori* probability of a class on model learning [2]. Although the criterion is attractive, there is no efficient and robust algorithm. One has to go back to the gradient decent scheme which is very time consuming. In this paper, we propose a new training procedure for the GMM based classifier. The basic principle is borrowed directly from the GVQ training [3], where the model is optimized to give a lower classification error rate for vector groups. A vector group can be thought as derived from a short sentence, therefore, this training procedure optimizes the sentence level performance, i.e., speaker identification rate, more directly. Evaluation experiments have been conducted to compare the performance of the GMM trained with different methods, it is shown that the GMM trained with this new method provides a much higher speaker identification rate than that solely trained with the EM algorithm.

2 ALGORITHM

In the GMM, the distribution density of feature vectors is modeled as a weighted sum of M multivariate Gaussian functions

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} c_i b_i(\vec{x}) \tag{1}$$

where c_i are the mixture weights and $b_i(\vec{x})$ are D-variate Gaussian functions. In practice, the diagonal covariance matrices are exclusively used. The complete Gaussian mixture density is parameterized by the mean vectors, variance vectors and mixture weights from all component densities. These parameters are collectively represented by the notation $\lambda = \{c_i, \vec{u}_i, \vec{\sigma}_i\}$. To design a GMM is to estimate the parameters of λ, which in some sense best matches the distribution of training vectors. By far the most popular and well-established method is to maximize the likelihood using the EM algorithm.

For a given sentence, a vector sequence, $\{\vec{x}_t\}_1^T$, can be obtained using the short-term analysis techniques. Suppose that the vectors are statistically independent, the average log-likelihood of the vector sequence with the model λ_s is given by

$$\ell(X|\lambda_s) = \frac{1}{T}\sum_{t=1}^{T} log\left(p(\vec{x}_t|\lambda_s)\right) \tag{2}$$

The test sentence will be classified as from the speaker whose model has the largest average log-likelihood, i.e.,

$$ID = \underset{1\le s\le L}{arg\,max}\left\{\ell(X|\lambda_s)\right\} \tag{3}$$

It is evidenced in Eq. (2) and (3) that a classification decision is made based on several vectors, therefore, the model should be optimized for the vector sequences. Besides, the output from every model plays an important role.

Suppose that there are *L* speakers in the system, all models have been created with the EM algorithm. Now the training data are labeled with the corresponding class membership, the models are retrained simultaneously with the following iteration procedure.

1. Randomly select a speaker, denote its class as j.
2. From the training data with the class label *j*, select *N* vectors $X = \{\vec{x}\}_1^N$ as a vector group.
3. Calculate the average log-likelihood from each GMM using Eq. (2), here *T*=*N*.
4. If the following two conditions are satisfied, go to step 5, otherwise go to step 1.
   (a) $\ell(X|\lambda_i)$ is the largest value but $i \neq j$;
   (b) $\left(\ell(X|\lambda_i) - \ell(X|\lambda_j)\right) / \ell(X|\lambda_i) < w$
   w is a small constant.
5. For each misclassified vector in $\{\vec{x}_t\}_1^N$, find out its largest Gaussian component from the model λ_i (denoted as m) and the largest Gaussian component from the model λ_j (denoted as n), adjusting the two mean vectors with

$$\vec{u}_{jn} \Leftarrow \vec{u}_{jn} + \alpha(\vec{x}_t - \vec{u}_{jn})$$
$$\vec{u}_{im} \Leftarrow \vec{u}_{im} - \alpha(\vec{x}_t - \vec{u}_{im})$$

   where α is a learning rate. After all vectors being processed, go to step 1.

The above iteration procedure can be repeated until the specified iteration number is reached. It is easy to verify that after the modification in step 5, the likelihood value from the correct model, λ_j, increases and that from the wrong model decreases. There are several parameters in this algorithm, including the learning rate (α), the number of training epochs, (a training epoch is defined as the number of iterations that equals the total number of training vectors), and the size of vector groups (*N*). Among these, the choice of *N* is very important. We have tried two different ways of selecting vectors to compose vector groups. The first approach is known as the sequential selection method by which the vectors in a group are selected from adjacent segments of speech. Another method is the random selection method. In this case, the vectors in a group may come from different sentences. We denote the model trained with above procedure as the learning Gaussian mixture model (LGMM).

3 EXPERIMENT

3.1 Speech Database and Feature Vectors

The evaluation speech came from the YOHO corpus [4]. A subset of the database including 20 male and 20 female speakers was used in the following experiments. In the test phase, each individual digit string (three two-digit numbers) was used as a test utterance, thus each evaluation result was obtained from 1600 tests (4 strings $\times$ 10 sessions $\times$ 40 speakers). From a voiced segment of speech, 16 MFCC coefficients were calculated to compose a feature vector. The analysis window size was 32 ms (256 samples) with 16 ms overlapping. The average length of the test utterances is about 48 frames (768 ms) after discarding the unvoiced segments.

3.2 The two-class problem

To see what happens when the GMM is trained with the method proposed in this paper, we first consider the case of two speakers. For a given observation vector $\vec{x}$, the minus-log likelihood-ratio is defined as

$$h(\vec{x}) = log\left(p(\vec{x}|\lambda_2)\right) - log\left(p(\vec{x}|\lambda_1)\right) \qquad (4)$$

Then $h(\vec{x})$ is a discriminant function of $\vec{x}$, that is, the vector $\vec{x}$ is assigned to the first speaker if $h(\vec{x}) < 0$, otherwise, $\vec{x}$ is classified as from the second speaker. We denote the data from the first speaker as ω_1 and those from the second speaker as ω_2. The histograms of $h(\vec{x})|\omega_1$ and $h(\vec{x})|\omega_2$ are plotted in Figure 1.

From the definition of $h(\vec{x})$, it is easy to understand that the shade area represents the portion of data from the first speaker being misclassified as from the second speaker. Similarly, the grid area is the error probability of data from the second speaker. Intuitively, the total areas of shade and grid in the LGMM panel is smaller than that in the GMM panel. From the mean values (η_i) and standard deviations (δ_i) shown in the figure, we see that the main factor that leads to less overlapping between two curves in the LGMM panel is due to the decrease of the variances.

If we assume the distributions of $h(\mathbf{x})|\omega_i$ are normal with $\eta = |\eta_1| = |\eta_2|$ and $\delta = \delta_1 = \delta_2$, it is easy to show that the correct classification rate is $\Phi(\eta/\delta)$, where $\Phi(\cdot)$ is the normal integration defined as

$$\Phi(\alpha) = \int_{-\infty}^{\alpha} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx \qquad (5)$$

Clearly, a larger ratio of η/δ will give a better classification performance. For the data shown in this figure, the average value of η/δ increases from 0.73 to 1.05 after applying the discriminative training procedure.

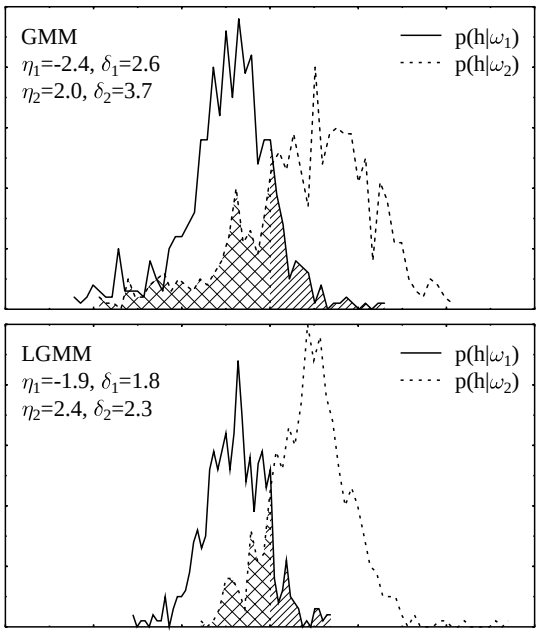

Figure 1 Upper panel: histograms of $h(x)|\omega_i$ from the GMM; lower panel: histograms from the LGMM.

3.3 Evaluation results

There are several algorithm related parameters that should be specified in advance. We systematically evaluated the effects of these parameters in respect of speaker identification performance. Figure 2 displays the speaker identification rate vs. the number of training epochs under different learning rates. Since the model was initialized with the EM algorithm, the starting point indicates the performance of the standard GMM. It is seen that in all cases the speaker identification performance improves after applying the discriminative training algorithm.

Training epochs	0	1	2	3
Log-likelihood	-3.8	-7.4	-9.0	-9.4

Table 1 Average log-likelihood vs. the number of training epochs.

From the results shown in Figure 2, it seems that only one or two training epochs are necessary. More training epochs may lead to performance degradation. An explanation to this phenomenon is that the algorithm only pays attentions to the local differences of distribution densities, if too many training epochs are applied, the models drift from their initial positions two far away and no longer approximate the distribution densities well. To show this, Table 1 gives the average log-likelihood value in different iteration epochs. The 0 epoch indicates the log-likelihood obtained with the model solely trained with the EM algorithm. It is seen that the log-likelihood value decreases with the training epochs (large negative value), suggesting that the approximation to the density functions is less accurate.

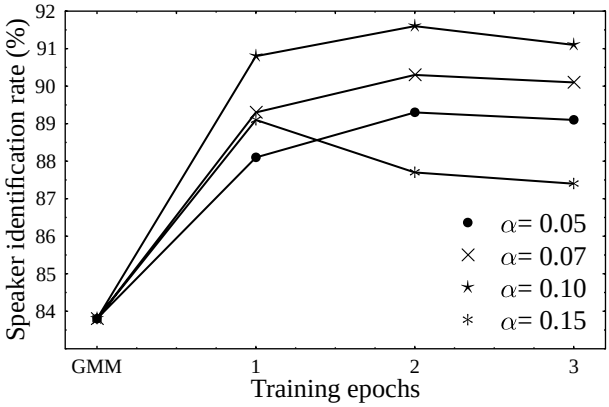

Figure 2 Speaker identification rate vs. the number of training epochs under different learning rates. The number of mixtures was 8 and the size of vector groups (N) was 4.

Let us look at the effect of learning rate. We found that a large learning rate (α>0.2) makes the algorithm unstable, in consequence, the performance is poor. It seems that α=0.1 is a proper choice.

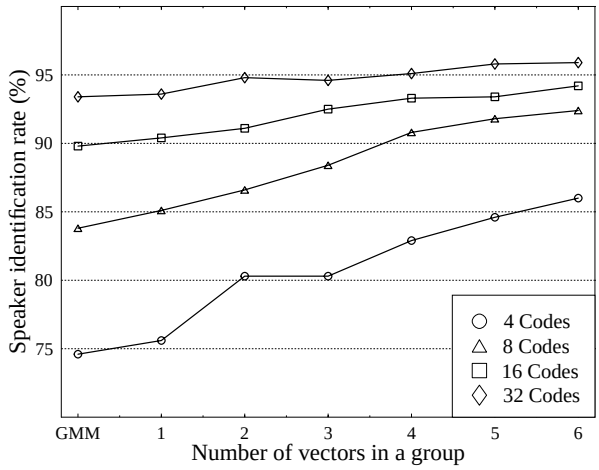

Figure 3 Speaker identification rate vs. the size of vector groups.

We fixed the learning rate at 0.1 and the number of training epoch at one, varied the size of vector groups. The evaluation results are shown in Figure 3. It is seen that, in general, the performance improves with the size

of vector groups. If a vector group contains only a single vector, the training procedure becomes to minimize the classification error for individual vectors. Due to the correlation between the vectors, a high frame level performance may not necessarily lead to a better speaker identification rate. It should be noticed that even though the speaker identification rate increases with the size of vector groups, for a fixed number of iterations, the training time increases also with the size of vector groups. We see that the gain of applying this discriminative algorithm is especially large for smaller models. For example, the identification rate with 4 mixtures increases from 74.6% to 86.0%, while the performance with 32 mixtures increases only from 93.4% to 95.9%.

We have tried two different ways of selecting vectors to compose vector groups. In another paper concerning VQ-based speaker identification [3], we evaluated the GVQ algorithm with the TIMIT database and found that the random selection method provides better performance than the sequential selection method. The reason is that the training and test sentences from the TIMIT contain totally different texts, the speaker identification is in text-independent model. In this study, we used the YOHO database. Even thought the training and test utterances are different, the vocabulary is very limited, they all are digit strings. If the model can capture some vocabulary related information from the training data, the performance for the test utterances may also improves. This guess is confirmed from the evaluation results shown in Table 2.

	Random selection	Sequential selection
GVQ	93.6%	95.0%
LGMM	94.9%	95.1%

Table 2 Comparing the performance of the VQ speaker model and the GMM trained based on the similar idea.

Here we compare the performance of the VQ codebook and the GMM. The number of code vectors and the number of mixtures are both set to 32. The VQ codebook trained with the LBG algorithm gives 90.7% identification rate, while the GMM trained with the EM algorithm is 93.4%. We noticed that even though the GMM trained with the EM algorithm outperforms the conventional VQ speaker model, the performance difference between the GVQ codebook and the LGMM is quite small. The reason is that the training procedure described here is directly borrowed from the GVQ algorithm. We simply treat the mean vectors in the GMM as a codebook, while keep the mixture weights and the covariance matrices untouched. It might be better to adjust those parameters also during the training. Unfortunately, we have not found a consistent way to modify these parameters.

4 CONCLUSION

We propose a discriminative training procedure to fine-tune the mean vectors in the GMMs after they are determined with the EM algorithm. The goal of the training is to reduce the classification error rate for vector groups. From the analysis with the two-class problem, we see that the main factor that leads to the error reduction is due to the decrease of the variance of the likelihood ratio. Evaluation experiments have been conducted to investigate the effects of various parameters in the algorithm. It is shown that the model trained with the learning rate (α) around 0.1 and the size of vector groups N=3 - 5 usually gives a good speaker identification performance.

For the YOHO database used in this paper, it is shown that the sequential selection method is slightly better than the random selection method. The reason is that the training and test utterances from the YOHO database contain only digit strings, while in another experiment [3], we have shown that the random selection method is superior to the sequential method with the TIMIT database. However, the performance difference between these two selection methods is quite small, which implies that the dynamic information plays a less important role in speaker identification than in speech recognition. The major benefit of considering several vectors at a time is that the correlation between vectors is implicitly taken into account.

5 REFERENCE

[1] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models,", IEEE Trans. on speech and audio processing, Vol. 2, pp. 72-83, 1995.

[2] H. Z. Li, J.-P. Haton, and Y. Gong, "On MMI learning of Gaussian mixture for speaker models," Proc. of European Conference on Speech Technology (EUROSPEECH), Vol. 1, pp. 363- 366, 1995, Madrid, Spain.

[3] J. He, L. Liu and G. Palm, "A new codebook training algorithm for VQ-based speaker recognition," Proc. IEEE, ICASSP'97, Vol. 2, pp. 1091-1094, Munich, GERMANY.

[4] J. Godfrey, D. Graff and A. Martin, "Public databases for speaker recognition and verification," ESCA Workshop on Automatic Speaker Recognition, Identification and Verification, pp. 39-42, 1994, Martigny, Switzerland.