Application of LDA to Speaker Recognition

Qin Jin, Alex Waibel

Interactive Systems Laboratory
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
qjin@cs.cmu.edu ahw@cs.cmu.edu

ABSTRACT

The speaker recognition task falls under the general problem of pattern classification. Speaker recognition as a pattern classification problem, its ultimate objective is design of a system that classifies the vector of features in different classes by partitioning the feature space into optimal speaker discriminative space. Linear Discriminant Analysis (LDA) is a feature extraction method that provides a linear transformation of *n*-dimensional feature vectors (or samples) into *m*-dimensional space ($m < n$), so that samples belonging to the same class are close together but samples from different classes are far apart from each other. In this paper we discuss the issue of the application of LDA to our Gaussian Mixture Model (GMM) based speaker identification task. Applying LDA improved the identification performance.

Keywords: Speaker recognition, Linear Discriminant Analysis, Gaussian Mixture Model

1. INTRODUCTION

Speaker recognition is the task of automatically recognizing who is speaking by identifying an unknown speaker among several reference speakers using speaker-specific information included in speech waves [1]. Speaker Recognition system exists anywhere when speakers are unknown and their identities are important. It makes the machine identification of participants in meetings, conferences, or conversations possible.

Speaker recognition task can be *text-independent* and *text-dependent*. By text-independent, we mean that the recognition procedure should work for any text in either training or testing. This is different from text-dependent recognition, where the text in both training and testing is the same or is known. Speaker recognition also can be classified into two further categories, *close-set* and *open-set* problems. The close-set problem is to identify a speaker from a set of *N* known speakers. While open-set problem is to decide whether the speaker of an unknown testing utterance belongs to a set of *N* speakers. There are two basic tasks in Speaker recognition: *Speaker Identification* and *Speaker Verification*. For Speaker identification the system should decide the unknown speaker's identity among *N* reference speakers while for speaker verification the system should decide whether the unknown speaker's identity is the right one as he/she claims. It is a binary decision problem (accept or reject). And Speaker verification can also be thought as a special case of the open-set problem.

Both the nonparametric and parametric models are used in speaker recognition tasks. Nearest neighbor and Vector Quantization modeling are most common nonparametric models used in speaker recognition tasks. Gaussian Mixture Model (GMM) is the representative parametric models and widely used in the speaker recognition tasks. The general structure of speaker recognition systems is described in figure 1. Speaker recognition task falls under the general problem of pattern classification.

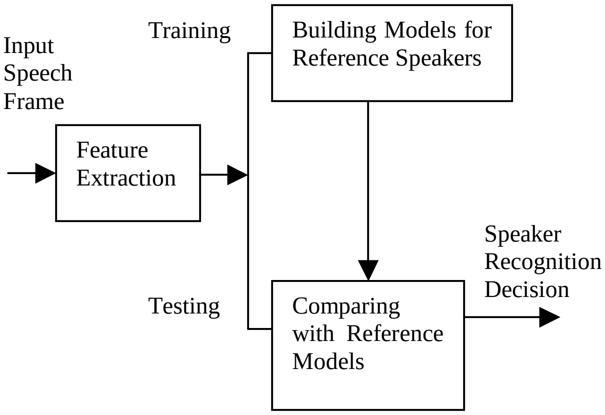

Input Speech Frame
Feature Extraction
Training
Building Models for Reference Speakers
Testing
Comparing with Reference Models
Speaker Recognition Decision

Figure 1: general structure of speaker recognition system

2. CAUSSIAN MIXTURE MODEL

2.1 Gaussian Mixture Model Basics

Mixture Models are a type of density model which comprise a number of component functions, usually Gaussians [2]. These component functions are combined to provide a multimodal density. Mixture models can provide greater flexibility and precision in modelling the underlying statistics of sample data [3][4][5]. Gaussian Mixture Models(GMM) are widely used in the speaker recognition literature [6].

We assume $\vec{z}_j$ (j=1~T) is a d dimensional feature vector and $Z = \left\{\vec{z}_{1,} \vec{z}_{2,} \ldots \vec{z}_T \right\}$ is the feature vectors set of a speaker's utterance. GMM use the following probability density function to describe the distribution of $\vec{z}_j$:

$$p\left(\vec{z}_j\middle|\theta\right)=\sum_{i=1}^{M}\left(p_i\,p\left(\vec{z}_j\middle|\theta_i\right)\right)$$

where:

M is the model order, or the number of components to be incorporated into the mixture model and θ_i is the *ith* component of the mixture model.

$$p\left(\vec{z}_j\middle|\theta_i\right)=\left(2\pi\right)^{-d/2}\left|\Sigma_i\right|^{-1/2}\exp\left(-\frac{1}{2}\left(\vec{z}_j-\vec{\mu}_i\right)^T\Sigma_i^{-1}\left(\vec{z}_j-\vec{\mu}_i\right)\right)$$

is the probability density function of a single Gaussian component. The parameter for the single component θ_i includes the mean vector $\vec{\mu}_i$ and covariance matrix Σ_i.

p_i is the weight of each single Gaussian component of the mixture model. It corresponds to the prior probability that feature vector $\vec{z}_j$ is generated by component θ_i and $\sum_{i=1}^{M} p_i = 1, p_i \geq 0$.

θ is the parameter of GMM. It includes p_i and θ_i (i=1~M).

Given Z, in which the feature vectors are assumed to be independent, our goal in the stage of building the mixture model for each speaker is to estimate the GMM parameter θ, so that we get the maximum value of $p\left(Z\middle|\theta\right)$, where $p\left(Z\middle|\theta\right)=\prod_{j=1}^{T}p\left(\vec{z}_j\middle|\theta\right)$ and $p\left(Z\middle|\theta\right)$ is differentiable with respect to θ. To estimate θ we apply *log*-function which is monotonically increasing to simplify the problem. Thus the following equation is obtained:

$$f=\nabla_\theta\left(\ln p\left(Z\middle|\theta\right)\right)=0$$

For example to estimate the mean vector of the *ith* component of GMM:

$$\begin{aligned} f &= \nabla_{\mu_i}\left(\ln p(Z\mid\theta)\right) \\ &= \sum_{j=1}^{T} p(\vec{z}_j\mid\theta)^{-1}\nabla_{\mu_i}\left(\sum_{i=1}^{M} p_i\,p(\vec{z}_j\mid\theta_i)\right) \\ &= \sum_{j=1}^{T} p(\vec{z}_j\mid\theta)^{-1} p_i\nabla_{\mu_i} p(\vec{z}_j\mid\theta_i) \\ &= \sum_{j=1}^{T} p(\vec{z}_j\mid\theta)^{-1} p_i\,p(\vec{z}_j\mid\theta_i)R_i^{-1}(\vec{z}_j-\vec{\mu}_i) \\ &= 0 \end{aligned}$$

then, we can get

$$\vec{\mu}_i=\left.\sum_{j=1}^{T}\left(P_{i,j}\vec{z}_j\right)\middle/\sum_{j=1}^{T}P_{i,j}\right.$$

where

$$P_{i,j}=p(\vec{z}_j\mid\theta)^{-1}p_i\,p(\vec{z}_j\mid\theta_i).$$

Similarly we can get the estimation formula for p_i and Σ_i:

$$\vec{\mu}_i=\left.\sum_{j=1}^{T}\left(P_{i,j}\cdot\vec{z}_j\right)\middle/\sum_{j=1}^{T}P_{i,j}\right.$$

$$\Sigma_i=\left.\sum_{j=1}^{T}\left(P_{i,j}(\vec{z}_j-\vec{\mu}_i)(\vec{z}_j-\vec{\mu}_i)^T\right)\middle/\sum_{j=1}^{T}P_{i,j}\right.$$

$$p_i=T^{-1}\cdot\sum_{j=1}^{T}P_{i,j}$$

Expectation-Maximisation (EM) [2][7] is a well established maximum likelihood algorithm for fitting a mixture model to a set of training data. We use EM algorithm to optimize the parameter estimation iteratively. It should be noted that EM requires an *a priori* selection of model order. Often a suitable number may be selected by a user, roughly corresponding to the the length of the training utterances.

2.2 Decision Rule

Given feature vectors set Y of the test utterances of an unknown speaker and GMM parameters $\left\{\theta^{S_k}\right\}_{k=1}^{N}$ of N reference speakers. θ^{S_k} is the GMM parameter of the *kth* speaker. The recognition decision should be the *jth* speaker if $p\left(\theta^{S_j}\middle|Y\right)=pr\left(\theta^{S_j}\middle|Y\right)=\underset{1\leq k\leq N}{\arg\max}\,pr\left(\theta^{S_k}\middle|Y\right)$ satisfies.

We assume equal priori probability of each speaker, for example $p\left(\theta^{S_k}\right)=\frac{1}{N}$. Based in this assumption and by using Bayes rule, we can get the following decision rule which is more convenient to apply. The recognition decision should be the *jth* speaker if $p\left(Y\middle|\theta^{S_j}\right)=\underset{1\leq k\leq N}{\arg\max}\,prY\left(Y\middle|\theta^{S_k}\right)$ satisfies.

3. APPLYING LDA TO GMM BASED SPEAKER IDENTIFICATION

3.1 Why LDA

The front-end used in many speaker recognition systems extracts, from the input signal, a set of coefficients based on a mel-cepstrum technique. In order to improve the system performance, we want to include as much speaker's characteristics as possible, such as dynamic cepstrum features (delta cepstrum, etc). However, in practice, we found that beyond a certain point, the inclusion of additional features degrades performance. Thus reducing the feature dimension has been a sensible approach towards improving the performance of a speech recognition system that uses auditory features. The speaker recognition as a pattern classification problem, its ultimate objective is design of a system that classifies the vector of features in different classes by partitioning the feature space into optimal speaker discriminative space. How to do the feature dimension reduction can get the optimal speaker discriminative space? Linear Discriminant Analysis (LDA) is a widely used technique for reducing the feature dimension.

We can also think LDA as a feature extraction method that provides a linear transformation of *n*-dimensional feature vectors (or samples) into *m*-dimensional space ($m < n$), so that samples belonging to the same class are close together but samples from different classes are far apart from each other.

3.2 Main Points of LDA

We define the *between-class* scatter matrix (or covariance matrix) is S_b and the *within* class scatter matrix is S_w for M-class pattern recognition problem. S_b indicates the deviation between the expected vectors for each pair of classes, while S_w shows the scatter of samples around the expected vector of their own class.

We try to minimize $Det(S_w)$ and maximize $Det(S_b)$ simultaneously or

$$Max\frac{|S_b|}{|S_w|} = Max|S_w^{-1}S_b|$$

A well-known class separability measure is the trace of the discrimination matrix $S_w^{-1}S_b$. We define

$$T(n) = tr(S_w^{-1}S_b)$$

Our goal is selecting a reduced number of *m* ($m<n$) features, by applying a $m \times n$ transformation matrix *A* to the original *n*-dimensional vector. We would then choose the matrix *A* such that $T(m)$ of the transformed *m*-dimension space is maximized. It can be shown that this is achieved by selecting the first *m* eigenvectors of the discrimination matrix $S_w^{-1}S_b$, whose eigenvectors $(\Phi_i, i = 1,2,\dots,n)$ are ordered by dominance of their eigenvalues

$$\lambda_1 > \lambda_2 > \cdots > \lambda_n$$

And *A* is

$$A = [\Phi_1, \Phi_2, \cdots, \Phi_m]^t$$

The new feature vector is $y = A\,x$.

The scatter matrices for the new data samples $\tilde{S}_w$ and $\tilde{S}_b$ are then both diagonal which means the coefficients are uncorrelated and they can be calculated as

$$\tilde{S}_b = AS_bA^t$$
$$\tilde{S}_w = AS_wA^t$$

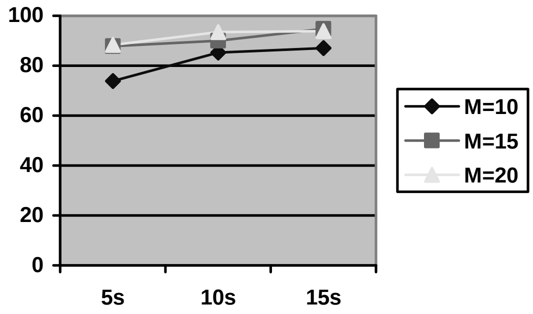
4. EXPERIMENTS RESULTS

4.1 Database

Our experiments were conducted using the speech data from the NIST 1999 Speaker Recognition Evaluation [8]. The Evaluation speech data is derived from the Switchboard-II, phase 2 corpus and consists 539 speakers (230 male, 309 female). There are two sessions for each speaker from two different telephone channels as the training data for NIST Speaker Recognition Evaluation. Each session is about one minute. We only chose the total 230 male speakers' training data as our experiment data and only tried testing on same telephone channels and sessions with training for each speaker. Applying the LDA improved the identification performance.

4.2 Results

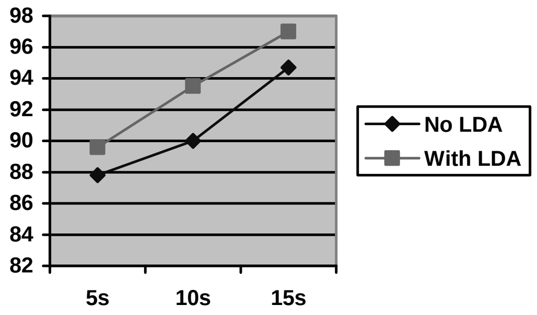
The experiments results reported in figure 2 are from same channel and session close-set speaker identification test. The length of training utterance for each speaker is 30 seconds. We compared the performance for different test utterance length and different model order.



% accuracy	*test*=5s	*test*=10s	*test*=15s
M=10	73.9	85.2	87.0
M=15	87.8	90.0	94.7
M=20	88.3	93.5	93.9

Figure 2: Identification accuracy based on GMM without application of LDA

From above results we can see when model order is 15 the identification performance is best. So only compared the identification accuracy difference with and without LDA application based on the GMM with model order of 15.



% accuracy	*test*=5s	*test*=10s	*test*=15s
No LDA	87.8	90.0	94.7
With LDA	89.6	93.5	97.0

Figure 3: Identification accuracy based on GMM with application of LDA

5. CONCLUSIONS

In this paper we first discussed the Gaussian Mixture Model approach for our close-set speaker identification system. We used EM algorithm to estimate the model parameters. Because EM requires an *a priori* selection of model order, determination of the model order is important for the system performance. Second we discussed the application of LDA to speaker identification system. It is obvious that pre-processing of a feature vector is advantageous for any type of classifier and may enhance performance. LDA is a widely used feature dimension reduction method, and as a feature extraction method it provides a linear transformation of *n*-dimensional feature vectors (or samples) into *m*-dimensional space ($m < n$), so that samples belonging to the same class are close together but samples from different classes are far apart from each other. So application of LDA improved the identification performance.

But due to the simplicity of LDA it will fail in some kinds of situations. So there are several extensions from LDA in order to solve the shortcomings of LDA. We didn’t discuss these methods in this paper. But it is sensible to try them and apply them to speaker recognition tasks.

6. REFERENCES

1. H. Gish and M.Schmit, “Text-Independent Speaker Identification”, *IEEE Signal Processing Magazine*, Oct.1994, pp 18-32.
2. R. A. Redner and H. F. Walker, “Mixture densities, maximum likelihood and the EM algorithm”, *SIAM Review*, vol. 26, no. 2, pp. 195-239, 1984
3. G.McLachlan, Mixture Models, New York:Marcel Dekker, 1988.
4. Fukunaga K., Introduction to Statistical Pattern Recognition, Academic Press, 1972 or 1990.
5. R.Schwartz, S.Roucos, M.Berouti, “The Application of Probability Density Estimation to Text-Independent Speaker Identification”, *Proc. ICASSP82*, Paris, pp.1649-1652.
6. D.A.Reynolds, R.C.Rose, “Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models”, *IEEE trans. speech and audio processing*, pp72-83.
7. A. Dempster, N. Laird, and D.Rubin, “Maximum likelihood from incomplete data via the EM algorithm”, *J. Royal Stat. Soc.*, vol.39, pp.1-38.
8. NIST 1999 Speaker Recognition Evaluation Plan, http://www.itl.nist.gov/iaui/894.01/spk99/spk99plan.html.