HIERARCHICAL SPEAKER IDENTIFICATION USING SPEAKER CLUSTERING

Bing Sun[†‡], Wenju Liu[†], Qiuhai Zhong[‡]

[†]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China
[‡]Beijing Institute of Technology, Beijing 100081, China
bsun | lwj @nlpr.ia.ac.cn, qhzhong64@bit.edu.cn

ABSTRACT

This paper explores an approach to speaker identification called speaker clustering in the GMM-based speaker recognition system in order to reduce the computational complexity. The ISODATA algorithm adapted for our purpose works well when we cluster speakers whose acoustic characteristics are similar with a distance measure. The time spent on HSI (hierarchical speaker identification) is approximately 30.3 percent than that spent on CSI (conventional speaker identification) when the number of registered speaker is 40 in our experiments. And with the increasing of the number of speakers time spent on HSI decrease comparing with CSI. It is shown that this approach can improve the speed of speaker identification system for practical purpose.

Keywords: Speaker recognition, ISODATA, Speaker clustering, Speaker identification

1. INTRODUCTION

Speaker recognition, which can be classified into identification and verification, is the process of automatically recognizing the speaker who is speaking on the basis of individual information included in speech waves. Speaker identification determines if a certain individual X is among a set of n registered speakers that he (she) is a registered speaker or that he (she) is an unregistered speaker ("Do I know you?",n+1 possible outcomes), whereas speaker verification tests the hypothesis that the speaker of a given utterance is the speaker of a given utterance ("Are you who you claim to be?", viz., a test with two possible outcomes).[1]

For proposed speaker identification, we should calculate S *a posteriori* probabilities for a given observation sequence and eventually find the speaker, which the given observation sequence belongs to, whose *a posteriori* probability is maximum where the number of speakers is S. The greater the number of S is, the more computation complex and memory space cost. Speaker clustering algorithm attempts to avoid this problem in our experiment.

Up to date, speaker clustering, as a kind of speaker adaptation methods proposed, has been applied primarily to improving *automatic speech recognition* (ASR) with some good results [2] but not used in speaker identification. There are many approaches for speaker clustering. Gender-dependent modeling is the most obviously and widely used clustering technique [3]. Gender clustering captures intra-speaker phonetic correlations largely because of the correlation between gender and vocal tract length. However, a

limitation of gender clustering is that only two (male, female) manually determined clusters are created. In order to allow an arbitrary number of clusters to be created and to allow for the use of intra-speaker correlations, ISODATA algorithm adapted was utilized.

The remainder of the paper is organized as follows. In the next section, we review GMM and how to utilize speaker clustering in our system. Section 3 introduces ISODATA algorithm adapted for our purpose. Section 4 then presents the results in our experiments. And finally, some conclusions will be given section 5.

2. SPEAKER CLUSTERING

Gaussian mixture models (GMMs), which have been successfully applied to speaker modeling in text-independent speaker identification [4], is used for modeling speaker in our system.

2.1 Gaussian Mixture Model: Review

In the Gaussian Mixture Models (GMMs), the distribution of the parameterization speech vector of a speaker is modeled by a weighted summation of Gaussian densities:

$$P(\vec{x} \mid l) = \sum_{i=1}^{N} w_i \, p_i(\vec{x}) \tag{1}$$

$$p_i(\vec{x}) = \frac{1}{(2p)^{D/2} \left| \sum_i \right|^{1/2}} \exp\left\{ -\frac{1}{2} (\vec{x} - \vec{m}_i)' \sum_i^{-1} (\vec{x} - \vec{m}_i) \right\} \tag{2}$$

where $\vec{x}$ is a D-dimensional random vector, $p_i(\vec{x})$, (i=1,..,N), is the component density characterized by the mean $\vec{m}_i$ and the covariance matrix $\sum_i$ and $l = \{w_i, \vec{m}_i, \sum_i\}$, which is estimated by an EM algorithm [5], is the speaker model.

For speaker identification, a group of S speakers {1,2,..,S} is represented by GMM's l_1, l_2,.., l_S. The objective is to find the speaker model which has the maximum *a posteriori* probability for a given speech sequence X.

$$\hat{S} = \arg\max_{1 \le k \le S} P(X \mid l_k) \tag{3}$$

In the next section, we will show how speaker clustering reduces computational complexity with the number of registered speaker increasing, comparing with the conventional speaker identification.

2.2 Classification

A speaker's acoustic characteristics can be shaped with various factors, including physical characteristics of the vocal tract and dialectic influences. Because these factors can be similar for many different speakers, it is possible to form speaker clusters within which speakers are highly similar. We make use of this method to reduce the computational complexity in the *speaker identification* (SI). When performing identification, we can then emphasize those clusters that most resemble the current speaker.

As we can see, this approach clusters the speakers whose speech features are similar and creates G clusters {1,2, ...G} whose models { $l_1^M, l_2^M, \cdots l_G^M$ } are built through training.

During the period of speaker identification, the first objective is to find the k-th cluster, including n_k speakers who is represented by GMM's l_{k1}, l_{k2},$\cdots$ l_{kn_k}, according to a given sequence X of a speaker. Just like the conventional speaker identification, the next and last objective is to find the speaker, whose model has the maximum *a posteriori* probability to the given sequence, among the n_k speaker within the verified cluster. Using this approach we only need to calculate (G+n_k) *a posteriori* probabilities instead

of N probabilities in the conventional speaker identification where N is the number of registered speakers. So the method of speaker identification based in speaker clustering, reduces the computational complexity in the period of recognition, and makes the system run faster.

Finally, we call this system as hierarchical speaker identification in that the recognition period is divided into two main steps, whose relations are hierarchic.

3. ISODATA ALGORITHM

3.1 Distance Measure

The first problem, which is to solve, is how to measure the distance between two speaker models before running speaker cluster method. That is to say, the more similar two speakers' speech feature is, the shorter the distance we define between two models is. In our system we use the distance measure borrowed from [6].

$$e = (m_1^i - m_2^j)^2 \tag{4}$$

$$d_{ij} = \frac{s_1^i}{s_2^j} + \frac{s_2^j}{s_1^i} + \frac{e}{s_2^j} + \frac{e}{s_1^i} \tag{5}$$

d_{ij} would be the distance between component i in model 1 and component j in model 2.

$$d(l_1, l_2) = \sum_{i=1}^{H} w_i^1 \min_j d_{ij} + \sum_{j=1}^{L} w_j^2 \min_i d_{ij} \tag{6}$$

where $d(l_1, l_2)$ means the distance between GMM's l_1, the number of its components being H, and l_2, the number of its components being L.

3.2 ISODATA Algorithm Adapted

We adapted ISODATA algorithm [7] for speaker clustering. The steps of algorithm run as follows:

(1) Regulating the control parameters

G is the number of cluster centers desired

$n_{\min}$ is the minimum number of speakers in a cluster

$n_{\max}$ equals the maximum number of speakers in a cluster

Assume that M is the number of clusters initialized and the model of cluster is $l_i^M, i = 1,2,...M$.

(2) According to

$$\begin{cases} d(l, l_k^M) < d(l, l_i^M) i = 1,2,...M, i \neq k \\ l \in \Gamma_k \end{cases} \tag{7}$$

N speakers are distributed into M clusters. Γ_k means the k-th cluster whose model is l_k^M .

(3) Remove the current cluster if the cluster contains Num_k speakers, with which the number is less than $n_{\min}$. Reduce M by 1. Go back to step 2 and redistribute these Num_k speakers into other clusters.

(4) Use corresponding speakers' speech data to update the cluster model l_k^M through training.

(5) Calculate the average distance of each speaker model from model of cluster which the current speaker belongs to.

$$d_k = \frac{1}{Num_k} \sum_{l \in \Gamma_k} d(l, l_k^M), k = 1,2,...M \tag{8}$$

(6) Compute the average distance of all speakers from their respective cluster.

$$\bar{d} = \frac{1}{N} \sum_{k=1}^{M} Num_k d_k \tag{9}$$

(7) Determine need to split

- If this is the last iteration, go to step 9;
- If $M < 2G$, go to step 8;
- If $M \geq 2G$, go to step 9.

(8) Split the cluster Γ_k into two clusters whose models are l_k^{M+} and l_k^{M-} respectively, when $d_k > \bar{d}$,

and $Num_k > n_{\max}$. Remove the old model l_k^M and M increase 1. The process to calculate the new two cluster models is listed as follows:

- Choose one model l arbitrarily from the cluster Γ_k and let $l_k^{M+} = \{l\}$;
- Calculate the distances between the other models of Γ_k and cluster model l_k^{M+}. Order these distances from the minimum to the maximum and find the model l' whose location is $(Num_k/2+2)$. Let $l_k^{M-} = \{l'\}$;
- Compute the distances from every speaker to two clusters and classify them into two clusters according to the shorter distances;
- Update the cluster models l_k^{M+}, l_k^{M-} through training the speaker speech data respectively.

(9) If this is last iteration, store cluster models and corresponding speakers. And then, the algorithm is over. Otherwise go to step 2 and add 1 to the number of iterations.

4. EXPERIMENTS RESULTS AND ANALYSES

The goal of this paper is to make use of speaker clustering method to a text-independent speaker identification task for improving the computational complexity comparing with conventional speaker identification. In this section, we first present the database, then introduce how to initialize the cluster models for ISODATA algorithm, and finally give the results comparison between CSI (conventional speaker identification) and HSI (hierarchical speaker identification)**.**

4.1 Database

The experiments were primarily conducted using a subset of 863 speech database of China National High Technology Project. Our database is a collection of conversations from 40 female speakers. For each speaker there are 15 conversations of approximately 4 seconds each for enrollment purposes and 18 conversations for the test (8 conversations each speaker for section 1 and the other 10 conversations for section 2)(2 conversations as a test speech). So there are 40*(4*2) =320 conversations for identification in section 1 and 40*(5*2) =400 conversations in section 2.

4.2 Initialization

In our experiments, the number of components of GMM is 64. In ISODATA algorithm, set 5 as the number G of clusters desired, the maximum number $n_{\max}$ of speakers in a cluster is 10 while the minimum number $n_{\min}$ is 5, the number of iterations allowed is 6. As we can see that the initialization is very important for ISODATA algorithm. The method used for initialization is listed as follows:

- Choose a speaker model l randomly from N speaker models and let $l_1^M = \{l\}$;
- Calculate the distances from other speaker models to cluster model l_1^M and order them from the minimum to the maximum;
- Find the model l whose location is k*[N/8]. Let $l_k^M = \{l\}, (k = 2,3,...G\}$.

After clustering has been completed and acoustic models are trained for these clusters, speaker identification is accomplished with the results which we will analyze in the next section.

4.3 Results and Analyses

Table 1 shows the results obtained from different methods during identification. As we can see from the table 1, recognition ratio of hierarchical speaker identification is a little bit lower than that of conventional speaker identification. That means our system makes mistakes for determining which cluster the testing conversation belongs to. That is to say it thinks the current data as cluster 1 but actually cluster 2. So the speaker from cluster 1, which our system

thinks the current sequence belongs to, is not the true one, who creates the sequence and belongs to cluster 2. Therefore, the errors are aroused from model clustering.

Test Set	CSI	HSI (1)	HSI (2)	HSI (3)
Section 1	98.75%	98.75%	98.75%	98.13%
Section 2	99%	98.5%	99%	98%

Table 1: Identification results

As far as we are concerned, there are two main approaches to bridge this. In the first place, we can directly find a better algorithm for speaker clustering. It is our further work. In the second place, we choose two more possible clusters in our system which the speaker, who creates the current testing speech, belongs to. And then, all the speakers who belong to the two clusters become candidates for further judgment. The column named **HSI(2)** in table 1 shows the results after we adapt our system. The recognition ratio is the same as CSI. But HSI(2) runs a bit slower than HSI(1). Even though, it is faster than CSI.

During speaker clustering, we found it takes a long time to run the ISODATA algorithm. To improve the system performance, a new method is used in our system to replace the step, spending most of the time in ISODATA algorithm, to train the cluster models using speaker speech data respectively. We regard the speaker model, from which the summation of distances to other speaker models within the cluster is the minimum, as cluster model. That is to say we use an appropriate speaker model as cluster model without training. HSI(3) illustrates its recognition ratio which is a little bit lower than that of HSI(1). However, we can efficiently solve it by training the cluster models with speech data of speakers which belong to the cluster respectively in the last iteration of the ISODATA. It is showed that each of cluster models is very robust and the speed of running speaker clustering is highly improved.

	Speakers	CSI	HSI
Time Spent 1	40	4.515	1.367
Time Spent 2	82	8.523	1.912
Time Spent 3	160	16.232	2.835

Table 2: Time spent on each identifying

As we can see from Table 2, the performance of HSI is much faster than that of CSI. The column named **Speakers** means the number of registered speakers. And the one named **CSI** represents how many seconds to be used per each identifying, and similarly in column **HSI**. Obviously the time spent on HSI is approximately 30.3 percent than that spent on CSI when the number of registered speaker is 40. And the proportion declines and is about 0.224 as the number of registered speaker is 82. In the same way, it is 0.175 when the number of speakers is 160. The results indicate that the more the number of registered speakers is, the faster the HSI system runs than CSI. So speaker clustering method great improves the speed of speaker identification while the recognition ratio does not decline obviously.

5. CONCLUTIONS AND DISCUSSIONS

In this paper, we evaluated the use of speaker clustering for text-independent speaker identification. This work primarily focuses on the task of classifying. ISODATA algorithm adapted is useful for our purpose. However, the problems with using the algorithm in clustering are the following: most of the time has been spent on training cluster models.

As we can see, the method in which the acoustic models are trained for each speaker cluster has a significant impact on both the recognition accuracy of the testing speech as well as the computation

required to perform clustering. In our experiments, we utilize one of speaker models to represent the cluster model at the start and obtain the acoustic model of clusters with training the speakers' data in the last iteration of speaker clustering algorithm. The experiments show that this method is very useful for our hierarchical speaker identification system.

6. ACKNOWLEDGEMENTS

This work is supported by grant from China National Science Foundation (No. 60172055), the "863" China National High Technology Project (No. 2001AA114181), and Beijing Government Science Foundation (No.4002012).

References

[1] Bimbot, F. and Chollet, G. Handbook of Standards and Resources for Spoken Language Systems. chapter 11. Mouton de Gruyter, Berlin.1997.

[2] Yuqing Gao, Mukund Padmanabhan and Michael Picheny, Speaker Adaptation Based on Pre-Clustering Training Speakers. EuroSpeach, Rhodes, Greece, Sep. 22-25, 1997, Vol.4, pp. 2095-2098.

[3] Ernest J. Pusateri and Timothy J. Hazen, Rapid Speaker Adaptation Using Speaker Clustering. In Proc. of ICASSP, Denver, Colorado, 2002.

[4] Douglas A. Reynolds and Richard C. Rose, Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. IEEE Transaction on Speech and Audio Processing, Vol.3, No.1, January 1995.

[5] A. Dempster, N. Laird and D. Rubin, Maximum Likelihood from Incomplete Data Via the EM Algorithm. J. Roy. Stat. Soc, Vol. 39, pp. 1-38., 1977.

[6] Homayoon S. M. Beigi, St. H. Mace and Jeffrey S. Sorensen, A Distance Measure between Collections of Distributions and Its Application to Speaker recognition. Proc. ICASSP98, Seattle, Washington, May 12-15, 1998.

[7] Zaoqi Bian and Xuegong Zhang, Pattern Recognition. Tsinghua University Press, pp. 237-239, 2000.