EFFICIENT SPEAKER IDENTIFICATION USING DISTRIBUTIONAL SPEAKER MODEL CLUSTERING

Vijendra Raj Apsingekar and Phillip L. De Leon

New Mexico State University
Klipsch School of Electrical and Computer Engineering
Las Cruces, New Mexico USA 88003
{vijendra, pdeleon}@nmsu.edu

ABSTRACT

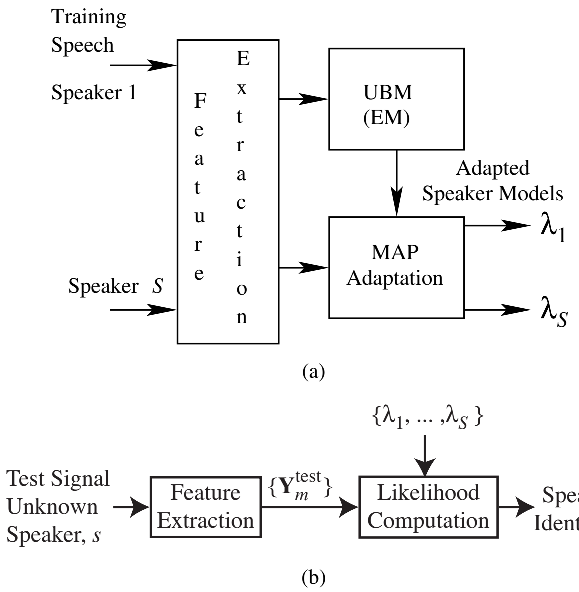
For large population speaker identification (SI) systems, likelihood computations between an unknown speaker's test feature vectors and speaker models can be very time-consuming and detrimental to applications where fast SI is required. In this paper, we propose a method whereby speaker models are clustered using a distributional distance measure such as KL divergence during the training stage. During the testing stage, only those clusters which are likely to contain high-likelihood speaker models are searched. The proposed method reduces the speaker model search space which directly results in faster SI. Any loss in identification accuracy can be controlled by trading off speed and accuracy. This paper implements GMM-UBM based SI system with MAP adapted speaker models and the results are presented on TIMIT, NTIMIT and NIST-2002 large population speech corpora.

1. INTRODUCTION

Speech is a compelling biometric as it is produced naturally and in many applications, such as in telephone transactions speech is the main modality. Also speaker recognition is an increasing area of research in the security applications. Speaker recognition is divided into speaker identification (SI) and speaker verification (SV). The objective of SI is to determine which voice sample from a set of known voice samples best matches the characteristics of an unknown input voice sample [1]. The objective of SV is to verify the identity claim [1]. SI is a two-stage procedure consisting of training and testing. In the training stage shown in Fig. 1(a), speaker-dependent feature vectors, $\mathbf{Y_m}$ are extracted from a training speech signal and a speaker model, $\boldsymbol{\lambda}_s$ is built. Of the various speaker modelling techniques, the Gaussian Mixture Model Universal Background Model (GMM-UBM) based approach and MAP adaptation of the speaker models has shown to be very successful in accurately identifying speakers from a large population and is presently state-of-the-art technique [2]. GMM-UBMs provide a probabilistic model of the distribution of feature vectors. A standard approach in estimating the parameters of the GMM-UBM (weights, mean vectors, and covariance matrices $\{w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}$) is to use the Expectation Maximization (EM) algorithm [3] and number of components in an GMM-UBM is $1024 - 2048$. Individual speaker models $\boldsymbol{\lambda}_s$ are adapted from the GMM-UBM [3]. In the testing stage shown in Fig. 1(b), features are extracted from a test signal (speaker unknown); features are compared and scored against all the S speaker models; and the most likely speaker identity, $\hat{s}$ is decided as

$$\hat{s} \quad = \quad \arg\max_{1\leq s\leq S} \sum_{m=1}^{M'} \log p(\mathbf{Y}_m^{\text{test}}|\lambda_s). \tag{1}$$

In assessing an SI system we measure the identification accuracy, computed as the number of correct identification tests divided by the total number of tests.



(a)

(b)

Fig. 1. (a) Training and (b) testing stages in SI

In this paper, we consider the problem of slow speaker *identification* for large population systems. In such SI systems (and SV systems as well), the log-likelihood computations required in (1) have been recognized as the bottleneck in terms of time complexity [2], [4]. Although accuracy is always the first consideration, fast identification is also an important factor in many applications such as speaker indexing and forensic intelligence [5], [6]. We improve upon our previously proposed speaker model clustering which was developed for GMM based SI systems to suite for GMM-UBM based SI systems [7], [8].

Among the earliest proposed methods to address the slow SI/SV problem on GMM-UBM based systems were prequantization (PQ) and speaker pruning and Gaussian pruning. In PQ, the test feature set is first compressed through subsampling (or another method) before likelihood computations [9]. PQ factors as high as 20 have been used without affecting SV accuracy. Application of PQ in order to speed-up SI has been investigated in [4] and results in a further real-time speed-up factor of as high as $5\times$ with no loss in identification accuracy using the TIMIT corpus. In speaker pruning [10], a small portion of the test feature set is compared against all speaker models. Those speaker models with the lowest scores are pruned out of the search space. In subsequent iterations, other portions of the test feature set are used and speaker models are scored and pruned until only a single speaker model remains resulting in an identification. Using the TIMIT corpus, a speed-up factor of $2\times$ has been reported with pruning [4].

Once the test signal is ready, features are extracted and first scored against UBM. Experiments conducted in [3], the authors have found that only few mixtures of a GMM-UBM contribute significantly to the likelihood score for a speech feature vector. Moreover, the adapted speaker model retain certain correspondence with the UBM, therefore likelihood score of the speaker model can be computed by scoring only the more significant mixtures. Generally, the top C mixtures are considered with $5 \leq C \leq 15$. These significant mixtures can be obtained by scoring the test feature vectors against the UBM and finding the mixtures from the UBM having the highest score [11]. Also while calculating the top C mixtures individual components of an GMM-UBM with lowest scores are pruned. After the top C scoring mixtures are obtained from the UBM, then test feature vectors are scored against these C mixtures in all the speaker models.

In [12], a hierarchical speaker identification (HSI) is proposed that uses *speaker clustering* which, for HSI purposes, refers to the task of grouping together feature sets from different speakers with similar acoustic data and modeling the superset, i.e. speaker cluster GMM. (In most other papers, speaker clustering refers to the task of grouping together unknown speech utterances based on a speaker's voice [13].) In HSI, a non-Euclidean distance measure between an individual speaker's GMM and the cluster GMMs is used to assign speakers to a cluster. Feature sets for intra-cluster speakers are then re-combined, cluster GMMs are re-built, distance measures are recalculated, and speakers are reassigned to "closer" clusters. The procedure iterates using the ISODATA algorithm until speakers have been assigned to an appropriate cluster. During the test stage, the cluster/speaker model hierarchy is utilized: first, log-likelihoods are computed against the given cluster GMMs in order to select the appropriate cluster for searching. Then log-likelihoods are computed against those speaker models in the cluster in order to identify the speaker. We note that a similar idea for reducing a search space using clusters or classes has long been used in the area of content-based image retrieval (CBIR) [14] but it appears that [12] was one of the first to use clusters for speeding up SI. Likewise, the use of speaker clusters have been used for fast speaker adaptation in speech recognition applications [15] as well as in the open-set speaker identification (OSI) problem [16].

Using a 40 speaker corpus, HSI requires only 30% of the calculation time (compared to conventional SI) while incurring an accuracy loss of less than 1% (details of the corpus and procedure for timing are not described). Unfortunately, HSI has a number of drawbacks including an extremely large amount of computation (which the authors acknowledge) required for clustering. Because of this required computation, the HSI method does not scale well with large population size. Although HSI was shown to speed up SI with little accuracy loss, the small number of speakers used in simulation does not provide any indication of how accuracy would degrade with much larger populations [17].

In a recent publication, a different approach toward efficient speaker recognition has been investigated. In [2], the authors *approximate* the required log-likelihood calculations in (1) with an approximate cross entropy (ACE) between a GMM of the test utterance and the speaker models; speed-up gains are realized through reduced computation in ACE. The authors acknowledge potential problems with constructing a GMM of the test signal and offer methods to reduce this bottleneck. Also, if the test signal is short the GMM may not be accurate. The speaker verification results presented in [2] show a theoretical speed-up factor of 5 without any degradation in false acceptance. Open-set, speaker identification results show a theoretical speed-up factor of 62 for ACE.

In our previous techniques in [7], [8] we used non-distributional distances such as Euclidean distance in our k-means algorithm for clustering. We vectorized the GMMs to conveniently use the Euclidean distance for both the clustering of speaker models and selection of clusters during the test stage. In this paper we consider the use of GMM-UBM based SI systems and use distributional distances for clustering.

This paper is organized as follows. In Section 2, we describe the distributional speaker model clustering. In Section 3, we describe the experimental evaluation and provide the results using TIMIT, NTIMIT and NIST-2002 corpora; these corpora are among the most common, large population speech databases used in the SI applications. We conclude the article in Section 4.

2. DISTRIBUTIONAL SPEAKER MODEL CLUSTERING

Speaker model clustering (SMC) was earlier introduced and successfully implemented on GMM based SI systems to speed-up the identification process during the testing in [7], [8]. Unlike the GMM-UBM systems, where the individual speaker models are adapted from a GMM-UBM, in GMM based systems speaker models are directly modeled using their training feature vectors using the EM algorithm. Thus each speaker model has M component densities (M ranging from 16 to 64) parameterized by $\{w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}$.

In [7], each speaker model is represented by a weighted mean vector (WMV) given by

$$\bar{\boldsymbol{\mu}} = \sum_{i=1}^{M} w_i \boldsymbol{\mu}_i. \qquad (2)$$

These WMVs are clustered using the k-means algorithm, Euclidean distance between the speaker models and cluster centroids is used as the distance measure in k-means, where centroid is defined as

$$\mathbf{r} = \frac{1}{K} \sum_{k=1}^{K} \bar{\boldsymbol{\mu}}_k, \qquad (3)$$

K is the number of speaker models with-in the cluster. During testing, the average of test feature vectors is calculated and nearest cluster centroid to this average (Euclidean) is identified and speaker with-in that cluster is searched. With this technique a speed-up of $2\times$ is achieved with little loss in accuracies on TIMIT and NTIMIT corpora.

In [8], instead of computing the Euclidean distance between WMV and centroid, a speaker model closest (Euclidean) to the centroid with-in a cluster is selected as the Cluster representative (CR). The log-likelihood based distance between the speaker model and CRs is used in k-means algorithm to cluster the speaker models. During testing, the test feature vectors are scored against the CRs and the cluster whos CR gives the maximum score is searched. We are able to speed-up the SI accuracy by a factor of $4\times$ with little or no loss in accuracies on TIMIT, NTIMIT and NIST-2002 corpora.

As the proposed speaker model clustering is a training stage clustering after the speaker models are built, it can be combined with test stage speed-up techniques such as prequantization (PQ) and pruning. When PQ and pruning are applied at test stage along with SMC we are able to achieve a speed-up of over $74\times$ with little or no loss in accuracies on all the three speech corpora.

2.1. Clustering using Distributional SMC

In this paper the speaker models are MAP adapted from the GMM-UBM and typically the number of components in an GMM-UBM systems are 1024-2048. With such high number of component densities the WMV of each adapted speaker model using (2) results in vectors which are very compact in the hyperspace. Such vectors in the k-means clustering results in clusters which are not improving the speed-ups factors.

Thus a new method of clustering has to be developed which at any step does not require the vectorization of the speaker models. Thus clustering based on purely distributional distances, such as Kullback-Leibler distance need to be developed. Here we propose distributional speaker model clustering as the distribution of speaker models is considered rather than vectorizing the speaker model.

Here we still use k-means algorithm to cluster and the distance measure used is an approximation of the KL-divergence given by [18],

$$\begin{aligned} d(\lambda_s, \lambda_n^{\mathrm{CR}}) \;\approx\; & \frac{1}{M} \sum_{m=1}^{M} \log p(\mathbf{Y}_{s,m}^{\mathrm{train}} | \lambda_s) - \\ & \frac{1}{M} \sum_{m=1}^{M} \log p(\mathbf{Y}_{s,m}^{\mathrm{train}} | \lambda_n^{\mathrm{CR}}), \qquad (4) \end{aligned}$$

where $\boldsymbol{\lambda}_{CR}$ is the CR of cluster n. The CR is chosen as

$$\boldsymbol{\lambda}_n^{\mathrm{CR}} = \arg \max_{1 \leq j \leq K} \sum_{i=1, i \neq j}^{K} \log p(\mathbf{Y}_i^{\mathrm{train}} | \boldsymbol{\lambda}_j), \qquad (5)$$

where K is the number of speaker models with in a cluster n and N is the total number of clusters. Thus to select a CR, all the speakers' training feature vectors with-in a cluster are scored against a speaker model. This score is calculated for all the speaker in that cluster and the maximum scoring speaker is selected as the CR. In calculating the above scores the training feature vectors of a particular speaker are not scored against their own model. Thus the CR is selected based on the log-likelihood measure rather than the Euclidean distance between the centroid and the WMV as in [8]. Algorithm 1 describes the speaker model clustering using above technique.

Algorithm 1 KL GMM-based Speaker model clustering

```
1: Initialize cluster centers $\lambda_n^{\mathrm{CR}}$, $1 \leq n \leq N$ using randomly-chosen speaker models
2: Compute distance using (4) from $\lambda_s$ to $\lambda_n^{\mathrm{CR}}$, $1 \leq s \leq S$
3: Assign each $\lambda_s$ to the cluster with the minimum distance
4: Compute new cluster representatives $\lambda_n^{\mathrm{CR}}$ using (5)
5: Goto step 2 and terminate when cluster membership does not change
```

2.2. Test Stage

During the test stage as shown in Fig. 2, once the test feature vectors are acquired, the log-likelihood score is computed against all the CRs and clusters with maximum scoring representative are searched given as

$$C_n = \arg \max_{1 \le n \le N} \left[\sum_{m=1}^{M'} \log p(\mathbf{Y}_{s,m}^{\text{test}} | \lambda_n^{\text{CR}}) \right]. \quad (6)$$

Rather than selecting a single cluster to search using criteria in (6) we can also use a subset of clusters ranked according to these equations. Using a subset of clusters allows a smooth trade-off between accuracy loss (due to searching too few clusters) and speed.

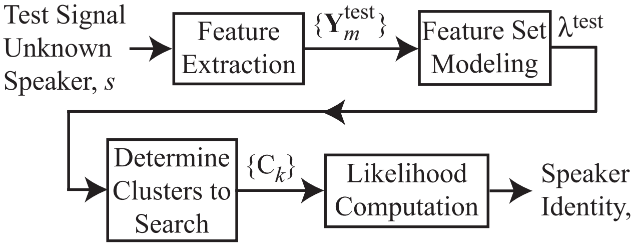


Fig. 2. Testing stage in SMC

3. EXPERIMENTS AND RESULTS

Experiments have been performed on the TIMIT, NTIMIT and NIST 2002 corpora. To demonstrate the applicability of the methods proposed in Section 2 to a wide variety of GMM-UBM systems, we have added some additional elements such as delta MFCCs, cepstral mean subtraction (CMS) and RASTA processing depending on the corpus being used. Specifically, our baseline system uses an energy-based voice activity detector to remove silence; feature vectors composed of 29 MFCCs for TIMIT, 20 MFCCs for NTIMIT and 13 MFCCs + 13 delta MFCCs for NIST 2002 extracted every 10 ms using a 25 ms hamming window; CMS and RASTA processing are applied to NIST 2002. A 1024 component density UBM is built for each corpus by concatenating the training feature vectors of all the speakers within that corpus. Individual speaker models have then been built by MAP adaptation of parameters of the mean alone with a relevance factor of 16. For TIMIT/NTIMIT, we use approximately 24s training signals and 6s test signals and for NIST 2002 (one speaker detection cellular task) we use approximately 90s training signals and 30s test signals. Our baseline SI accuracies are 100%, 73.33% and 96.67% on TIMIT, NTIMIT and NIST-2002 respectively.

We partitioned the speaker model space into N clusters using a range of values for N and measured SI accuracy rates. We found $N = 100$ to give good performance with the TIMIT/NTIMIT corpora and $N = 50$ with the NIST 2002 corpus. In order to evaluate the proposed approach, we measure SI accuracy as a function of the percentage of clusters searched as shown in Fig. 3. This percentage is an approximation to the search space reduction in (1), since the number of speaker models in each cluster are not exactly the same but are more or less equally-distributed. Using our approach, we are able to search as few as 10% of the clusters and incur a 2.7%, 2.0%, and 1.3% loss in SI accuracy with the TIMIT, NTIMIT, and NIST 2002 corpora respectively; searching 20% of the clusters resulted in accuracy loss of 1.68%, 0.5% and 0% respectively (shown in Table 1). Loss in accuracy associated by searching 20% of the clusters is very insignificant. Searching 20% of the clusters the speed-up factor achieved is approximately 5×.

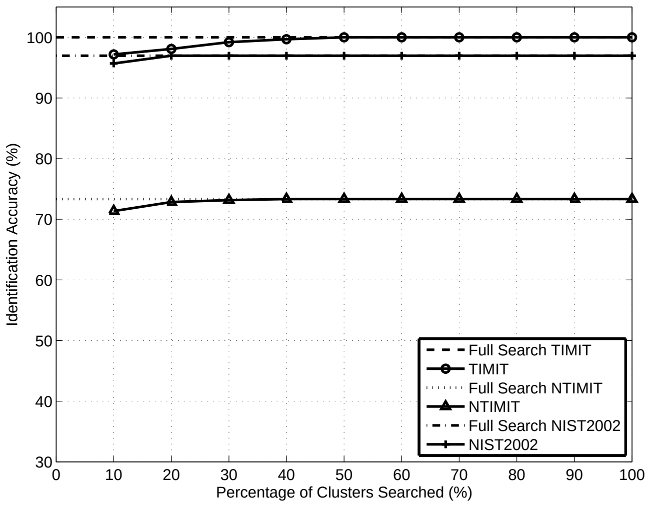


Fig. 3. Results using SMC

4. CONCLUSIONS

In SI, log-likelihood calculations in the test stage have been recognized as the bottleneck in terms of time complexity. In this paper, we have improved upon our earlier work which utilizes distributional speaker model clustering for reducing the number of speaker models that have to be scored against, thus enabling faster and efficient SI. In particular we have introduced distributional distances such as log-likelihood based selection of cluster representatives and KL distance based clustering. for the TIMIT, NTIMIT and NIST-2002 corpora, we are able to search as few as 20% of the clusters and loss in SI accuracy associated at this percent of clusters is very insignificant. We are able to achieve a speed-up factor of 5× with the proposed method. In addition, we could also use the test stage speed-up techniques such as pre-quantization, speaker

Table 1. Accuracies using the distributional speaker model clustering.

Speech Corpora	10% of Clusters Searched	20% of Clusters Searched	100% of Clusters Searched
TIMIT	97.30%	98.32%	100%
NTIMIT	71.35%	72.83%	73.33%
NIST-2002	95.69%	96.97%	96.97%

pruning and Gaussian pruning along with the clustering for even greater speed-ups.

5. REFERENCES

[1] D. Reynolds and R. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Trans. Signal Processing*, vol. 3, no. 1, pp. 72–83, Jan. 1995.

[2] H. Aronowitz and D. Burshtein, "Efficient speaker recognition using approximated cross entropy (ACE)," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 15, no. 7, pp. 2033–2043, Sep. 2007.

[3] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[4] T. Kinnunen, E. Karpov, and P. Franti, "Real-time speaker identification and verification," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 14, no. 1, pp. 277–288, Jan. 2006.

[5] J. Makhoul, F. Kubala, T. Leek, L. Daben, N. Long, R. Schwartz, and A. Srivastava, "Speech and language technologies for audio indexing and retrieval," *Proc. IEEE*, vol. 88, no. 8, pp. 1138–1353, Aug 2000.

[6] H. Aronowitz, D. Burshtein, and A. Amir, "Speaker indexing in audio archives using test utterance gaussian mixture modeling," in *Proc. Int. Conf. Spoken Lang. Proc. (ICSLP)*, 2004.

[7] P. L. De Leon and V. R. Apsingekar, "Reducing speaker model search space in speaker identification," in *Proc. IEEE Biometrics Symposium*, 2007.

[8] V. R. Apsingekar and P. DeLeon, "Efficient speaker identification using speaker model clustering," *Proc. European Signal Processing Conf. (EUSIPCO)*, 2008.

[9] J. McLaughlin, D. A. Reynolds, and T. Gleeson, "A study of computation speed-ups of the GMM-UBM speaker recognition system," in *Proc. 6th European Conf. Speech Communication and Technology (Eurospeech)*, 1999, pp. 1215–1218.

[10] B. L. Pellom and J. H. L. Hansen, "An efficient scoring algorithm for gaussian mixture model based speaker identification," *IEEE Signal Processing Lett.*, vol. 5, no. 11, pp. 281–284, Nov. 1998.

[11] R. Zheng, S. Zhang, and B. Xu, "Text-independent speaker identification using gmm-ubm and frame level likelihood normalization," *International Symposium on Chinese Spoken Language Processing*, pp. 289–292, Dec. 2004.

[12] B. Sun, W. Liu, and Q. Zhong, "Hierarchical speaker identification using speaker clustering," in *Proc. Int. Conf. Natural Language Processing and Knowledge Engineering*, 2003.

[13] W. Tsai, S. Cheng, and H. Wang, "Automatic speaker clustering using a voice characteristic reference space and maximum purity estimation," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 15, no. 4, pp. 1461–1474, May 2007.

[14] A. M. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[15] L. J. Rodriguez and M. I. Torres, "A speaker clustering algorithm for fast speaker adaptation in continuous speech recognition," *Lecture Notes in Computer Science: Text, Speech and Dialogue*, vol. 3206/2004, 2004, springer.

[16] P. Angkititrakul and J. Hansen, "Discriminative in-set/out-of-set speaker recongition," *IEEE Trans. Audio, Speech, and Language Process.*, vol. 15, no. 2, pp. 498–508, Feb. 2007.

[17] D. Reynolds, "Large population speaker identification using clean and telephone speech," *IEEE Signal Processing Lett.*, vol. 2, no. 3, pp. 46–48, Mar. 1995.

[18] J. Goldberger and H. Aronowitz, "A distance measure between GMMs based on the unscented transform and its application to speaker recognition," in *Proc. of Interspeech*, 2005, pp. 1985–1988.