A Study on Universal Background Model Training in Speaker Verification

Article *in* IEEE Transactions on Audio Speech and Language Processing · October 2011

DOI: 10.1109/TASL.2010.2102753 · Source: IEEE Xplore

CITATIONS
55

READS
909

2 authors:

Taufiq Hasan
Bangladesh University of Engineering and Technology
36 PUBLICATIONS **664** CITATIONS
SEE PROFILE

John H. L. Hansen
University of Texas at Dallas
507 PUBLICATIONS **8,259** CITATIONS
SEE PROFILE

A study on Universal Background Model training in Speaker Verification

Taufiq Hasan *Student Member, IEEE* and John H. L. Hansen,*† *Fellow IEEE*

Abstract

State-of-the-art Gaussian Mixture Model (GMM) based speaker recognition/verification systems utilize a universal background model (UBM), which typically requires extensive resources, especially if multiple channel and microphone categories are considered. In this paper, we systematically analyze speaker verification system performance when the UBM data is selected and purposefully altered in different ways, including variation of the amount of data, sub-sampling structure of the feature frames, and variation in the number of speakers. An objective measure is formulated from the UBM covariance matrix which is found to be highly correlated with system performance when the data amount was varied while keeping the UBM data set constant, and increasing the number of UBM speakers while keeping the data amount constant. The advantages of feature sub-sampling for improving UBM training speed is also discussed, and a novel and effective phonetic distance based frame selection method is developed. The sub-sampling methods presented are shown to retain baseline EER system performance using only 1% of the original UBM data, resulting in a drastic reduction in UBM training computation time. This, in theory, dispels the myth of "There's no data like more data" for the purpose of UBM construction. With respect to the UBM speakers, the effect of systematically controlling the number of training (UBM) speakers versus overall system performance is analyzed. It is shown experimentally that increasing the inter-speaker variability in the UBM data while maintaining the overall total data size constant gradually improves system performance. Finally, two alternative speaker selection methods based on different speaker diversity measures are presented. Using the proposed schemes, it is shown that by selecting a diverse set of UBM speakers, the baseline system performance can be retained using less than 30% of the original UBM speakers.

Index Terms: Speaker verification, universal background model, intelligent speaker selection.

Mail All Correspondence To:

Prof. John H.L. Hansen
Center for Robust Speech Systems (CRSS), Erik Jonsson School of Engineering and Computer Science, Dept. of Electrical Engineering, University of Texas at Dallas
2601 N. Floyd Road, EC33, Richardson, TX 75080-1407, U.S.A

*This project was funded by AFRL through a subcontract to RADC Inc. under FA8750-09-C-0067, and partially by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering held by J. Hansen.

†The authors note that a probe investigation on data selection for UBM training was presented in [1], with some of these findings motivating further investigation in Sec. IV.

I. INTRODUCTION

In recent years, Gaussian mixture model (GMM) based approaches in text independent speaker identification systems have received considerable attention. Although, many distinct algorithms have been developed in this area, the use of GMMs for modeling acoustic features have become almost exclusive. The most fundamental GMM based speaker recognition methods include the classical maximum a-posteriori (MAP) adaptation of UBM parameters [2] (GMM-UBM) and support vector machine (SVM) modeling of GMM super-vectors (GMM-SVM) [3]. Both of these approaches improve when using additional normalization schemes, such as factor analysis [4], eigenvoice [5], or nuisance attribute projection (NAP) [6]. Many groups employ these schemes successfully as individual subsystems in the recent 2008 National Institute of Science and Technology (NIST) speaker recognition (SRE) evaluations [7]–[10].

An important mutual element of these sub-systems is the UBM. It is essentially a very large GMM trained to represent the speaker independent distribution of the speech features [2] for all speakers in general, and is employed as the expected alternative speaker model during the verification task. It is also employed in open-set speaker recognition systems as well. In the two primary GMM based systems (GMM-UBM, GMM-SVM), all speaker models are dependent on the UBM, making it a key element. However, despite its importance, focused research on UBM training has not yet been conducted in the literature [2]. The general trend is to use as many speakers and speech comprising a wide range of speech/channel conditions, without much thought regarding performance tradeoff. Though, recent research has focused on other aspects of the UBM, such as [11] where an adaptive individual background model training is considered, or [12], where the application of speaker normalization techniques on the UBM data is investigated, the basic and fundamental questions regarding the construction of a UBM and its implications in system performance is still an open unaddressed question. In this paper, we give an in depth consideration of the UBM training process and attempt to gain insight on how system performance is related to specific UBM composition.

There are a number of different parameters involved in the UBM training process. It is possible to classify these parameters into two broad categories as, a) algorithm parameters and b) data parameters. The algorithm parameters are the variations in the training process which include the number of mixtures, method of training, number of iterations, method of initialization, etc. The data parameters include different ways of defining the subset of

available training data. These parameters consider the corpus, the amount of data, number of speakers in the data, amount of data per speaker, method of selecting speakers, ways of using the feature vectors, data balancing according to channel, microphone, language, or other variability, and so on. Since the only available objective criteria that can measure the quality of a UBM is overall/final system performance, finding a better UBM becomes a challenge since it will generally rely on a trial and error based endeavor, making it very impractical to vary all the mentioned parameters and find the optimal combination that gives the best performance. Thus, in order to limit the scope of this research, we only focus on a limited set of the data parameters, and attempt to analyze their effect on system performance in order to answer some fundamental questions concerning UBM training.

The first question that arises concerning the UBM data is the required amount. A common assumption in UBM training is that the more data used, the better the system performance. UBMs with 512, 1024, 2048 or more mixtures are sought after, with the assumption that they represent the definitive world speaker acoustic space. Research groups involved in the NIST SRE typically use 5 min utterances from all NIST 2004-2005 data along with the Switchboard Cellular I and II data [10]. However, there is no concrete evidence that using the maximum amount of data would guarantee the best overall performance. According to [2], as long as the development speaker population is kept the same, a small amount of data is sufficient for reasonable system performance. This suggests that, the degree of inter-speaker variation in the data is more important than the absolute amount of data per speaker. In this paper, we systematically analyze this aspect of UBM training, determine a measure of data variability from the UBM parameters, and relate this to system performance.

The issue of using a subset of features from a given UBM utterance arises as an inevitable consequence of using a reduced amount of data. The simplest methods for sub-sampling feature frames would include decimation and random feature selection, which have already been utilized to improve the CPU training time of a GMM [13]. Clearly, these feature sub-sampling methods do not consider the actual acoustic content of the features, and select features rather blindly. Here, we consider an adaptive phoneme dependent feature sub-sampling scheme for effectively capturing the subtle nuances of features in each utterance using a very small amount of data. The goal is to maximize the data variability that can be captured from a given UBM utterance using a minimal number of features. This is motivated by considering the feature selection issue at the phoneme level. Since inter-speaker variability in the UBM data has a higher contribution to system performance [2], intra-speaker phoneme

variation should be less relevant for the UBM. When a long duration utterance is used for a speaker, some phones will occur more frequently and with greater duration, and therefore would contribute to probability density function (PDF) components in the UBM that represent the intra-speaker distribution of that phoneme, causing an imbalance. Thus, reducing the development data by means of proper selection of the training feature vectors will obviously improve computation speed, with a possible improvement in overall system performance as well.

The inter-speaker variability of the UBM data is directly related to the number of speakers present in the data-set. In our study, we systematically control the number of speakers in the UBM data in an attempt to gain valuable insight on an overall system performance relation with inter-speaker variability. Considering the idea that speaker diversity is a key factor for UBM data selection, approaches of diverse speaker selection is also investigated.

This paper is organized as follows: in Section II we discuss the definition of the UBM and how data parameters should be set for an ideal UBM. Section III describes our baseline system. In Section IV, we analyze the effect of changing the amount of data in the UBM and in Section V, we consider feature sub-sampling approaches. Next, in Section VI we analyze the effect of changing the number of unique speakers in the UBM data and in Section VII we discuss different ways of sub-sampling speakers. Finally, in Section VIII we draw conclusions. It should be noted that UBM training is analogous to baking a cake, there are many ingredients that can be adjusted, but the final outcome will still have the same basic structure. It is expected that improved UBM construction may not have a significant impact in performance metrics such as EER or minDCF. However, from a scientific perspective, formulating a more effective UBM using less training data should reduce computational requirements and produce a more balanced UBM that represents the overall speaker acoustic space.

II. The ideal UBM

As noted earlier, the UBM is a speaker-independent Gaussian Mixture Model (GMM) trained with speech acoustic features from a large set of speakers to represent the general, speaker independent distribution of features. The concept of a UBM becomes important because of the likelihood ratio test performed in the verification task. Given an observation O, and a hypothesized speaker S, the task of speaker verification can be stated as a hypothesis

test between:

$$\begin{aligned} H_0 &: O \text{ is from speaker } S, \\ H_1 &: O \text{ is not from speaker } S. \end{aligned} \tag{1}$$

In general, the hypothesis H_0 and H_1 are represented by a speaker dependent model λ_S and the background model $\lambda_{\bar{S}}$. Thus, for the observed feature vectors $\mathbf{X}$, the likelihood ratio test is performed by evaluating:

$$\Lambda(\mathbf{X}) = \frac{p(\mathbf{X}|\lambda_S)}{p(\mathbf{X}|\lambda_{\bar{S}})} \begin{cases} \geq \theta \text{ accept } H_0 \\ < \theta \text{ reject } H_0 \end{cases} \tag{2}$$

Thus, ideally, the background model $\lambda_{\bar{S}}$ should be a model that represents the entire space of all possible alternatives to the hypothesized speaker S, which leads to a speaker specific background model. This approach has been adopted by many researchers in the past [11], [14]–[16]. However, creating a speaker specific background model for each enrolled speaker can be computationally expensive, especially for a large number of speakers, which is typically the case in the NIST SRE evaluations [7]. Also, it may not always be necessary to represent all outside speakers, only those that may attempt to enter the speaker verification system as imposters. Thus, most modern speaker verification systems use a single speaker independent background model, (i.e. the UBM) for modeling the alternative hypothesis in the likelihood ratio test. Generally, the UBM is trained using a large amount of data coming from a variety of different speakers and channel/microphone conditions, so that the model contains at least some aspect of the variabilities that could be encountered in the unknown test data.

A. Data Balancing

Inspired by the general guideline of UBM data as presented in [2], the data requirements for an ideal UBM can be specified as follows. Assume that there is only transmission channel and microphone variability present in the available development data. Let the following variables be,

$$S = \{s_i\},\ 1 \leq i \leq N_s, \tag{3}$$

$$M = \{m_i\},\ 1 \leq i \leq N_m, \text{ and} \tag{4}$$

$$C = \{c_i\},\ 1 \leq i \leq N_c \tag{5}$$

denote the set of all available speakers, microphone types, and transmission channel types, respectively, in a single gender database. Here N_s, N_m, and N_c denote the number of speakers, microphones and transmission channels, respectively. Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2 ... \mathbf{x}_N\}$ denote the set of all available features in the database. Next, define the following data sets:

$$\begin{aligned} \mathbf{X}_{s_i} &= \{\mathbf{x}_j | \mathbf{x}_j \text{ belongs to speaker } s_i\}, \\ \mathbf{X}_{m_i} &= \{\mathbf{x}_j | \mathbf{x}_j \text{ was recorded with mic. } m_i\}, \\ \mathbf{X}_{c_i} &= \{\mathbf{x}_j | \mathbf{x}_j \text{ comes from channel } c_i\}. \end{aligned} \tag{6}$$

Obviously, considering the total number of speakers, microphones, and channels, the union of each reflects the total data available.

$$\mathbf{X} = \left(\cup_{i=1}^{N_s} \mathbf{X}_{s_i}\right) = \left(\cup_{i=1}^{N_m} \mathbf{X}_{m_i}\right) = \left(\cup_{i=1}^{N_c} \mathbf{X}_{c_i}\right). \tag{7}$$

If $\mathbf{X}_{\text{I}} \subset \mathbf{X}$ denote the set of features that should be used for the ideal UBM, it should contain features from all these variabilities in equal proportions with respect to the test data. If, the prior probabilities of the occurrence of a speaker, microphone type, or channel condition in the test data is known, the feature set $\mathbf{X}_{\text{I}}$ should fulfill the following constraints:

$$n(\mathbf{X}_{s_i} \cap \mathbf{X}_{\text{I}}) = \mathcal{R}\left(p(s_i) n(\mathbf{X}_{\text{I}})\right) \quad \forall s_i \in S, \tag{8}$$

$$n(\mathbf{X}_{m_i} \cap \mathbf{X}_{\text{I}}) = \mathcal{R}\left(p(m_i) n(\mathbf{X}_{\text{I}})\right) \quad \forall m_i \in M, \tag{9}$$

$$n(\mathbf{X}_{c_i} \cap \mathbf{X}_{\text{I}}) = \mathcal{R}\left(p(c_i) n(\mathbf{X}_{\text{I}})\right) \quad \forall c_i \in C, \tag{10}$$

where $n(\cdot)$, $\mathcal{R}(\cdot)$ and $p(\cdot)$ indicate the number of elements in a set, the round-off operation and the prior probability of an element in the test data, respectively. Typically, there is no prior knowledge about the test data condition distribution, leaving the system designer with the only option of assuming these prior probabilities are equal. This is known as balancing of the UBM data as discussed in [2]. It should be noted that (8) assumes all the speakers to be considerably diverse in nature otherwise similar speakers (i.e. the cohorts) may introduce unbalance in the data. This issue is further discussed in Section VII.

B. Data Amount

It is clear that the set $\mathbf{X}$ of all available data features should be large enough to represent all variabilities faithfully. For example, if there is only 5 min of data for a cordless phone

in the entire 10 hour UBM data set, trying to balance the microphone type would require us to use only 5 min of data from each microphone type, which may lead to insufficient amounts of UBM data. However, what defines a data amount to be sufficient or insufficient is rather vague and probably system dependent. Generally, for a given data set, the amount of data that appropriately represents the variability of the entire corpus should be sufficient for the UBM. Since human speech features can only occupy a limited region in the feature space due to physiology constraints, it is expected that the variability of the data would be saturated when the data amount becomes very large, assuming other conditions (e.g. channel, microphone, or language variability) are kept the same. Let $\vartheta(\cdot)$ represent a function that can measure the variability of the UBM data, then as the size of the set $\mathbf{X_I}$ increases, $\vartheta(\mathbf{X_I})$ should approach some constant value. Mathematically, this can be written as follows,

$$\lim_{n(\mathbf{X_I})\to\infty} \vartheta(\mathbf{X_I}) = \sigma. \tag{11}$$

Having defined the scope of the data characteristics, it is now possible to consider alternative UBM training schemes. A baseline system scenario is first considered in the next section.

III. Baseline System Description

A. System Overview

Since the objective of this research is focused only on UBM training, a fairly standard GMM-UBM [2] baseline system is employed without any mismatch compensation and score normalization. A prime reason for not using expanded or enhanced GMM based systems (i.e., joint factor analysis (JFA) or Eigenchannel [4]), is because they require time consuming training of the Eigenchannel and Eigenvoince matrices each time the UBM is retrained. Given the number and extent of the experiments required in this study, it was decided that using such enhancements would be impractical. Future studies could further explore the impact in UBM construction with other system processing tasks.

For the front-end, 39-dimensional MFCC features (MFCC+Δ+$\Delta\Delta$) are extracted using a 25 ms analysis window with 10 ms shift. Next, feature warping [17] based on applying a 3-s sliding window, is performed. To remove silence frames, a phone recognizer based voice activity detector (VAD) is utilized. For baseline UBM training, 1024 mixtures are used. UBM training is performed using the maximum likelihood (ML) criterion with HTK [18] tools, performing 15 iterations per mixture split. Here, only male speaker trials are considered. For modeling, gender dependent UBMs are adapted to each enrollment speaker dependent

model using classical MAP adaptation [2] with one iteration and a relevance factor of 19. In scoring, a standard 20-best expected log likelihood ratio scoring is employed. The 5 min tel-tel condition trials [7] of the NIST 2008 SRE are used for all evaluations.

B. UBM Database

Here two different sets of database are employed for the UBM: 1) 2019 male utterances from the NIST SRE 2004 1-s (5 min training) data, consisting of 126 unique speakers, 2) 5685 male utterances from NIST SRE 2004 and NIST SRE 2006 (that include channel and microphone labels), consisting of 392 unique speakers. In these two cases, the resulting baseline equal error rate (EER) was 11.43% and 11.41%, respectively.

C. Computational resources

The speaker ID system was implemented on a high-performance Rocks computing cluster running the CentOS Linux distribution. The cluster comprises 18 HP Intel Quad-Core Xeon 2.33 GHz CPU's, yielding 72 CPU cores. A total of 126 GB RAM is available internally on the system. A 4 TB external RAID disk array is attached to the cluster by means of a storage area network (SAN). The array is connected with the cluster nodes through a 1 Gbit Ethernet switch. For calculation of CPU times for UBM training, the time required for each parallel process in the 72 CPUs were separately calculated and accumulated. Thus, the CPU times reported in this work represent a close estimation of the training time that would be required if only a single CPU was available.

IV. UBM DATA : WHAT IS A SUFFICIENT AMOUNT

As discussed in Section II, the UBM data should contain enough variability in a sufficient amount to represent each of the different channel and microphone conditions. However, using an enormously large database may not be necessary for the best performance, as long as the required variability is maintained. In this section, system performance variation is examined by increasing the total amount of data for UBM training, while keeping the speaker population the same.

An experiment was performed using the UBM database-1 as described in Section III. It should be noted that in this UBM data set, a single speaker may occur multiple times in different channel/mic conditions. This database is used mainly because of the following reasons:

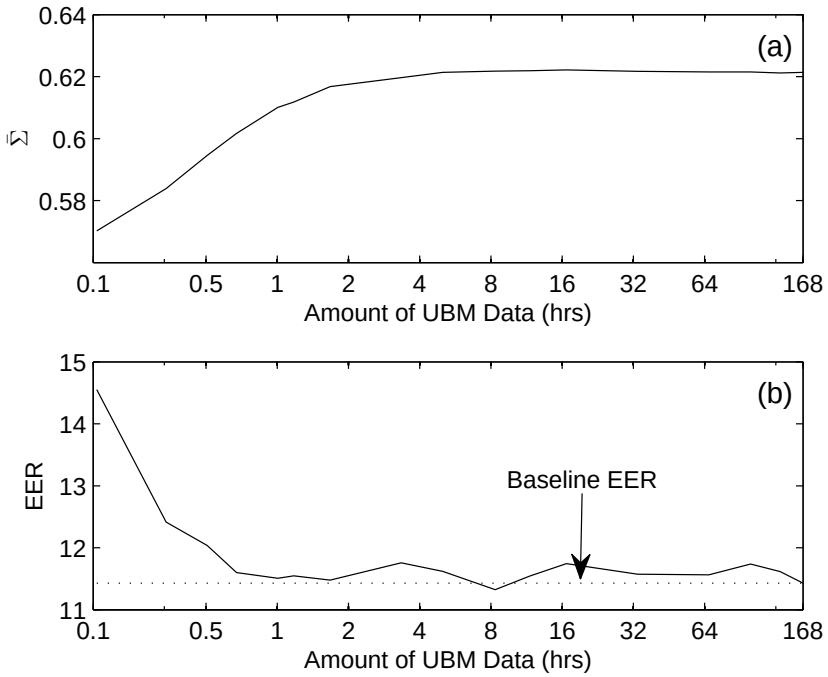

Fig. 1. (a) Variation of UBM average weighted variance ($\bar{\Sigma}$) with total amount of UBM data (hrs), (b) variation of system EER with total amount of UBM data (hrs). Feature frames are selected uniformly from each utterance.

- The NIST SRE 04 1-s corpus contains sufficient variability;
- Other research groups have shown success in UBM training using this set alone ([8], [9], [19]);
- The actual number of speakers in this experiment is not a primary concern;
- Each utterance is 5 min in duration, making it convenient to extract equal amounts of data from each utterance uniformly.

The total duration of the data is 168.25 hours. Next the UBM is trained using only the first n feature frames from each utterance, and evaluated the GMM-UBM system (described in Sec III) for the male trials only. For different values of n, the equivalent total data amount in hours, the equal error rate (EER) and CPU times required for training are obtained.

A simple formula is proposed to measure the variability of the data. Since a variance measure of a 39 dimensional feature vector may not provide an accurate measure of variability of the data, an analysis of the UBM covariance matrix is performed. Here, a parameter, average weighted variance (AWV) is defined which is computed from the UBM diagonal

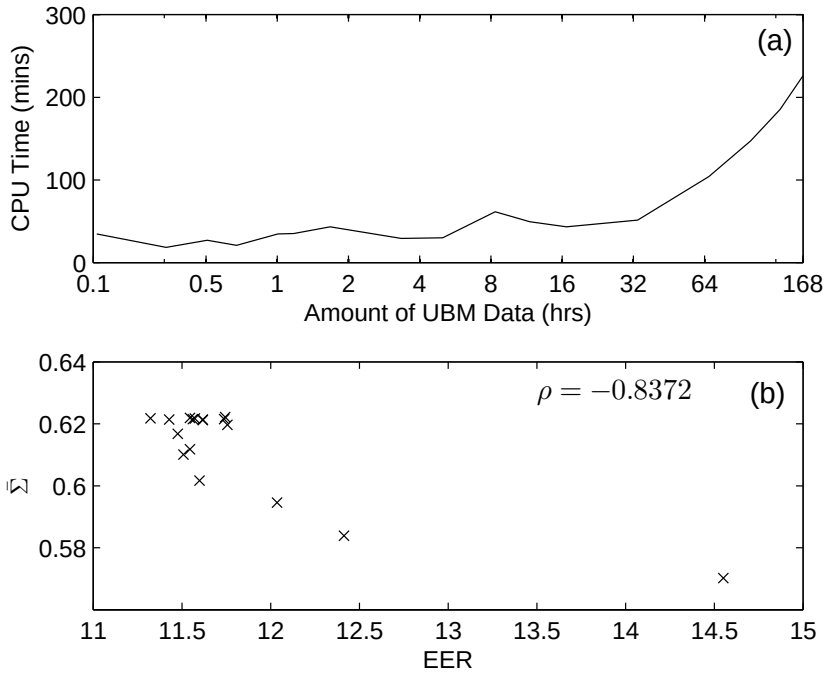

Fig. 2. (a) UBM training CPU time variation with changing amount of UBM data (hrs). (b) Scatter plot showing the correlation between EER and average weighted variance ($\bar{\Sigma}$). All data points from Fig. 1(a) and (b) are used to generate this scatter plot.

covariance matrices as follows. Let the UBM be expressed as,

$$f(\mathbf{x}|\lambda_{\text{UBM}}) = \sum_{i=1}^{M} w_i \frac{1}{(2\pi)^{K/2}|\mathbf{\Sigma_i}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_i)^T \mathbf{\Sigma_i}^{-1}(\mathbf{x}-\mu_i)} \tag{12}$$

where w_i, μ_i, $\mathbf{\Sigma}_i$, K and M denote the weights, mean vectors, covariance matrices, feature dimension and number of Gaussian mixtures, respectively. Assuming a diagonal covariance matrix $\mathbf{\Sigma_i}$, let

$$\mathbf{\Sigma_i} = \begin{bmatrix} \sigma_{i,1}^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_{i,K}^2 \end{bmatrix}. \tag{13}$$

Next, define the average weighted variance (AWV), $\bar{\Sigma}$, as

$$\bar{\Sigma} = \frac{1}{K} \sum_{i=1}^{M} \sum_{j=1}^{K} w_i \sigma_{i,j}^2. \tag{14}$$

From Fig. 1(b), it is clear that performance is comparable to the baseline system using only $\sim$1.5 hour of UBM data, which results in about $\sim$2.7 seconds of data from each utterance. This is not very surprising since this $\sim$1.5 hour of data contains all the inter-

speaker variabilities present in all the utterances in the original NIST SRE 04 corpus used. Very interestingly, from Fig. 1(a), a clear relation with $\bar{\Sigma}$ and system EER is observed. After using more than 1.5 hours of data, the variance parameter $\bar{\Sigma}$, saturates, which indicates that the 2.7 seconds of data used from each utterance is actually sufficient to represent the variability it has to offer for the UBM, in this case. Obviously this duration measure cannot be completely generalized. The scatter plot in Fig 2(b) shows the correlation between $\bar{\Sigma}$ and EER more clearly, with these two parameters having a correlation coefficient of -0.8372. From Fig. 2(a), an exponential relation can be seen between UBM training CPU time and the total UBM data amount. In these experiments, it was observed that using about 1.5 hours of data requires about 30 CPU minutes to train the UBM, while more than 200 CPU minutes are needed if all the data is used. Since CPU computation time can be considered linearly proportional to the complexity of training, this indicates that, contrary to popular belief, that "more training data is better", the addition of five times the computational resources with more than 160 hours of training data actually has a negligible contribution to improving overall system performance.

Thus, the conclusion drawn here is that if the selected UBM data set is well chosen, (i.e., contains sufficient speech/speaker variability) using all the features of each utterance is not necessary. In the next section, we investigate how system performance is affected based on alternative feature selection methods.

V. Sub-sampling of feature frames

In the previous section, it was established that only 2.7 seconds of data from each 5 min utterance of the UBM data is sufficient for system performance equivalent to the baseline configuration. In this section, several alternative approaches for selecting this subset of development data for UBM training are considered for more effective representation of the feature space of each utterance. Time versus frequency spectrograms of these approaches are illustrated in Fig. 3 (b)–(e) using a spectrogram of an original TIMIT utterance, shown in Fig. 3(a). The use of the first n feature vectors from each utterance, as done in the previous section, is termed as "leading feature selection" (LFS) and is depicted in Fig. 3(b). As noted earlier, sub-sampling the feature frames can also be done uniformly or randomly [13]. These methods are denoted as UFS ("uniform feature selection") and RFS ("random feature selection") [13], and illustrated in Fig. 3(c) and (d), respectively. Now, though the sub-sampling methods LFS, UFS and RFS would reduce computation time, they are overly simplified and completely data

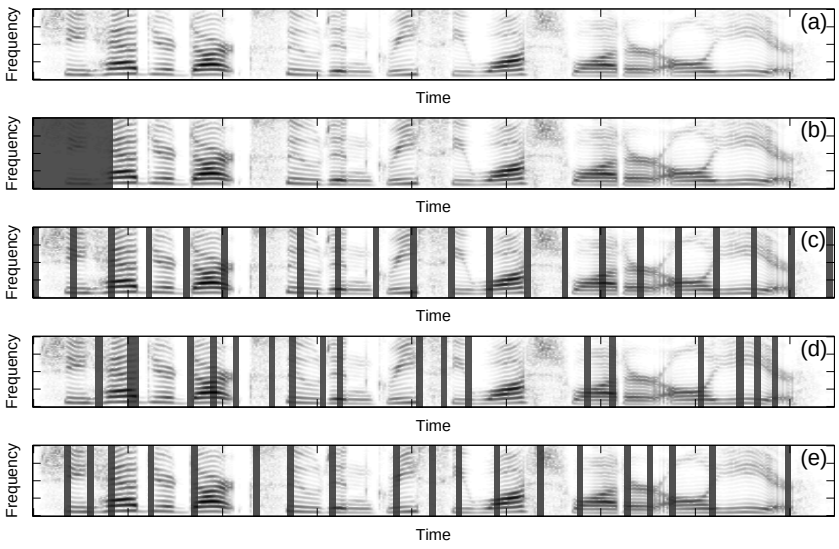

Fig. 3. Conceptual illustration of the feature selection schemes. (Selected frames are shown in dark.) (a) Original utterance spectrogram, (b) LFS, (c) UFS, (d) RFS and (e) IFS.

independent. These methods do not consider the specific distribution of the phonetic content over time. Thus, we propose a generic method termed as "intelligent" feature selection (IFS), which aims to select a diverse set of n training feature frames from input training utterances. This method assesses the similarity of successive frames using a phonetically motivated distance measure, with selection of a feature frame only if the corresponding dissimilarity is higher than some threshold. In Fig. 3 (e), a conceptual IFS method that attempts to select a frame from the beginning of each distinct phoneme is illustrated.

Clearly, there can be variations in this approach if alternative distance criteria between features are used. Since one design criteria here is training speed, in this phase the Euclidean distance is used due to its' simplicity.

A. Feature selection based on Euclidean distance

In this section, an intelligent feature selection (IFS) scheme is proposed based on the simple Euclidean distance between features (IFS-EU). The aim here is to estimate similarity between successive feature frames using this distance measure, and select a feature frame only if the frame is sufficiently different from those previously selected. From an intuitive understanding of the Euclidean distance, it is noted that this distance measure is related to the smoothed log-spectral distance [20] when applied to cepstral feature vectors. The formulation is started by deriving the probability density function (PDF) of the distance function between feature vectors.

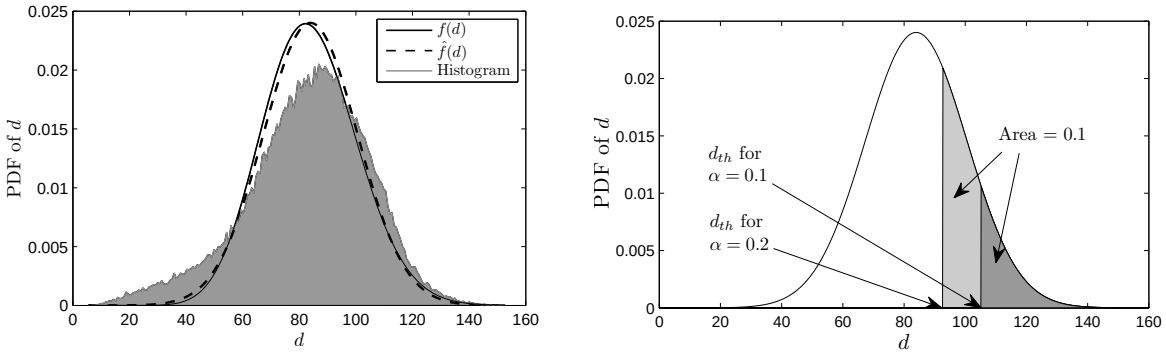

Fig. 4. (a) Comparison of the theoretical PDF, its Gaussian approximation and the actual PDF obtained from histogram. 13 dimensional MFCC coefficients were used and the parameter $\bar{\lambda}$ was calculated directly from the data. For this data, $\bar{\lambda} = 281.6836$, $\mu_D = 83.9506$ and $\sigma_D^2 = 276.07$. (b) A PDF of inter-feature Euclidean distance and the proposed distance threshold (shown for $\alpha = \ 0.1$ and 0.2).

1) PDF of Euclidean distance between features: Assume that the K dimensional feature vectors of the development speaker data, originating from a specific phone, can be modeled by an independent, wide sense stationary (WSS), white Gaussian vector random sequence $\mathbf{X}[n]$ with a covariance function matrix $\mathbf{K}_{XX}[m,n]$ given by,

$$\mathbf{K}_{XX}[m,n] = \text{diag}(\lambda_1 ... \lambda_K)\delta[m-n], \tag{15}$$

where m, n denotes the feature indices, and λ_p ($p = 1, .., K$) are the variances of the individual cepstral coefficients. The Euclidean distance between the mth and nth feature vector will be,

$$d(m,n) = ||\mathbf{X}[m] - \mathbf{X}[n]||^{\frac{1}{2}}. \tag{16}$$

The feature vectors have a common mean, and thus the term inside the parenthesis in (16) will be a zero mean vector random sequence. Also, due to the independence assumption, $d(m,n)$ is independent of m and n. Thus,

$$d(m,n) = d = ||\mathbf{Z}||^{\frac{1}{2}}, \tag{17}$$

where $\mathbf{Z}$ is a zero mean Gaussian random vector having a covariance matrix $\mathbf{K}_{ZZ}$, and found to be,

$$\mathbf{K}_{ZZ} = \text{diag}(2\lambda_1 ... 2\lambda_K). \tag{18}$$

The factor of 2's are introduced because each element of $\mathbf{Z}$ is constructed from the subtraction of two independent white Gaussian random variables having variances λ_p ($p =$

$1, .., K$). From (17), it is possible to write,

$$d^2 = \sum_{i=1}^{K} Z_i^2 = \sum_{i=1}^{K} (2\lambda_i) W_i^2, \text{ where } W_i \sim \mathcal{N}(0,1). \tag{19}$$

For simplification, assume that the effect of the individual λ_i values in (19) can be approximated using a lumped parameter $\bar{\lambda}$. Thus,

$$d^2 \approx 2\bar{\lambda} \sum_{i=1}^{K} W_i^2 = 2\bar{\lambda} Y, \tag{20}$$

where $\bar{\lambda}$ is defined as the average variance given by,

$$\bar{\lambda} = \frac{1}{K} \sum_{i=1}^{K} \lambda_i. \tag{21}$$

In (20), $Y = \sum_{i=1}^{K} W_i^2$ is a squared sum of zero mean independent Gaussian random variables, and thus will follow a chi-squared distribution given by,

$$f_Y(y) = \frac{(1/2)^{\frac{K}{2}}}{\Gamma(K/2)} y^{(\frac{K}{2}-1)} e^{-y/2}. \tag{22}$$

From (20), $d = \sqrt{2\bar{\lambda}Y}$. Using this transformation in (22), the PDF of d can be obtained as,

$$f_D(d) = \frac{2^{1-K}}{\Gamma(K/2)} \frac{d^{K-1}}{\bar{\lambda}^{K/2}} \exp\left(-\frac{d^2}{4\bar{\lambda}}\right). \tag{23}$$

The mean and variance of this distribution can be found as,

$$\mu_D = \frac{2\sqrt{\bar{\lambda}}\Gamma(\frac{1+K}{2})}{\Gamma(K/2)} \text{ and} \tag{24}$$

$$\sigma_D^2 = 2K\bar{\lambda} - \mu_D^2, \text{ respectively.} \tag{25}$$

Note that the PDF of d will provide the likelihood of the distance between any two features in the data set. In other words, if the goal is to select feature vectors that are farther apart on average, a set of features should be selected in which each pair has a distance greater than μ_D.

2) Calculation of distance threshold: In this feature selection problem, the data is processed on a frame-by-frame basis. Assuming that the PDF parameters are known for the current frame, select the next frame if its distance from the current frame is greater than a

threshold d_{th}. For a fixed value $\alpha \in [0, 1]$, define d_{th} as,

$$P[d > d_{th}] = \int_{d_{th}}^{\infty} f_D(z)\mathrm{d}z = \alpha. \tag{26}$$

The process is illustrated in Fig. 4(b) for $\alpha = 0.1$ and 0.2. This implies that a feature vector is selected only if its' distance from the current feature is so high that the event is less probable than α, suggesting a high likelihood of a change in the phone represented by the feature. It is observed that the PDF, $f_D(d)$, can be closely approximated by a Gaussian distribution having a mean μ_D and variance σ_D^2. Fig. 4(a) compares the function $f_D(d)$, its Gaussian approximation $\hat{f}_D(d)$, and a histogram plot, estimated from 13 dimensional MFCC coefficients of a UBM utterance. Using the Gaussian approximation, it is possible to obtain from (26),

$$d_{th} = \mu_D + \sqrt{2}\sigma_D \mathrm{erfc}^{-1}(2\alpha), \tag{27}$$

where erfc^{-1} is the inverse of the complementary error function (erfc). Here, erfc() is defined as:

$$\mathrm{erfc}(x) = \sqrt{\frac{2}{\pi}} \int_0^x e^{-t^2}\mathrm{d}t.$$

3) Estimation of PDF parameters: Here a recursive method is employed for estimating the feature vector mean and variance similar to [21]. Denoting $\boldsymbol{\lambda}[n]$ as the vector containing the diagonal elements of $\mathbf{K_{XX}}[0, 0]$, and $\boldsymbol{\mu_X}[n]$ as the mean vector of the nth frame, the equations

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[n] = \beta_m \hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[n-1] + (1-\beta_m)\mathbf{X}[n], \text{ and} \tag{28}$$

$$\hat{\boldsymbol{\lambda}}_{\boldsymbol{X}}[n] = \beta_v \hat{\boldsymbol{\lambda}}_{\boldsymbol{X}}[n-1] + (1-\beta_v)||\mathbf{X}[n] - \hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[n]|| \tag{29}$$

are used, where β_m, $\beta_v \in [0, 1)$ are overall smoothing parameters.

4) Implementation: Let i denote the current frame index and set $j = i+1$. For initialization ($i = 1$), $\mathbf{X}[i]$ is always selected, and $\hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[i]$ and $\hat{\boldsymbol{\lambda}}_{\boldsymbol{X}}[i]$ are calculated from $\mathbf{X}[i]$ and $\mathbf{X}[j]$ as,

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[i] = 0.5(\mathbf{X}[i] + \mathbf{X}[j]) \text{ and} \tag{30}$$

$$\hat{\boldsymbol{\lambda}}_{\boldsymbol{X}}[i] = 0.5(\mathbf{X}[i] \cdot^2 + \mathbf{X}[j] \cdot^2) - \hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[i] \cdot^2, \tag{31}$$

where $()\cdot^2$ denotes an element-wise square operation. Next, $\bar{\lambda}$ and d_{th} are calculated using (21) and (27). Now, j is iteratively incremented by 1 and $d(i, j)$ is calculated from (16). The values $\hat{\boldsymbol{\mu}}_{\boldsymbol{X}}[i]$ and $\hat{\boldsymbol{\lambda}}_{\boldsymbol{X}}[i]$ are updated in each step using (28) and (29), along with the

TABLE I

COMPARISON OF DIFFERENT UBM TRAINING SCHEMES WITH RESPECT TO EER AND TRAINING CPU TIME.

Method	**%data**	**EER** (%)	**CPU Time** h:mm
Baseline	100%	11.43	3:46
LFS	1%	11.48	0:24
UFS	1%	11.54	0:22
RFS	1%	11.41	0:18
IFS-EU	1%	**10.99**	0:27

threshold d_{th}. If $d(i,j) > d_{th}$ is found, $\mathbf{X}[j]$ is selected. Next, $i = j$ and $j = i + 1$ is set and the process is repeated until the desired number of features are selected. In our experiments, the settings used are $\alpha = 0.1$, $\beta_m = 0.8$ and $\beta_v = 0.6$.

B. Performance of sub-sampling schemes

The EER performance along with the computation time required for UBM training using the set of presented approaches is shown in Table I. Baseline performance with $100\,\%$ of the data used to train the UBM is $11.43\,\%$. It is clear that all four UBM training methods considered here, using a mere $1\,\%$ of UBM data employed can provide performance equivalent to the baseline system with up to a 7 times reduction in CPU computation time. In addition, using the proposed feature selection scheme, denoted by IFS-EU, it is noted that a $\sim$0.4 $\%$ reduction in EER is achieved in comparison to the baseline system. This is because the selected features in the IFS-EU method are better able to represent the diverse speaker pool, while suppressing some of the fine model traits of the intra-speaker phoneme variability, which, it is believed, are less important for construction of an effective UBM. However, we clarify that we do not claim that the 99% data not used for UBM training are definitely harmful for the system. We simply emphasize the fact that more data is not necessarily better, and sub-sampling this data properly can provide equivalent or even better performance than using all the data.

VI. UBM DATA: NUMBER OF SPEAKERS

In this section, the impact of changing the number of unique speakers in the UBM training data on system performance is considered. It was shown in the previous section, that if the data contains sufficient variability, a very small portion of data should be sufficient for training the UBM. Now, different speakers should possess different feature speech/physiology characteristics, indicating that an increase in the number of speakers in the UBM should lead

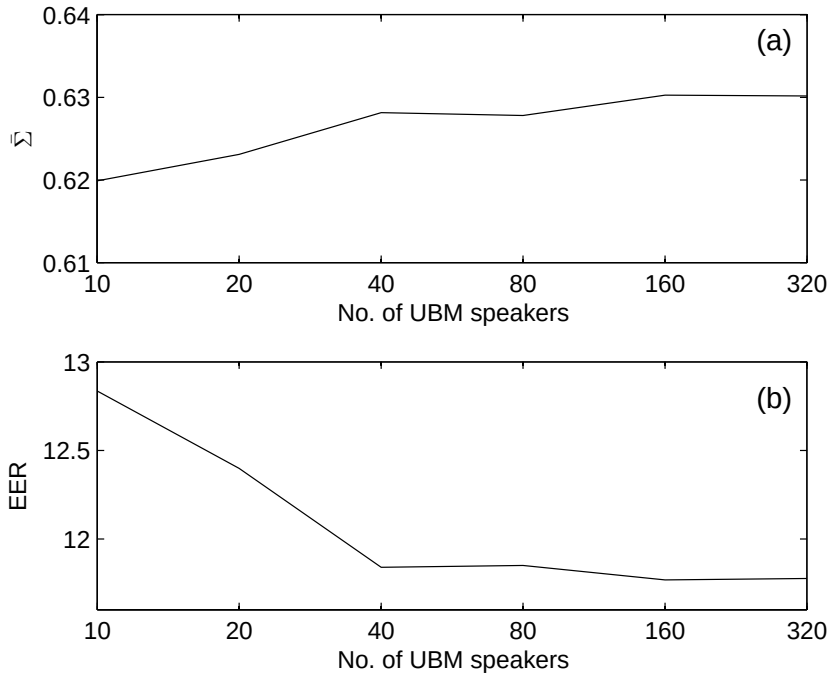

Fig. 5. Variation of (a) AWV ($\bar{\Sigma}$) and (b) system performance with the change of number of UBM speakers.

to an increase in the variance of the data. Intuitively, this should be beneficial to overall system performance. An experiment is performed to validate this hypothesis.

In this experiment, the amount of data was kept fixed and the number of unique speakers was varied from 10 to 320 in an exponential manner. The UBM dataset-2 was used in this case because it has a larger number of speakers. For each UBM training run, the specified number of speakers are selected randomly from the pool and system performance is computed for the UBM trained with those speakers' features. Five independent experimental runs are performed and the average of those EER are calculated. The AWV values are also calculated for each UBM using (14). In Fig. 5(a) and (b), AWV ($\bar{\Sigma}$) and EER are plotted against the number of UBM speakers. As we have expected, the system performance is improved drastically, as the number of UBM speakers are increased, with an increase of the AWV. At some level, overall performance saturates. Thus, we justify that introducing a new speaker increases the variability of the UBM data (as reflected in the increase in AWV), which can benefit system performance. This in effect helps further justify the argument of [2] regarding the amount of training data.

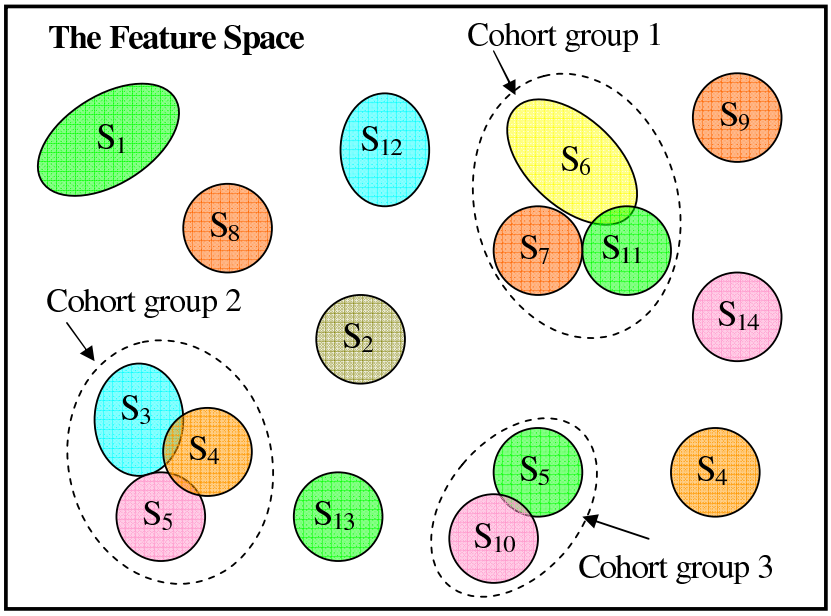

Fig. 6. A schematic diagram of the speaker space in the UBM.

VII. Selection of UBM speakers

In this section, the goal is to investigate the issue of using a subset of all the available speakers in the UBM. It has been established that having a large number of dissimilar speakers in the UBM data aids in improving system performance. However, it is known that many speakers have similar acoustic properties, (i.e. the cohort speakers) that may again introduce an unbalance in the UBM data. This issue can be illustrated with a hypothetical feature space in Fig. 6. In this data-set, if equal amounts of data from all speakers are used to train the UBM, the similar speakers that are clustered together, (i.e. cohort groups 1,2 and 3) will be emphasized in the UBM. This would result in a higher score from the UBM in the likelihood ratio test if the a test speaker is from that cohort group. Now, it would be better if these speakers are spread out in the feature space as much as possible so that the entire acoustic space is uniformly covered (assuming a uniform open speaker test space). However, practically there are some problems in that scenario.

- The feature space is multidimensional, which means uniformly covering this space would require an infinite number of speakers.
- In reality, the speaker features are not very easily distinguishable as in the simplistic illustration in Fig. 6, rather they are highly overlapping.

Thus, the motivation here is to use a reduced number of speakers than all the available speakers according to some speaker divergence criteria, so that closely related speakers are not used in the UBM (i.e., if the speaker is already included in the training set, do not include an acoustically close neighbor as well).

A. KL divergence based speaker selection (KL-D)

Here a UBM speaker selection method is developed using the Kullback-Leibler (KL) divergence between speaker models. For each UBM speaker s_i, $i \in (1, N_s)$, a GMM model Λ_i is trained. To calculate the similarity between GMMs, the symmetric KL divergence [22], [23], is used, given by,

$$D^s_{KL}(\Lambda_i, \Lambda_j) = E_{\Lambda_i(\mathbf{X})}\left[\log \frac{\Lambda_i(\mathbf{X})}{\Lambda_j(\mathbf{X})}\right] E_{\Lambda_j(\mathbf{X})}\left[\log \frac{\Lambda_j(\mathbf{X})}{\Lambda_i(\mathbf{X})}\right], \tag{32}$$

where $\Lambda_i(\mathbf{X})$ and $\Lambda_j(\mathbf{X})$ are likelihoods of occurrence of the observation vector $\mathbf{X}$, given that it belongs to speaker model Λ_i and Λ_j, respectively. Next, compute the $N_s \times N_s$ divergence matrix, obtaining the KL score for each pair of speakers. To measure how diverse a speaker i is from the all other speakers, define a diversity factor D_i, given by,

$$D_i^{(\text{KL})} = \frac{1}{N_s} \sum_{j \in N_s, j \neq i} D^s_{KL}(\Lambda_i, \Lambda_j). \tag{33}$$

This relation means, D_i is a measure of the average divergence of the model Λ_i, from all other speaker models. Thus, after computing all D_i values, they are sorted according to their absolute value, and the top N_D most divergent speakers are selected for the UBM.

B. Speaker Selection using prototype UBM (P-UBM)

In this method, to find the most divergent speakers, all the data is pooled and a prototype UBM model, Λ_0 is trained. Assuming this UBM holds a central position in the GMM space, an attempt to find speakers that are most divergent from this UBM is performed. The diversity factor for each speaker i is computed simply from the likelihood of occurrence of that speaker's features given the model Λ_0.

$$D_i^{(\text{P})} = \Lambda_0(\mathbf{X_i}). \tag{34}$$

In a similar way, the $D_i^{(\text{P})}$ values are sorted and the top N_D divergent speakers from the prototype UBM is selected. It is noted that this is a simplistic method for speaker selection and does not guarantee that the selected speakers are diverse among themselves.

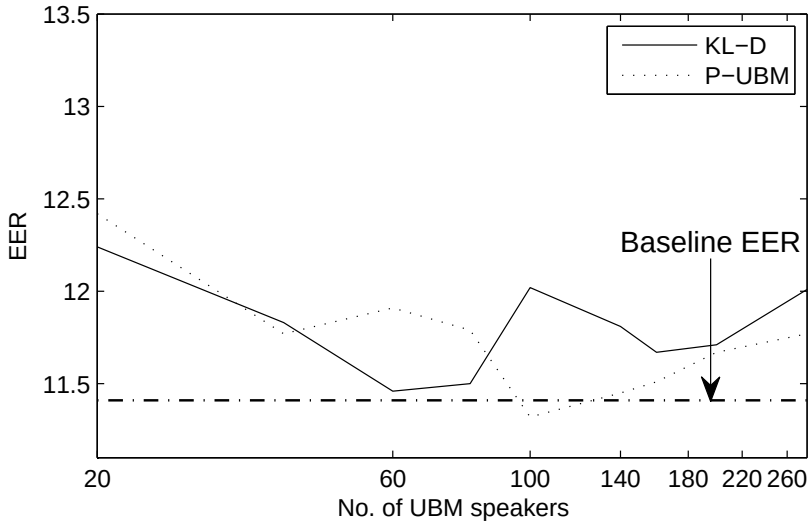

Fig. 7. System performance variation with the change of number of UBM speakers selected using different methods.

C. Results of speaker selection methods

In these experiments, the UBM database-2 is used and the total amount of data is held fixed to 1.5 hours. The system is evaluated by varying the number of speakers, N_D from 20 to 300 in an exponential fashion while the proposed KL-D and P-UBM methods were used for selecting the best speakers. The exact number of frames were selected from each utterance using the LFS method (described in Secion V) from the selected speakers' data such that the total amount of data equals 1.5 hours. For all values of N_D, the EER values obtained are plotted in Fig. 7. It should be noted that 1.5 hours of data used in this case is only 0.3% of the complete UBM database-2 (which contains 473.75 hours of data). Thus, according to experiments in Section IV, this amount is not sufficient to retain the baseline system performance. However, this lower amount is still employed so that (a) it is more convenient to analyze the effect of the selected speakers data, and (b) enough data per speaker is available for the case of a lower number of selected speakers.

From Fig. 7 we observe that, though there are some fluctuations[1] it can be seen that both methods perform very close to the baseline system using a significantly lower number of speakers (i.e., 60 and 100 for methods KL-D and P-UBM, respectively). Notice that this close to baseline performance is achieved with only 1.5 hours of data, instead of 473.75

[1]We believe this fluctuation in EER is due to the fact that the GMMs are trained on multiple utterances of the same speaker from different channels. This creates a mild bias toward the dominating channel type in each GMM, resulting in slight channel dependent clustering in some cases. Future studies could explore UBM construction by removing channel effects from the UBM utterances using techniques like JFA, total variability features [24], IIFA [25], etc.

TABLE II

COMPARISON OF DIFFERENT SPEAKER SELECTION APPROACH FOR UBM TRAINING WITH RESPECT TO EER AND NO. OF SPEAKERS.

Method	**%data**	**Number of speakers**	**EER** (%)
Baseline	100%	392	11.41
KL-D	0.31%	60	11.46
KL-D	0.62%	60	**11.30**
P-UBM	0.31%	100	11.32

hrs of data. Interestingly, after a certain point, system performance actually degrades as the number of speakers is increased in the proposed methods, which we believe is due to the introduction of similar/redundant speakers that are creating an imbalance in the UBM data. As expected, the performance does not reach the baseline for a larger number of speakers for either method due to the use of 1.5 hours of data which is not sufficient to retain the variability of the dataset. Now that we identified the regions in the plot where the proposed methods perform the best, we attempt to increase the data amount for the selected speakers for further performance improvement. For the KL-D approach using 60 speakers, we increased the amount of data from 1.5 hours to 3 hours and obtained an improvement in EER from 11.46% to 11.30%, which is better than the baseline EER. For the P-UBM method, using 100 speakers and 1.5 hours of data already provides a slight improvement over baseline system which uses the full UBM database-2. The results are summarized in Table II.

Note that there are 392 speakers in UBM database-2, which means less than 30% of the speakers were used in both proposed methods. Also, less than 1% of the total amount of data was used for training the UBM. Thus, we conclude that if a diverse set of speakers can be carefully selected, a much lower number of speaker data can provide performance equivalent to/better than the baseline system.

VIII. CONCLUSIONS

In this study an organized method is developed for determining the data to be selected for effective UBM training. Rigorous experiments were performed showing the relationship between data variance and overall speaker verification system performance. Four efficient sub-sampling schemes for feature frame selection were presented with potential benefits of reducing the computation time by up to 7 fold. A new intelligent feature frame sub-sampling algorithm is proposed which is experimentally shown to outperform the baseline system that uses all the available data. The implication of selectively using speaker data for UBM

construction was analyzed and two effective speaker selection methods were proposed and evaluated. The results show that carefully selected reduced speech data size and speaker count are sufficient to achieve effective speaker verification performance.

REFERENCES

[1] T. Hasan, Y. Lei, A. Chandrasekaran, and J. H. L. Hansen, "A novel feature sub-sampling method for efficient universal background model training in speaker verification," in *Proc. IEEE ICASSP'2010*, Dallas, Texas, March 2010, pp. 4494–4497.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models,," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19 – 41, 2000.

[3] W. Campbell, D. Sturim, D. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP 2006*, vol. 1, Toulouse, France, May 2006.

[4] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 980–988, July 2008.

[5] R. Kuhn, J. Junqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, pp. 695–707, 2000.

[6] A. Solomonoff, W. Campbell, and I. Boardman, "Advances in channel compensation for SVM speaker recognition," in *Proc. IEEE ICASSP 2005*, vol. 1, 18-23, 2005, pp. 629–632.

[7] "The NIST year 2008 speaker recognition evaluation plan," 2006. [Online]. Available: http://www.nist.gov

[8] H. Li, B. Ma, K.-A. Lee, H. Sun, D. Zhu, K. C. Sim, C. You, R. Tong, I. Karkkainen, C.-L. Huang, V. Pervouchine, W. Guo, Y. Li, L. Dai, M. Nosratighods, T. Tharmarajah, J. Epps, E. Ambikairajah, E. S. Chng, T. Schultz, and Q. Jin, "The I4U system in NIST 2008 speaker recognition evaluation," in *Proc. IEEE ICASSP 2009*, Apr. 2009, pp. 4201–4204.

[9] W. Guo, Y. Long, Y. Li, L. Pan, E. Wang, and L. Dai, "iFLY system for the NIST 2008 speaker recognition evaluation," in *Proc. IEEE ICASSP 2009*, April 2009, pp. 4209–4212.

[10] L. Burget, M. Fapo, V. Hubeika, O. Glembek, M. Karafit, M. Kockmann, P. Matejka, P. Schwarz, and J. Cernock, "BUT system description: NIST SRE 2008," in *Proc. 2008 NIST Speaker Recognition Evaluation Workshop*. National Institute of Standards and Technology, 2008, pp. 1–4.

[11] Y. Bar-Yosef and Y. Bistritz, "Adaptive individual background model for speaker verification," in *Proc. Interspeech'09*, 2009.

[12] A. Sarkar, S. Umesh, and S. P. Rath, "Text-independent speaker identification using vocal tract length normalization for building universal background model," in *Proc. Interspeech'09*, 2009.

[13] C. Barras, X. Zhu, J.-L. Gauvain, and L. Lamel, *The CLEAR'06 LIMSI Acoustic Speaker Identification System for CHIL Seminars*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2007, vol. 4122.

[14] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," *Speech Commun.*, vol. 17, no. 1-2, pp. 91–108, 1995.

[15] A. E. Rosenberg, J. DeLong, C. H. Lee, B. H. Juang, and F. K. Soong, "The use of cohort normalized scores for speaker verification," in *ISCA ICSLP-92*. ISCA, 1992, pp. 599–602.

[16] T. Matsui and S. Furui, "Similarity normalization method for speaker verification based on a posteriori probability," in *ISCA Workshop on Automatic Speaker Recognition, Identification and Verification*, 1994.

[17] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. 2001: A Speaker Odyssey*, 2001, pp. 213–218.

[18] S. Young, "HTK reference manual," *Cambridge University Engineering Department*, 1993.

[19] E. Dalmasso, F. Castaldo, P. Laface, D. Colibro, and C. Vair, "Loquendo - Politecnico di Torino's 2008 NIST speaker recognition evaluation system," in *Proc. IEEE ICASSP 2009*, April 2009, pp. 4213–4216.

[20] S. Furui, *Digital Speech Processing, Synthesis, and Recognition*. Marcel Dekker Inc, 1989.

[21] M. M. Bruce, *Estimation of variance by a recursive equation*. NASA Technical note, 1969.

[22] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951. [Online]. Available: http://www.jstor.org/stable/2236703

[23] M. Ben and F. Bimbot, "D-map: a distance-normalized map estimation of speaker models for automatic speaker verification," in *Proc. IEEE ICASSP'03*, April 2003, pp. II–69–72 vol.2.

[24] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front-end factor analysis for speaker verification," *submitted to IEEE Transaction on Audio, Speech and Language Processing.*, 2010.

[25] Y. Lei and J. Hansen, "Factor analysis-based information integration for Arabic dialect identification," in *Proc. IEEE ICASSP'2010*, Dallas, Texas, March 2010, pp. 4337–4340.