

A Fast Implementation of Factor Analysis for Speaker Verification

Qingsong Liu¹, Wei Huang², Dongxing Xu², Hongbin Cai², Beiqian Dai¹

¹University of Science and Technology of China, Hefei, China

²Shanda Innovation Institute, Shanghai, China

liuqs@mail.ustc.edu.cn, {huangwei03,xudongxing,caihongbin}@snda.com, bq dai@ustc.edu.cn

Abstract

The problem of session variability in text-independent speaker verification has been tackled actively for a few years. The factor analysis approach has been successfully applied for solving the session variability problem. However, it suffers from a large amount of computational overhead. In this paper, a fast implementation of factor analysis approach with GMM Gaussian pre-selection is proposed. In our method, the EM statistics are calculated only using the Gaussians selected by cluster UBM to improve the speed of estimating factor analysis model. Experimental results on the NIST SRE 2006 evaluation show that the presented approach can provide as much as a 7 to 8x speedup over the baseline factor analysis system with the similar performance.

Index Terms: speaker verification, session variability, factor analysis

1. Introduction

The challenge in the current speaker verification is to recognize a speaker given enrollment data extracted from one particular speech recording and test data extracted from other recordings. This brings on the mismatch between training and testing when speaker models are compared with speaker data collected from different microphones, channels, and environments. Speaker models estimated by classical MAP approach contain not only the speaker information, but also include the enrollment recording conditions and other useless information. Current speaker recognition systems compensate for session (channel or inter-speaker) variability by feature normalization techniques such as feature warping [1] and feature mapping [2]; as well as score compensation techniques including HNorm [3] and Tnorm [4].

In [5], the factor analysis (FA) approach is successfully applied for compensating inter-session variability and shows great improvement with large corpora such as the Switchboard database. The novelty brought by this approach is that it assumes the session variability space to be continuous compared to the feature mapping technique. However the computational requirement of FA makes it difficult to be implemented in the realtime speaker verification system, which is the primary motivation for developing an efficient implementation to speed up the factor analysis approach.

The work in this paper is based on a theoretical framework by Kenny [5]. We anatomized the steps of factor analysis and found that about 90 percent computation is brought by calculating the *a posteriori probability* of each frame with respect to all Gaussian mixture of UBM. In this paper, the details of fast factor analysis implementation using GMM mixture pre-selection are described to speed up both the estimation of speaker model and test. We use a cluster GMM to reduce the computation of EM statistics. Experiments on NIST SRE 2006 evaluation show

that the new implementation has excellent performance and efficiency compared to classical MAP and baseline factor analysis system.

The paper is organized as follows: a brief overview of factor analysis approach is provided in section 2. In section 3, an efficient implementation of factor analysis is described in detail. Section 4 lists the experimental protocol while section 5 presents the results of the approach. Section 6 is the conclusion.

2. Factor Analysis

We assume a fixed GMM structure containing a total of N mixture components. Let F be the dimension of the acoustic feature vectors. The GMM mean supervector is defined as the concatenation of all Gaussian means: its dimension is $NF(N * F)$.

In the factor analysis approach [6], a basic assumption is given that a speaker- and session-dependent GMM mean supervector m can be decomposed into a sum of two supervectors.

$$m = m_s + m_h \tag{1}$$

where supervector m_s depends on the speaker and supervector m_h depends on the channel. They are statistically independent and normally distributed. Secondly, we assume that the distribution of m_s has a hidden variable description of the form:

$$m_s = m_0 + Vy + Dz \tag{2}$$

where m_0 is a speaker- and session-independent supervector with $NF * 1$ dimension, actually, m_0 is the mean supervector of UBM. V is a rectangular matrix of low rank and y is a normally distributed random vector, D is a $NF * NF$ diagonal matrix and z is a normally distributed NF dimensional random vector. We will refer to the components of y as speaker factors and z as common speaker factor. Also, m_h has a hidden variable description of the form:

$$m_h = Ux \tag{3}$$

where U is a $NF * R$ matrix of rank R . We refer to the components of x as channel factors. The speaker factors and the channel factors play different roles in that, for a given speaker, the values of speaker factors y are assumed to be the same for all recordings of one speaker but the channel factors x are assumed to vary from one recording to another.

Given an enrollment utterance for a speaker, the speaker supervector m_s can be estimated by subtracting the channel supervector m_h for the enrollment utterance from the data. For test utterance, channel compensation (removing the channel information) can be performed on the feature level. Therefore, the success of factor analysis approach relies on a good estimation of the U matrix, known as the eigenchannels (or channel

loadings). In our approach, a sufficiently high amount of data which have a high number of different recordings per speaker are needed to estimate U .

To estimate the latent variable of equation (2) and (3), the zero and first order EM statistics need to be calculated on the training data with respect to the UBM. Let N_s and $N_{(h,s)}$ be vectors containing the zero order speaker-dependent and session-dependent EM statistics, and X_s and $X_{(h,s)}$ are vectors containing the first order speaker-dependent and session-dependent EM statistics. For a speaker s and a session h of s , respectively:

$$N_s[g] = \sum_{o_t \in s} \gamma_g(o_t); \quad N_{(h,s)}[g] = \sum_{o_t \in (h,s)} \gamma_g(o_t) \quad (4)$$

$$X_s[g] = \sum_{o_t \in s} \gamma_g(o_t) \cdot o_t; \quad X_{(h,s)}[g] = \sum_{o_t \in (h,s)} \gamma_g(o_t) \cdot o_t \quad (5)$$

where g is the Gaussian index, $\gamma_g(o_t)$ is the *a posteriori probability* of Gaussian g for the observation o_t . In the equation (4) and (5), $\sum_{o_t \in s}$ means the sum over all frames belonging to the speaker s and $\sum_{o_t \in (h,s)}$ means the sum over all frames belonging to the session h of speaker s . After these four EM statistics are calculated, the speaker factor y , channel factor x and matrix U can be estimated by ML (Maximum-Likelihood) criterion and the details are described in [6].

3. Factor analysis with GMM components pre-selection

In equation (4) and (5), we must calculate the *a posteriori probability* of each Gaussian mixture for all the observed frames. For example, if the length of training data is about 5 minutes (generate 30,000 frames with 20ms frame length, 10ms frame shift) and UBM consists of 512 Gaussian mixtures, we must perform approximately 15,360,000 (30,000*512) computations for EM statistics, which is a huge computation for training a target model. Moreover, there are thousands of recordings need to be calculated to estimate U matrix. In this section, a cluster UBM [7] is used to select Gaussian components and speed up the estimation of factor analysis model.

3.1. Cluster UBM

Figure 1 describes the $\gamma(o_t)$ of one frame o_t with respect to all Gaussian mixtures of UBM. The number of Gaussians is 512. It is obvious that only a few Gaussian mixtures observed by frame o_t are useful for calculating EM statistics and the number of observed Gaussian mixtures is much less than Gaussian number of UBM. Moreover, the $\gamma(o_t)$ of unobserved Gaussian mixtures are approximately zero and useless to calculate EM statistics.

Therefore, we can decrease the computational load if only the observed Gaussian mixtures are used to calculate the EM statistics. However, since the observed Gaussian mixtures are different for each frame, In this paper, a cluster GMM based Gaussian selector (as shown in Figure 2) is designed to select those observed UBM components. we also introduce an approach (called CUBM-FA) to estimate GMM mixture pre-selection based factor analysis model.

Figure 2 shows a hierarchy GMM, $\{G_1, \dots, G_N\}$ are the Gaussian mixture densities of UBM, and $\{C_1, \dots, C_k\}$ are local Gaussian mixtures which are formed by clustering the Gaussian components of UBM (known as cluster UBM, or CUBM).

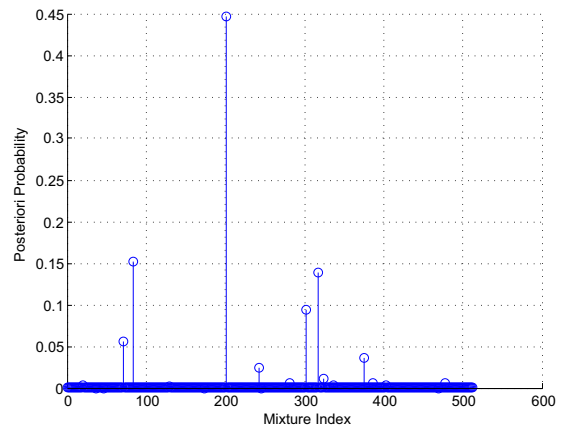


Figure 1: A *a posteriori* probabilities of one frame with respect to UBM.

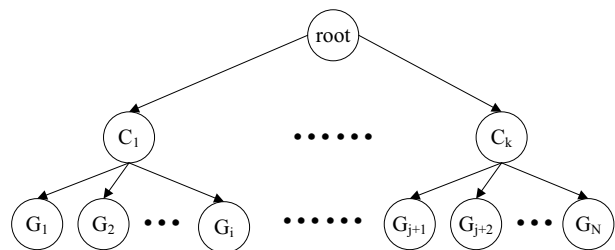


Figure 2: Cluster GMM based Gaussian selector.

K is the number of local Gaussian mixtures and N is the number of UBM Gaussian components. All Gaussian components $\{C_1, \dots, C_k\}$ belong to a root node.

The Gaussian mixtures of UBM are clustered to K classes to obtain the CUBM. The most used distance measure is Kullback-Leibler Divergence (KLD) which is used to measure the distance between two Gaussian densities f and g . Here, we use symmetrised divergence, defined as the following equation (6).

$$SD(f, g) = \int f(x) \log \frac{f(x)}{g(x)} dx + \int g(x) \log \frac{g(x)}{f(x)} dx \quad (6)$$

We calculate the symmetrised KLD between current Gaussian with all the other Gaussian mixtures and then merge the nearest two Gaussian mixtures to a new Gaussian [8], by using equation (7). Then the merged Gaussian replaces the original two Gaussian mixtures and this forenamed processes repeat until we get K classes.

$$\begin{aligned} \mu &= \frac{c_i \mu_i + c_j \mu_j}{c_i + c_j}, \quad c = c_i + c_j \\ \Sigma &= \frac{c_i \Sigma_i + c_j \Sigma_j + \frac{c_i c_j}{c_i + c_j} (\mu_i - \mu_j)(\mu_i - \mu_j)^T}{c_i + c_j} \end{aligned} \quad (7)$$

After CUBM is obtained, the procedure of fast implementation of factor analysis can be summarized by three steps and described as follows:

1. For each frame, CUBM can be viewed as a pre-selector of UBM components. We begin by calculating the $\gamma_c(o_t)$ of an observed vector o_t with respect to the CUBM components.

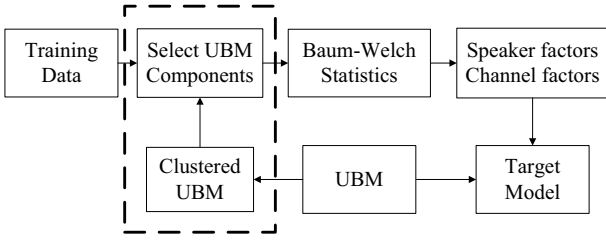


Figure 3: Training phase of GMM components pre-selection based factor analysis.

2. Select the CUBM component c_{\max} which has the maximal $\gamma(o_t)$.

$$c_{\max} = \arg \max_c \gamma_c(o_t) \quad (8)$$

3. Calculate EM statistics only with Gaussian mixtures of UBM falling into c_{\max} , and set EM statistics with other mixtures which are not in c_{\max} to zero directly.

$$\gamma_g(o_t) = \begin{cases} \gamma_g(o_t) & g \in c_{\max} \\ 0 & g \notin c_{\max} \end{cases} \quad (9)$$

As mentioned above, from Figure 1, we can find that $\gamma(o_t)$ of unobserved mixtures are almost near zero. So in equation (9), it is reasonable to simply zero these values directly, reducing the number of computations necessary.

Figure 3 is the block diagram of training phase of GMM components pre-selection based factor analysis system. As described above, only the selected UBM components are used to calculate the EM statistics when estimating the latent variables (x, y) and U matrix of equation (4). By using CUBM, for each frame, we must compute $K + N_i$ times computation of $\gamma(o_t)$ ($K + N_i \ll N$), where N_i is the number of UBM components belonging to c_{\max} . For example, for a UBM with 512 Gaussian mixtures, if the UBM Gaussian mixtures are clustered to 16 classes and on average, each class consists of 64 Gaussian mixtures. The calculation of $\gamma(o_t)$ for each frame is performed only 48 times $(16+512/16)$. Therefore, in theory, CUBM can speed up the factor analysis approach about 10 times with minimal accuracy lost.

3.2. Top-N based selection

Along with the increasing of K , the average number of UBM Gaussians which fall into each CUBM component will shrink, resulting in increased efficiency but poorer performance. This is due to the fact that for each frame, only the selected UBM components are used to estimate EM statistics. With fewer UBM Gaussians it is difficult to estimate the statistics accurately and will result in erroneous selection of UBM components.

We propose a Top-N based class selection approach to improve the performance of CUBM-FA approach and balance the efficiency and accuracy. Before using $\gamma_c(o_t)$ to estimate of statistics, we firstly calculate $\gamma_c(o_t)$ of the observed vector o_t with respect to the CUBM components $\{C_1, \dots, C_k\}$ and rank the scores. The N Gaussian classes with top scores are selected instead of choosing the single best in previous approach. This Top-N based selection can improve the accuracy of choosing UBM mixtures, but decrease the speed of the system compared to CUBM-FA. The experiments which show the balance between system performance and efficiency are presented in section 5.

4. Experimental Setup

4.1. Database

Speaker verification experiments are performed based upon the NIST SRE 2005 as a development set, and 2006 database for the validation set [9], male speakers only. The 2005 protocol consists of 274 speakers, 9012 tests (951 target tests, the rest are impostor trials) while the 2006 protocol consists of 354 speakers, 9720 tests (741 target tests, the rest are impostor trials). Each utterance contains about 2.5 minutes of speech in average. The intersession variability matrix U is enrolled on the NIST SRE 2004 database with 2938 examples of 124 speakers (around 20 iterations to reach convergence), UBM is estimated based upon the NIST SRE 2004 and 2005 development data set.

In this paper, we report the Minimal DCF (MinDCF) value obtained a posteriori (taking the following form $DCF = 0.1 * P_{Miss} + 0.99 * P_{FalseAlarm}$). The Equal Error Rate (EER) and DET curves are also provided as another performance measure [9].

4.2. Feature extraction

12-dimensional MFCCs are extracted every 10ms using a Hamming window of 20ms. The first-order deltas and second-order deltas are appended to the cepstral vectors and form a 36 dimensional feature vectors. Then cepstral mean and variance normalization (CMVN) is applied to the MFCCs to remove linear channel effects. RASTA filter and feature warping are also applied to improve the performance. In our experiments, the UBM Gaussian number is 512 and we set R (the rank of U) to 40, so the size of session variability matrix U is 18432×40 .

5. Results

The following detailed the experimental results obtained with the approach given in this paper. Both performance and computation of three systems (CUBM-FA, FA and GMM-UBM) are compared in this section. The results of Top-N based selection are also presented. We compute the computational cost in runtime of estimating all target models, and The average time for one target model is shown in the third column in table 1 and table 2.

Table 1: Performance and computation with varying number of CUBM components

	EER(%)	MinDCF	Cost Time(secs)
GMM-UBM	8.94	0.0433	—
FA baseline	3.76	0.0208	9.53
CUBM-FA 2	4.25	0.0224	5.11
CUBM-FA 4	4.53	0.0243	3.86
CUBM-FA 8	4.79	0.0252	3.03
CUBM-FA 16	4.92	0.0294	1.20

The results (both performance and computation) of three systems (CUBM-FA, FA and GMM-UBM) are compared in table 1. It shows that both factor analysis and the CUBM-FA can improve the performance significantly compared to GMM-UBM. In CUBM-FA system, the mixture number of CUBM can affect the performance and computation directly. The results indicate that CUBM with 16 mixtures achieved more than 45% relative reductions in EER while having similar computation compared to GMM-UBM. Furthermore, CUBM with 2

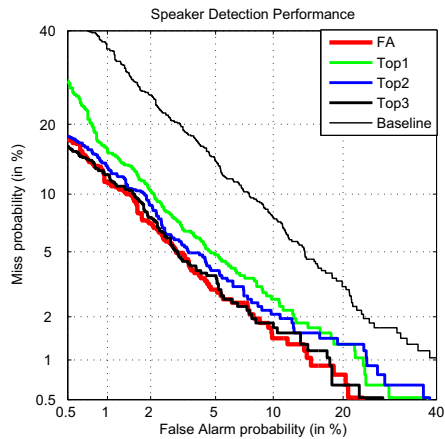


Figure 4: Performance of Top-N based Gaussian selection.

mixtures also speed up the estimation several times over FA baseline system with slight increase in EER.

Table 2: Performance of Top-N based Gaussian selection.

	EER(%)	MinDCF	Cost Time(secs)
GMM-UBM	8.94	0.0433	-
FA baseline	3.76	0.0208	9.53
CUBM-FA 16(top 3)	3.89	0.0210	1.24

We perform Top-N based selection experiments to improve the accuracy of choosing UBM Gaussians and solve the problem of performance reduction for the growth of CUBM components. we also set the number of CUBM Gaussians to 16 for all systems. Figure 4 shows the performance of Top-N based selection with different N . The results indicate that the performance of CUBM-FA(the black line) is close to the baseline factor analysis(the red line) with the selection of Top-3. From table 2 we also found that the efficiency of CUBM-FA is much better than baseline factor analysis (the spent time has reduced from 9.53 secs to 1.24 secs) while having almost the same performance (both around 3.8% in EER).

6. Conclusions

In this paper, we proposed a fast implementation of factor analysis based speaker verification by a GMM components pre-selection which well balances between the efficiency and performance, we use the cluster UBM as Gaussians selector, therefore, for each frame, only the observed top-N Gaussians of CUBM are used to estimate its EM statistics. this CUBM-FA approach is successfully applied in NIST SRE 2006 database. we can speed up the estimation of speaker model about 8 times over baseline factor analysis system with the similar performance.

7. Acknowledgements

This work was supported by the innovation foundation for graduate student of USTC (University of Science and Technology of China),No. KD2008056.

8. References

- [1] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification", in ODYSSEY01, 2001, pp. 213C218.
- [2] D.A. Reynolds, "Channel robust speaker verification via feature mapping", in ICASSP-2003, 2003, pp. II: 53C56.
- [3] D.A. Reynolds and R.C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models", IEEE Transactions on Speech and Audio Processing, vol. 3, no.1, pp. 72C83, 1995.
- [4] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems", Digital Signal Processing, vol. 10, pp. 42C54, 2000.
- [5] Kenny P., "Joint factor analysis of speaker and session variability: Theory and algorithms", Online: <http://www.crim.ca/perso/patrick.kenny/>
- [6] Kenny P., Boulianne G., Ouellet P., Dumouchel P., "Speaker and session variability in GMM-based speaker verification", IEEE Trans. Speech and Audio Proc., 15(4):1448-1460, 2007.
- [7] Xiang B. and Berger T., "Efficient text-independent speaker verification with structural Gaussian mixture models and neural network", IEEE Trans. Speech and Audio Proc., 11(5):447-456, 2003.
- [8] Bonastre J.-F., Scheffer N., Fredouille C., Matrouf D., "NIST'04 speaker recognition evaluation campaign: new LIA speaker detection platform based on ALIZE toolkit", in Proceedings of NIST Speaker Recognition Evaluation, Toledo, Spain, 2004.
- [9] The NIST Year 2006 Speaker Recognition Evaluation Plan, Online: http://www.nist.gov/speech/tests/spk/2006/sre-06_evalplan-v9.pdf