A Straightforward and Efficient Implementation of the Factor Analysis Model for Speaker Verification

Dris Matrouf1, Nicolas Scheffer1, Benoit Fauve2, Jean-François Bonastre1

1LIA, University of Avignon, France
2Speech and Image Research Group, University of Wales Swansea, UK

Abstract

For a few years, the problem of session variability in text-independent automatic speaker verification is being tackled actively. A new paradigm based on a factor analysis model have successfully been applied for this task. While very efficient, its implementation is demanding. In this paper, the algorithms involved in the eigenchannel MAP model are written down for a straightforward implementation, without referring to previous work or complex mathematics. In addition, a different compensation scheme is proposed where the standard GMM likelihood can be used without any modification to obtain good performance (even without the need of score normalization). The use of the compensated supervectors within a SVM classifier through a distance based kernel is also investigated. Experiments results shows an overall 50% relative gain over the standard GMM system on NIST SRE 2005 and 2006 protocols (both at the DCFmin and EER).

Index Terms: Speaker Verification, Session variability, Factor Analysis, GMM Supervectors.

1. Introduction

The use of Gaussian Mixture Models (GMM) in a GMM-UBM framework has been a standard in the speaker verification [1]. For a couple of years, new techniques that take session variability (or speaker intra-variability) into account have emerged, allowing to increase system performances drastically.

Among the approaches aiming at reducing the effect of session variability, feature mapping was often used alongside channel-labelled data, with the assumption of a discrete channel space. The novelty brought by the factor analysis model is that it assume the channel (or session) variability space to be continuous. In this model, the session variability effect is incorporated in the speaker model through session-dependent GMM mean supervectors offsets, constrained in a low dimensional subspace.

The work in this paper is based on a theoretical framework proposed by Kenny [2], and the reduced model in [3], called eigenchannel MAP. This paper has two main objectives. The first one is to propose a straightforward implementation of the factor analysis model. This model involves complex mathematics, and a lot of referral to other works if someone is willing to implement it. This paper can be seen as auto-sufficient in order to implement the whole approach. The second objective is to propose a new compensation scheme. In contrast to previous works, the presented strategy aims to subtract the session effect in both training and testing data, whereas others focus in compensating the target model with the test session effect.

Therefore, several advantages emerge: a highly performing system without the need of score normalization, a standard GMM likelihood function that can be applied without any modification and the possibility of the resulting supervectors to be use directly in a SVM classifier. This association between the factor analysis and SVM allows to benefit from the FA decomposition power and SVM classification power.

The paper is organized as follows: in section 2, the factor analysis decomposition (eigenchannel MAP estimation) used in this work is described. In section 3, estimation formulas and algorithms are given to allow a direct implementation of this model. In section 4, the adopted strategy for session compensation is detailed as well as the score computation process. Section 5 details the experimental protocol while section 6 presents the results of the approach.

2. Session variability modeling

A speaker model can be decomposed into three different components: a speaker-session-independent component, a speaker dependent component and a session dependent component. A GMM mean supervector is defined as the concatenation of the GMM component means. Let $D$ be the dimension of the feature space, the dimension of a supervector mean is $MD$ where $M$ is the number of Gaussian in the GMM. A speaker and session independent model is usually estimated in speaker verification to represent the inverse hypothesis: the UBM model. Let this model being parameterized by $\theta = \{m, \Sigma, \alpha\}$. In the following, $(h,s)$ will indicate the session $h$ of the speaker $s$. The factor analysis model, in our case the eigenchannel MAP estimator, can be written as:

$$m_{(h,s)} = m + D\gamma_s + UX_{(h,s)},$$

where $m_{(h,s)}$ is the session-speaker dependent supervector mean, $D$ is $MD \times MD$ diagonal matrix, $\gamma_s$ the speaker vector (a $MD$ vector), $U$ is the session variability matrix of low rank $R$ ($a MD \times R$ matrix) and $X_{(h,s)}$ are the channel factors, a $R$ vector (theoretically $x_{(h,s)}$ does not dependent on $s$). Both $\gamma_s$ and $x_{(h,s)}$ are normally distributed among $N(0, I)$. $D$ satisfies the following equation $I = D\Sigma D'$ where $\tau$ is the relevance factor required in the standard MAP adaptation ($DD'$ represents the $a priori$ covariance matrix of $\gamma_s$).

3. Straightforward implementation of the factor analysis model

The success of the factor analysis model relies on a good estimation of the $U$ matrix, thanks to a sufficiently high amount of data, where a high number of different recordings per speaker is available.

Notations: Let $A$ be a $MD \times K$ matrix formed by concatenating vertically $M$ matrices of dimensions $D \times K$. Let us denote $\{A\}^{g_k}$ the $g^{th}$ matrix in $A$ (usually corresponding to the $g^{th}$ component in the model).
3.1. General statistics

General statistics on the data have to be computed to estimate the latent variables and the $U$ matrix of equation 1. These are the zero order and first order statistics with respect to the UBM model. Let $N_s$ and $\mathbf{N}_{(h,s)}$ be vectors containing the zero order speaker-dependent and session-dependent statistics respectively (both of dimensions $M$ for a particular utterance),

$$\mathbf{N}_s[g] = \sum_{t \in s} \gamma_g(t); \quad \mathbf{N}_{(h,s)}[g] = \sum_{t \in (h,s)} \gamma_g(t),$$  

(2)

where $\gamma_g(t)$ is the a posteriori probability of Gaussian $g$ for the observation $t$. In the equation $\sum_{t \in s}$ means the sum over all frames belonging to the speaker $s$ and $\sum_{t \in (h,s)}$ means the sum over all frames belonging the session $h$ of speaker $s$. Let $\mathbf{X}_s$ and $\mathbf{X}_{(h,s)}$ be respectively vectors containing the first order speaker-dependent and session-dependent statistics. For an utterance, the dimensions of $\mathbf{X}_s$ and $\mathbf{X}_{(h,s)}$ are equal to $M \times D$:

$$\{\mathbf{X}_s\}[g] = \sum_{t \in s} \gamma_g(t) \cdot t; \quad \{\mathbf{X}_{(h,s)}\}[g] = \sum_{t \in (h,s)} \gamma_g(t) \cdot t,$$

(3)

3.2. Latent variables estimation

In the following, the estimation of the channel factors and the speaker vector are given. Let $\mathbf{X}_s$ and $\mathbf{X}_{(h,s)}$ be respectively speaker and channel dependent statistics defined as follows:

$$\{\mathbf{X}_s\}[g] = \{\mathbf{X}_s\}[g] - \sum_{h \in s} \mathbf{N}_{(h,s)}[g] \cdot \{\mathbf{Ux}_{(h,s)}\}[g],$$

$$\{\mathbf{X}_{(h,s)}\}[g] = \{\mathbf{X}_{(h,s)}\}[g] - \{\mathbf{m + Dy}_s\}[g] \cdot \sum_{h \in s} \mathbf{N}_{(h,s)}[g],$$

(4)

$\mathbf{X}_s$ is used to estimate the speaker vector (session effects are removed) while $\mathbf{X}_{(h,s)}$ is used to estimate channel factors (speaker effects are removed). Let $\mathbf{L}_{(h,s)}$ be $R \times R$ matrix, and $\mathbf{B}_{(h,s)}$ a vector of dimension $R$, both defined as:

$$\mathbf{L}_{(h,s)} = 1 + \sum_{g \in \text{UBM}} \mathbf{N}_{(h,s)}[g] \cdot \{\mathbf{U}\}_g^f \cdot \Sigma_g^{-1} \cdot \{\mathbf{U}\}_g,$$

$$\mathbf{B}_{(h,s)} = \sum_{g \in \text{UBM}} \{\mathbf{U}\}_g^f \cdot \Sigma_g^{-1} \cdot \{\mathbf{X}_{(h,s)}\}_g,$$

(5)

where $\Sigma_g$ is the covariance matrix of the $g^{th}$ UBM component. By using $\mathbf{L}_{(h,s)}$ and $\mathbf{B}_{(h,s)}$, $\mathbf{x}_{(h,s)}$ and $\mathbf{y}_s$ can be obtained by using the following equations:

$$\mathbf{x}_{(h,s)} = \mathbf{L}_{(h,s)}^{-1} \cdot \mathbf{B}_{(h,s)},$$

$$\{\mathbf{y}_s\}_g = \frac{\tau}{(\tau + \mathbf{N}_s[g])} \cdot \mathbf{D}_g \cdot \Sigma_g^{-1} \cdot \{\mathbf{X}_{(h,s)}\}_g,$$

(6)

where $\mathbf{D}_g = \Sigma_g^{1/2}$, $\tau$ is the MAP relevance factor ($14.0$ in our experiments).

3.3. Intersession matrix estimation

The matrix $\mathbf{U}$ can be estimated line by line, with $\{\mathbf{U}\}_g^f$ being the $g^{th}$ line of $\{\mathbf{U}\}_g$ then:

$$\{\mathbf{U}\}_g^f = \mathbf{L}_g^{-1} \cdot \mathbf{R}_g^{1},$$

(7)

\footnote{All posterior probabilities are computed on the UBM model}

Algorithm 1: Estimation algorithm of $\mathbf{U}$

For each speaker $s$ and session $h$:

1. $\mathbf{y}_s \leftarrow 0$, $\mathbf{x}_{(h,s)} \leftarrow 0$
2. $\mathbf{U} \leftarrow \text{random}$ ($\mathbf{U}$ is initialized randomly);
3. Estimate statistics: $\mathbf{N}_h, \mathbf{N}_{(h,s)}, \mathbf{X}_s, \mathbf{X}_{(h,s)}$ (eq.2 and 3);
4. for $i = 1$ to nb_iterations do
5. Center statistics: $\mathbf{X}_s, \mathbf{X}_{(h,s)}$ (eq.4);
6. Estimate $\mathbf{L}_{(h,s)}$, $\mathbf{B}_{(h,s)}$ (eq.5);
7. Estimate $\mathbf{x}_{(h,s)}$ and $\mathbf{y}_s$ (eq.6);
8. end
9. Estimate matrix $\mathbf{U}$ (eq. 7 and 8).

Algorithm 2: Estimation of latent variables on a single utterance

Result: Estimate $\mathbf{m}_{(h,s), t}$, $\mathbf{y}_s$, $\mathbf{x}_{(h,s), t}$

$\mathbf{y}_s \leftarrow 0$, $\mathbf{x}_{(h,s), t} \leftarrow 0$;
2. Estimate statistics: $\mathbf{N}_{h,s}, \mathbf{N}_{(h,s), t}, \mathbf{X}_{s,t}, \mathbf{X}_{(h,s), t}$ (eq.2 and 3);
3. for $i = 1$ to nb_iterations do
4. Center statistics: $\mathbf{X}_{s,t}, \mathbf{X}_{(h,s), t}$ (eq.4);
5. Estimate $\mathbf{L}_{(h,s), t, s}$ and $\mathbf{B}_{(h,s), t}$ (eq.5);
6. Estimate $\mathbf{x}_{(h,s), t}$ and $\mathbf{y}_{s,t}$ (eq.6);
7. end

where $\mathbf{R}_u^p$ and $\mathbf{L}_u^p$ are given by:

$$\mathbf{L}_u = \sum_{s \in h} (\mathbf{L}_{(h,s)}^{-1} + \mathbf{x}_{(h,s)}\mathbf{x}_{(h,s)}^T) \cdot \mathbf{N}_{(h,s)}[g]$$

$$\mathbf{R}_u^p = \sum_{s \in h} (\mathbf{X}_{(h,s)}[g])^T \cdot \mathbf{X}_{(h,s)}[g]$$

(8)

The algorithm 1 presents the adopted strategy to estimate the session variability matrix with the above developments (the standard likelihood function can be used to assess the convergence), while the algorithm 2 refers to the case of the estimation of $\mathbf{x}$ and $\mathbf{y}$ for a single utterance (a single iteration is performed in practice). The majority of the computational time is spent in Cholesky decompositions of $R \times R$ matrices in eq. 6 and 7. This process can however be parallelized.

4. Hybrid domain session compensation for the verification task

The following paragraph details the strategy employed to perform the session variability compensation. The verification task is defined as follows. A speaker $s_{tar}$ is enrolled by the system with his training data $Y_{s_{tar}}$. Given a sequence of speech frames $Y = \{y_1, \ldots, y_T\}$ and the speaker $s_{tar}$, the speaker verification task consists in determining if $Y$ was spoken by $s_{tar}$ or not. Using the factor analysis decomposition in both training and testing data, one can write:

$$\mathbf{m}_{(h_{tar}, s_{tar})} = \mathbf{m} + \mathbf{D}_{s_{tar}} + \mathbf{U}_{s_{tar}}$$

$$\mathbf{m}_{(h_{test}, s_{test})} = \mathbf{m} + \mathbf{D}_{s_{test}} + \mathbf{U}_{s_{test}}$$

(9)

where the speaker $s_{tar}$ in the training data and $s_{test}$ in the testing data have been distinguished. To deal with the session variability, the strategy adopted by [4, 3] assumes that the test speaker have the same identity as the target speaker, i.e. $y_{s_{test}} = y_{s_{tar}}$. The channel component $\mathbf{U}_{s_{test}}$ of the test
segment is estimated under this assumption. Indeed, the session component in the target model $U_{x_{tar}}$ is replaced by one estimated in the test data $U_{x_{test}}$. The world model in the score equation remains unchanged. This strategy has several drawbacks: the target speaker model is changing for each test and performance are good when using score normalization techniques.

In this work, a hybrid domain normalization strategy is proposed, aiming to withdraw the session component in the test and training data, that is:

$$m_{x_{tar}} = m + Dy_{x_{tar}}, \quad m_{x_{test}} = m + Dy_{x_{test}}.$$  

(10)

In this strategy, speakers are assumed to be different and are treated separately. The following paragraphs explain how to achieve this strategy in a system, when using a LLR-based verification approach or a kernel distance-based verification approach.

4.1. LLR based scoring

The speaker verification score is an expected log-likelihood ratio:

$$LLK(Y|m_{x_{tar}}, m_{x_{test}})) - LLK(Y|m),$$  

(11)

where $LLK(\cdot | \cdot )$ indicate the average of the log-likelihood function over all frames. Here, GMMs have the same covariance matrices as well as the same mixture weights (both dropped from the equation for clarity). Two session compensation approaches can be adopted: the first one is to perform compensation at the frame level, where session compensation can be seen as a front-end, the second one that we propose is an hybrid compensation, that is the session variability is subtracted from the target speaker model (model domain) and the compensation in the testing data is performed at the frame level (feature domain).

The following formula is used to remove the session effect for each frame $t$ (also successfully used by [5]):

$$\tilde{t} = t - \sum_{g=1}^{M} \gamma_g(t) \cdot \{U \cdot x_{x_{test}}\}|g|.$$  

(12)

4.2. Kernel based scoring and SVM modeling

By using equation 10, the factor analysis model estimates super-vectors containing only speaker information, normalized with respect to the session variability. In [6], the authors proposed a probabilistic distance kernel that computes a distance between GMMs, well suited for a SVM classifier. Let $X_s$ and $X_{s'}$ be two sequences of speech data corresponding to speakers $s$ and $s'$, the kernel formulation is given below:

$$K(X_s, X_{s'}) = \sum_{g=1}^{M} \left( \sqrt{\Sigma_g} \Sigma_g^{- \frac{1}{2}} m_s \right) \cdot \left( \sqrt{\Sigma_g} \Sigma_g^{- \frac{1}{2}} m_{s'} \right).$$  

(13)

This kernel is valid when only means of GMM models are varying (weights and covariance are taken from the world model). $m_s$ is taken here from the model in eq. 10, i.e. $m_s = m + Dy_s$.

5. Protocol

All experiments were performed using the ALIZE and LIA_SpkDet toolkit\(^2\).[7]

5.1. Database, protocols, toolkits

Speaker verification experiments, presented in section 6, are performed based upon the NIST SRE 2005 as a development set, and 2006 database for the validation set, male speakers only (referred to as 2005 and 2006 protocol). The 2005 protocol consists of 274 speakers, 9012 tests (951 target tests, the rest is impostor trials) while the 2006 protocol consists of 354 speakers, 9720 tests (741 target tests, the rest is impostor trials)\(^3\). Results are given in terms of equal-error-rate (EER) and the minimum of DCF (an a posteriori decision). Train and test utterances contain 2.5 minutes of speech in average (telephone conversation, where around 30% of speech frames per speaker have been retained). The inter-session variability matrix is enrolled on the NIST-SRE-2004 database with 2938 examples with 124 speakers (around 20 iterations to reach convergence). From the same database, 200 impostors speakers are used for score normalization and negative examples in the SVM classifier.

5.2. SVM training

The LIA_SpkDet toolkit now benefits from the LIBSVM [8] library to induce SVM and to classify instances. SVM models are trained with an infinite (very large in practice) C parameter thus avoiding classification error on the training data (hard margin behaviour). The negative labelled examples are speakers from the normalisation cohort.

5.3. Baseline GMM-UBM system

The baseline system is a standard GMM-UBM system [9]. The background model in the experiments is the same as the background model in the LIA submission in the NIST-SRE-2006 campaign (male set only)\(^4\). Training is performed based upon Fisher database\(^5\), and consists of about 10 millions of speech frames. Speaker models are derived by Bayesian adaptation on the Gaussian component means, with a relevance factor of 14.

Frames are composed of 19 LFCC parameters, its derivatives, and 11 second derivatives (the frequency window is restricted to 300-3400 Hz). A normalization process is applied, so that the distribution of each cepstral coefficient is 0-mean and 1-variance for a given utterance. The background model has 512 components whose variance parameters are floored to 50% of the global variance (0.5). It is worth noting that feature warping pre-processing was not necessary to obtain good performance. Table 1 shows the results of the baseline system and T-norm scores (Z and ZT-norm do not bring any improvement).

<table>
<thead>
<tr>
<th></th>
<th>SRE-05</th>
<th>SRE-06</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DC/Min</td>
<td>EER</td>
</tr>
<tr>
<td>Nonorm</td>
<td>3.83</td>
<td>7.15</td>
</tr>
<tr>
<td>Thorm</td>
<td>3.05</td>
<td>8.52</td>
</tr>
</tbody>
</table>

Table 1: Results of the baseline GMM-UBM system on the 2005 and 2006 protocol. DCFmin (x100), EER(%).

6. Experimental results

In the following is detailed the experimental results obtained with the implementation given in this paper.

Table 2 investigates the effect of the session variability subspace rank, without any score normalization. The best performing system have been found to have a rank of 40. It is

\(^2\)An open-source software available at http://www lia.univ-avignon.fr/heberges/ALIZE/

\(^3\)2005 protocol corresponds to the core condition, labeled as det\(^7\) and the 2006 protocol corresponds to the core condition, labeled as det\(^8\)


Table 2: Varying the rank subspace. 2005 Protocol. DCFmin(x100), EER(%). The best rank in terms of DCFmin is 40.

<table>
<thead>
<tr>
<th>Subspace rank</th>
<th>DCFmin (x100)</th>
<th>EER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>4.42</td>
<td>1.54</td>
</tr>
<tr>
<td>20</td>
<td>3.83</td>
<td>1.93</td>
</tr>
<tr>
<td>30</td>
<td>2.05</td>
<td>1.95</td>
</tr>
<tr>
<td>40</td>
<td>1.83</td>
<td>1.99</td>
</tr>
</tbody>
</table>

Table 3: Score normalization techniques on the factor analysis model (rank=40). Znorm always brings an improvement over the baseline. DCFmin(x100), EER(%)|

<table>
<thead>
<tr>
<th>Technique</th>
<th>DCFmin</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nonorm</td>
<td>1.83</td>
<td>2.97</td>
</tr>
<tr>
<td>Tnorm</td>
<td>1.72</td>
<td>2.15</td>
</tr>
<tr>
<td>ZTnorm</td>
<td>1.64</td>
<td>2.53</td>
</tr>
<tr>
<td>Znorm</td>
<td>1.84</td>
<td>2.83</td>
</tr>
</tbody>
</table>

Table 4: GMM factor analysis supervectors with distance kernel (rank=40). 2005 and 2006 protocol. DCFmin(x100), EER(%). Tnorm brings the most significant gain.

<table>
<thead>
<tr>
<th>Technique</th>
<th>DCFmin</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (40)</td>
<td>1.97</td>
<td>4.83</td>
</tr>
<tr>
<td>Znorm</td>
<td>1.92</td>
<td>5.36</td>
</tr>
<tr>
<td>Tnorm</td>
<td>1.61</td>
<td>4.42</td>
</tr>
<tr>
<td>ZTnorm</td>
<td>1.58</td>
<td>4.51</td>
</tr>
</tbody>
</table>

8. References


