I-MATRIX FOR TEXT-INDEPENDENT SPEAKER RECOGNITION

Liang He, Jia Liu

Tsinghua National Laboratory for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

ABSTRACT

This paper proposes an i-matrix for text-independent speaker recognition. The framework of the proposed i-matrix is similar to an i-vector. However, the presented method takes short-time cepstral feature matrices as inputs to explore both cepstral feature distribution and temporal information for the recognition task in the phase of statistical modeling. In the i-matrix, the variability of an utterance is constrained by two subspaces U and V, which are estimated by an iterative method on a large database. When U and V are well built, each utterance is represented by an i-matrix. Decision function is a cosine kernel. Experiments were carried out on the tel-tel-English condition of NIST SRE 2008 core task. Compared with an i-vector-LDA, the average EER and MDCF of an i-matrix-LDA showed a relative decrease of 4.82% and 5.12% respectively.

Index Terms— I-vector, i-matrix, Gaussian mixture models, text-independent speaker recognition

1. INTRODUCTION

After front-end processing and feature extraction [1], the core task of text-independent speaker recognition becomes a comparison of two cepstral feature sequences. The cepstral feature sequence can be seen as a temporal sequence with a concept, i.e. speaker ID. And for a given temporal sequence, our task is to identify the speaker ID .

Most recent state-of-art statistical modeling methods, such as joint factor analysis (JFA) [1, 2, 3] and i-vector (ivec) [4], originates from Gaussian mixture model-universal background model (GMM-UBM) [5]. The GMM-UBM has two implicit assumptions. One is that each speaker is characterized by a unique probability density function (PDF). The other is that temporal information is insignificant in the phase of statistical modeling [2], which simplifies the recognition problem. Under these two assumptions, each cepstral feature vector from a single speaker is assumed to be an instance generated independently from a speaker PDF. The first assumption is reasonable while the second assumption makes the use of dynamic short-time information impossible. The short-time information is proven to be beneficial in a text-independent speaker recognition system with high-level feature [6].

To explore both cepstral feature distribution and temporal information for the recognition task during the phase of statistical modeling, this paper presents an i-matrix. Different from the i-vector, the inputs of the i-matrix are short-time cepstral feature matrices. A UBM with weights, mean *matrices* and diagonal covariance matrices is built via an expectation-maximum (EM) algorithm. A left-hand subspace U and a right-hand subspace V are assumed to reduce the degree of freedom. The assumption, derivation and application of the i-matrix will be illustrated in detail in the following sections.

The remainder is as follows. Section 2 reviews the i-vector, section 3 proposes the i-matrix, and section 4 gives experimental results on the tel-tel-English condition of NIST SRE 2008 core task. Finally, a conclusion is summarized in section 5.

2. I-VECTOR

One shortage in the traditional GMM-UBM system is the high degree of freedom during enrollment and test phase. To reduce it, the i-vector assumes a total variability subspace T which constrains free parameters in a low dimensional subspace. And each utterance is represented by a subspace loading factor $\boldsymbol{w}$

$$\boldsymbol{\mu}_n = \boldsymbol{\mu}_{\text{ubm}} + T\boldsymbol{w}_n \tag{1}$$

where $\boldsymbol{\mu}_{\text{ubm}}$ denotes the mean supervector of a UBM and subindex n denotes the n-th utterance, $1 \leq n \leq N$. N is the number of training utterances.

The total variability subspace T and covariance matrix Σ are estimated by maximizing the summation of log likelihood function on a very large training database [1, 3]. Once T and

This work was supported by the China Postdoctoral Science Foundation under Grant No. 2012M510448, and in part by the National Natural Science Foundation of China under Grant No. 61005019, 61105017 and 90920302.

[1] In this paper, the extracted feature includes both basic cepstral feature and its derivatives, i.e. MFCC + delta, double delta + feature warping.

[2] In the phase of feature extraction, temporal information is modeled by taking delta and double delta operations. But in the phase of statistical modeling, we take a extracted cepstral feature vector, including basic feature and its derivatives, as a static point. Here, the temporal information means the dynamic information among several cepstral feature vectors. It is ignored in a traditional GMM-UBM system.

Σ are fixed, $\boldsymbol{w}$ is computed by a variational Bayesian estimation [7].

3. I-MATRIX

The i-vector takes cepstral vectors as the input, which can't take advantage of temporal information in the phase of statistical modeling. Temporal information, often studied in the high-level system, such as pronunciation, phonotactics and prosody, is proven to be effective. Here, the i-matrix is designed to take not only cepstral feature distribution information but also temporal information for the recognition task.

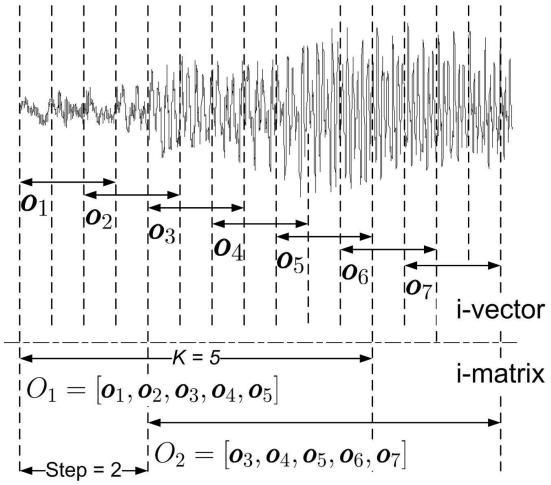

Fig. 1. Comparison of cepstral feature inputs between i-vector and i-matrix

3.1. Assumption

To begin with, we rewrite the GMM formula with matrices as inputs

$$f(O) = \sum_{i=1}^{M} \frac{\omega_i}{(2\pi)^{\frac{F\times K}{2}} |\text{diag}(\text{vec}(\Sigma_i))|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \| \left(O-\mu_i\right)^t .* \Sigma_i.^{\wedge -1} .* \left(O-\mu_i\right) \|_F\right) \quad (2)$$

where F denotes the dimension of cepstral feature, K denotes the frame number of adjacent cepstral features, O is a $F \times K$ matrix (short-time cepstral feature sequence). M denotes the Gaussian mixture number, ω_i is the i-th weight, μ_i is the i-th $F \times K$ mean matrix, Σ_i is a $F \times K$ *diagonal* covariance matrix [3], $.^{\wedge}$ and $.*$ are element-by-element power and product respectively [4], and $\| \cdot \|_F$ is the Frobenius norm. vec$(\cdot)$ is an operation which converts a matrix into a supervector by stacking each column and diag$(\cdot)$ is an operation which converts a vector into a diagonal matrix.

The i-matrix assumes two variability subspaces U ($MF \times R_U$, R_U is the rank of U) and V ($MK \times R_V$, R_V is the rank of V). For n-th utterance and i-th mixture, the model assumption is

$$\mu_{n,i} = \mu_{\text{ubm},i} + U_i X_n V_i^t \quad (3)$$

with the i-matrix X_n ($R_U \times R_V$). The U is similar to the role of T in the i-vector and the V is the subspace which captures temporal information.

3.2. Derivation

Our goal is to maximize the summation of log-likelihood over an auxiliary training data corpus

$$\arg_\theta \max \sum_{n=1}^{N} \sum_{t_n=1}^{T_n} \log \sum_{i=1}^{M} f(O_{t_n}|\theta_i) \quad (4)$$

where T_n is the duration of the n-th utterance. For clarity, we use $\theta_i = \{\omega_i, \mu_i, \Sigma_i, U_i, V_i, X_n\}$ to denote the parameter set of the i-th component.

For a O_{t_n}, we have

$$\begin{aligned}
&\log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta_i)\right] \\
=&\log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta'_i)\frac{f(O_{t_n}|\theta_i)}{f(O_{t_n}|\theta'_i)}\right] \\
&-\log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta'_i)\right] + \log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta'_i)\right] \\
=&\log\left[\sum_{i=1}^{M} \frac{f(O_{t_n}|\theta'_i)}{\sum_{j=1}^{M} f(O_{t_n}|\theta'_j)}\frac{f(O_{t_n}|\theta_i)}{f(O_{t_n}|\theta'_i)}\right] + \log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta'_i)\right] \\
\geq&\sum_{i=1}^{M} \frac{f(O_{t_n}|\theta'_i)}{\sum_{j=1}^{M} f(O_{t_n}|\theta'_j)} \log\frac{f(O_{t_n}|\theta_i)}{f(O_{t_n}|\theta'_i)} + \log\left[\sum_{i=1}^{M} f(O_{t_n}|\theta'_i)\right]
\end{aligned} \quad (5)$$

where θ' is a known parameter set. Note that

$$\sum_{i=1}^{M} \frac{f(O_{t_n}|\theta'_i)}{\sum_{j=1}^{M} f(O_{t_n}|\theta'_j)} = 1, \quad (6)$$

the inequality of above equation holds for the convex property of logarithmic function. Thus, we just need to select θ to satisfy

$$\arg_\theta \max \sum_{i=1}^{M} \frac{f(O_{t_n}|\theta'_i)}{\sum_{j=1}^{M} f(O_{t_n}|\theta'_j)} \log f(O_{t_n}|\theta_i) + \text{const} \quad (7)$$

[3] Although the form of Σ_i is a full matrix, the physical meaning is a diagonal matrix.

[4] They are borrowed from Matlab.

We take a special case $\{\omega_{\text{ubm},i}, \boldsymbol{\mu}_{\text{ubm},i}, \Sigma_{\text{ubm},i}, U_i, V_i, X_n = 0\}$ as θ' and have

$$\begin{aligned}
&\sum_{n=1}^{N}\sum_{t_n=1}^{T}\sum_{i=1}^{M}\gamma_{\text{ubm},i}(O_{t_n})\log f(O_{t_n}|\theta_i) \\
&=\sum_{n=1}^{N}\sum_{t_n=1}^{T_n}\sum_{i=1}^{M}\gamma_{\text{ubm},i}(O_{t_n})\Bigg[\log\frac{\omega_i}{(2\pi)^{\frac{F\times K}{2}}|\text{diag}(\text{vec}(\Sigma_i))|^{\frac{1}{2}}} \\
&-\frac{1}{2}\|\left(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t\right).*\Sigma_i.^{\wedge-1}.* \\
&\left(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t\right)\|_F\Bigg]
\end{aligned} \tag{8}$$

where

$$\gamma_{\text{ubm},i}(O_{t_n}) = \frac{f(O_{t_n}|\theta_{\text{ubm},i})}{\sum_{j=1}^{M} f(O_{t_n}|\theta_{\text{ubm},j})} \tag{9}$$

To simplify the above equation, we define some symbols

$$\begin{aligned}
Z_{n,i} &= \sum_{t_n=1}^{T_n}\gamma_{\text{ubm},i}(O_{t_n}) \\
F_{n,i} &= \sum_{t_n=1}^{T_n}\gamma_{\text{ubm},i}(O_{t_n})(O_{t_n}-\mu_{\text{ubm},i}).*\Sigma_i.^{\wedge-\frac{1}{2}} \\
S_{n,i} &= \sum_{t_n=1}^{T_n}\gamma_{\text{ubm},i}(O_{t_n})\left[(O_{t_n}-\mu_{\text{ubm},i}).*\Sigma_i.^{\wedge-\frac{1}{2}}\right] \\
&\quad\left[(O_{t_n}-\mu_{\text{ubm},i}).*\Sigma_i.^{\wedge-\frac{1}{2}}\right]^t
\end{aligned} \tag{10}$$

and

$$U_iX_nV_i^t.*\Sigma_i.^{\wedge-\frac{1}{2}} = \bar{U}_i\bar{X}_n\bar{V}_i^t \tag{11}$$

So the last term of Equation (8) is written as follows

$$\begin{aligned}
&\sum_{n=1}^{N}\sum_{t_n=1}^{T}\sum_{i=1}^{M}\gamma_{\text{ubm},i}(O_{t_n})\|\left(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t\right).* \\
&\quad\Sigma_i.^{\wedge-1}.*\left(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t\right)\|_F \\
&=\sum_{n=1}^{N}\sum_{i=1}^{M}\text{tr}\Big\{S_{n,i}-2F_{n,i}(\bar{U}_i\bar{X}_n\bar{V}_i^t)^t+\bar{U}_i\bar{X}_n\bar{V}_i^t(\bar{U}_i\bar{X}_n\bar{V}_i^t)^t\Big\}
\end{aligned} \tag{12}$$

There is no analytical solution for the above equation and we turn to an iterative method which is similar to iterative NAP [8]. The parameters are optimized by repeating three steps:

Step 1. When $\bar{V}_i$ and $\bar{U}_i$ are fixed, $\bar{X}_n$ is solved by taking derivation with respect to $\bar{X}_n$

$$\sum_{i=1}^{M} Z_i(\bar{U}_i^t\bar{U}_i)\bar{X}_n(\bar{V}_i^t\bar{V}_i) = \sum_{i=1}^{M}\bar{U}_i^tF_i\bar{V}_i \tag{13}$$

Let $A_i = \bar{U}_i^t\bar{U}_i$, $B_i = (\bar{V}_i^t\bar{V}_i)$, and $C = \sum_{i=1}^{M}\bar{U}_i^tF_i\bar{V}_i$, the above equation is

$$\begin{aligned}
&\sum_{i=1}^{M} Z_iA_i\bar{X}_nB_i = C \\
&\left[\sum_{i=1}^{M} Z_i(B_i\otimes A_i)\right]\text{vec}(\bar{X}_n) = \text{vec}(C) \\
&\text{vec}(\bar{X}_n) = \left[\sum_{i=1}^{M} Z_i(B_i\otimes A_i)\right]^{-1}\text{vec}(C)
\end{aligned}, \tag{14}$$

where $[\cdot]^{-1}$ is the pseudoinverse and $\otimes$ is Kronecker product.

Step 2. When $\bar{V}_i$ and $\bar{X}_n$ are fixed, $\bar{U}_i$ is solved as follows

$$\bar{U}_i = \frac{1}{Z_i}\sum_{n=1}^{N}(F_i\bar{V}_i\bar{X}_n^t)\left[\sum_{n=1}^{N}(\bar{X}_n\bar{V}_i^t\bar{V}_i\bar{X}_n^t)\right]^{-1} \tag{15}$$

Step 3. When $\bar{U}_i$ and $\bar{X}_n$ are fixed, $\bar{V}_i$ is solved as follows

$$\bar{V}_i = \frac{1}{Z_i}\sum_{n=1}^{N}(F_i\bar{U}_i\bar{X}_n^t)\left[\sum_{n=1}^{N}(\bar{X}_n\bar{U}_i^t\bar{U}_i\bar{X}_n^t)\right]^{-1} \tag{16}$$

3.3. Application

From the above derivation, we present the realization of the i-matrix in this subsection. The estimation procedure of $\bar{U}_i$, $\bar{V}_i$ and Σ_i are given in Table 1. Once the global parameters are well estimated, an utterance is represented by an i-matrix with Equation (14). The decision function is the cosine kernel function

$$\frac{\text{vec}(\bar{X}_a)^t\text{vec}(\bar{X}_b)}{\sqrt{\text{vec}(X_a)^t\text{vec}(X_a)}\sqrt{\text{vec}(X_b)^t\text{vec}(X_b)}} \tag{17}$$

where subindexes a and b denote two arbitrary utterances.

3.4. Discussion

1. Different from the standard procedure of an i-vector, the Σ_i in the i-matrix is estimated by collecting statistics rather than solving an equation, because there is no analytic expression for it. The computation of Σ_i is as follows

$$\begin{aligned}
\Sigma_i = &\frac{1}{\sum_{n=1}^{N}\sum_{t_n=1}^{T}\gamma_{\text{ubm},i}(O_{t_n})}\Bigg[\sum_{n=1}^{N}\sum_{t_n=1}^{T}\gamma_{\text{ubm},i}(O_{t_n}) \\
&(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t).*(O_{t_n}-\mu_{\text{ubm},i}-U_iX_nV_i^t)\Bigg]
\end{aligned} \tag{18}$$

2. The first term of Equation (8) is not optimized for the difficulty of solving equation. However, once the Equation (12) is minimized, the first term of Equation (8) is reduced accordingly in a loose sense. This is also an awkward thing in the i-matrix.

Table 1. Estimation procedure of $\bar{X}_n$, $\bar{U}_i$, $\bar{V}_i$ and Σ_i

Initialization: $\bar{U}_i$ and $\bar{V}_i$ are randomly initialized.
Procedure:
1. Estimate Σ_i on the training database with $X_n = 0$.
2. Collect statistics $Z_{n,i}$, $F_{n,i}$ and $S_{n,i}$ with Equation (10).
3. Compute $\bar{X}_n$ with Equation (14).
4. Update $\bar{U}_i$ with Equation (15).
5. Estimate Σ_i with updated $\bar{U}_i$.
6. Collect statistics $Z_{n,i}$, $F_{n,i}$ and $S_{n,i}$ with Equation (10).
7. Compute $\bar{X}_n$ with Equation (14).
8. Update $\bar{V}_i$ with Equation (16).
9. Estimate Σ_i with updated $\bar{V}_i$.
10. Goto Step 2 if not terminated.
Termination: $3 \sim 6$ iterations.

3. We solve $\bar{U}_i$, $\bar{V}_i$ and $\bar{X}_n$ out rather than U_i, V_i and X_n.
4. The training procedure of the UBM in an i-matrix is the same as the standard procedure [5].
5. We make no prior assumption of X_n. The presented method is a non Bayesian method.
6. We can also convert a short-time cepstral feature matrix O_{t_n} into a vector vec(O_{t_n}) and build an i-vector system accordingly. However, this method violates the intrinsic structure of O_{t_n} and introduces lots of nuisance information by the vec$(\cdot)$ operation. Besides, it brings more computation burden.

4. EXPERIMENTS

4.1. Databases

Experiments were carried out on the common condition 7 of NIST SRE 2008 core task (c7-08). The NIST SRE 2008 core task is named short2-short3. There are 8 common conditions. The c7-08 task is the telephone-telephone-English (tel-tel-English) condition, containing 1265 models, 1567 test segments and 17761 trials. We used previous NIST evaluation data and Switchboard corpus (SWB) to estimate our system parameters. Table 2 summarized the data we used [5].

Table 2. Data corpus for the UBM, T, V,U,Σ and LDA

	SWB	SRE04	SRE05	SRE06
UBM	×	×		
$\bar{V}$,$\bar{U}$,Σ	×	×	×	×
LDA		×	×	×

[5] We didn't use any normalization, e.g. ztnorm, snorm.

4.2. Configuration

Speech/silence segmentation was performed by a G.723.1 VAD detector. A 13-dimensional MFCC was extracted, with appended delta and acceleration coefficients. 39-dimensional vectors were subjected to feature warping. UBMs with 1024 Gaussian components were gender-dependent. The rank of T, U and V were 600, 300 and 50 respectively. The number of adjacent features was $K = 11$, the step was 3 and the iteration number was 5. Length normalization was applied. The decision function was the cosine kernel. The EER (Equal error rate) and MDCF (Minimal detection cost function defined by NIST SRE 2008) were adopted as performance measurements.

4.3. Results and Analysis

Table 3 presents the experimental results on the c7-08 task. From the table, we conclude that the i-matrix outperforms the i-vector. The average EER and MDCF are reduced by 5.60% and 7.87% respectively when cosine kernel is applied directly. And the average EER and MDCF are reduced by 4.82% and 5.12% respectively when linear discriminant analysis (LDA) is followed [9]. We attribute the performance improvement to the use of temporal information.

Table 3. Experimental results on the c7-08 task

c7-08	Female		Male	
	EER(%)	MDCF	EER(%)	MDCF
ivec	7.20	0.299	6.38	0.278
imat	6.80	0.267	6.02	0.264
ivec-LDA	3.73	0.160	3.15	0.130
imat-LDA	3.56	0.151	2.99	0.124

5. CONCLUSION

In this paper, we present an i-matrix to address text-independent speaker recognition problem. The i-matrix, which is based on short-time cepstral feature sequence, takes advantage of both cepstral feature distribution and temporal information. This is the main reason for the system performance improvement. The derivation of the presented i-matrix is from the convex of logarithmic function and the parameters are estimated by an iterative method. A simple experiment was carried out on the common condition 7 of NIST SRE 2008 core task. Experimental results demonstrate the effectiveness of this novel approach. Further work involves the selection of K, analysis of computation burden and proof of convergence.

6. REFERENCES

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

[2] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 4, no. 4, pp. 1435–1447, may 2007.

[3] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 13, no. 3 , pp. 345–354, may 2005.

[4] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, may 2011.

[5] 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, jan 2000.

[6] W.M. Campbell, J.P. Campbell, T.P. Gleason, D.A. Reynolds, and Wade Shen, "Speaker verification using support vector machines and high-level features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2085–2094, sep 2007.

[7] P. Kenny, D. Reynolds, and F. Castaldo, "Diarization of telephone conversations using factor analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 6, pp. 1059–1070, dec 2010.

[8] W.M. Campbell, Z.N. Karam, and D.E. Sturim, "Inner product discriminant functions," *Advances in Neural Information Processing Systems 22, Cambridge, MA, 2009, MIT Press*, 2009.

[9] P.N. Belhumeur, J.P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, jul 1997.