JOINT BAYESIAN GAUSSIAN DISCRIMINANT ANALYSIS FOR SPEAKER VERIFICATION

Yiyan Wang, Haotian Xu, Zhijian Ou

Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University, Beijing, China
wangyiya14@mails.tsinghua.edu.cn, xht13@mails.tsinghua.edu.cn, ozj@tsinghua.edu.cn

ABSTRACT

State-of-the-art i-vector based speaker verification relies on variants of Probabilistic Linear Discriminant Analysis (PLDA) for discriminant analysis. We are mainly motivated by the recent work of the joint Bayesian (JB) method, which is originally proposed for discriminant analysis in face verification. We apply JB to speaker verification and make three contributions beyond the original JB. 1) In contrast to the EM iterations with approximated statistics in the original JB, the EM iterations with exact statistics are employed and give better performance. 2) We propose to do simultaneous diagonalization (SD) of the within-class and between-class covariance matrices to achieve efficient testing, which has broader application scope than the SVD-based efficient testing method in the original JB. 3) We scrutinize similarities and differences between various Gaussian PLDAs and JB, complementing the previous analysis of comparing JB only with Prince-Elder PLDA. Extensive experiments are conducted on NIST SRE10 core condition 5, empirically validating the superiority of JB with faster convergence rate and $9-13\%$ EER reduction compared with state-of-the-art PLDA.

Index Terms— Speaker recognition, Joint Bayesian, PLDA

1. INTRODUCTION

The state-of-the-art in speaker recognition is currently dominated by the i-vector approach, that models both speaker and channel variabilities in a single low-dimensional space termed the total variability subspace [1]. An i-vector based speaker verification system mainly consists of three components, which are i-vector extractor based on Gaussian Mixture Models (GMMs) or Deep Neural Networks (DNNs) [2], i-vector post-processing (e.g. length normalization) and discriminant analysis. Note that this approach defers the decomposition of speaker and intersession variabilities to the stage of discriminant analysis, which is particularly important for this approach.

An attractive discriminant analysis technique is to construct likelihood ratio score based on probabilistic generative models such as the widely used Probabilistic Linear Discriminant Analysis (PLDA) [3] with many variants. Except the heavy-tailed PLDA [4], most variants are Gaussian, such as Prince-Elder PLDA [3], Simplified PLDA (SPLDA) [5], Kaldi PLDA [6], two-covariance model [7], and Ioffe PLDA [8]. Basically, Gaussian PLDA assumes that the j-th i-vector from speaker i obeys the following decomposition [1] :

$$x_{ij} = Fz_i + \epsilon_{ij}$$

where the latent speaker factor z_i and the intersession residual ϵ_{ij} are both Guassians and independently distributed. F is the loading matrix spanning the speaker subspace. Denote by H_I the intra-personal hypothesis that one set of i-vectors x_1 and another set of i-vectors x_2 belong to the same speaker, and H_E the extra-personal hypothesis that they are from different speakers. The speaker verification problem can then be solved by thresholding the likelihood ratio score $p(x_1, x_2|H_I)/p(x_1, x_2|H_E)$.

The performance of PLDA largely depends on how it can be trained effectively to learn the within-class variability, which characterizes intersession residuals, and the between-class variability, which characterizes differences among speakers. Some improvements include using data domain adaptation of PLDA parameters [5] and discriminative training [9]. In this paper, we are primarily concerned with addressing the two basic challenging issues for the current Gaussian PLDA family. First, for PLDAs with subspace modeling, it is difficult to determine the subspace dimension which is crucial for performance. Low subspace dimension often leads to under-fitting, while high subspace dimension results in over-fitting. Second, whether using subspace modeling or not, current PLDAs suffer from slow convergence of their implemented EM iterations. As analyzed in[10], different parameterizations and selections of hidden variables in designing EM updates have significant effect on the convergence performance. These basic issues hinder improving the performance of current PLDAs.

We are mainly motivated by recent the work of Joint Bayesian (JB) method [10], which was originally proposed for face verification. In JB, there is no need to determine the subspace dimension, and it achieved faster convergence and more accuracy in [10] for face verification. We apply JB to speaker verification and make three main contributions. 1) We find that the EM updates with approximated statistics suggested in [10] does not work in speaker verification problem. Instead, the EM iterations with exact statistics are employed and give better performance. 2) Inspired by Fisher LDA, we propose to do simultaneous diagonalization (SD) of the within-class and between-class covariance matrices to achieve efficient testing. Compared to the SVD-based efficient testing method in [10], the new SD method can still be applied to reduce the testing complexity even in the case that the number of training samples per subject are different. 3) We scrutinize similarities and differences between various Gaussian PLDAs and JB, complementing the analysis of comparing JB only with Prince-Elder PLDA in [10]. Moreover, extensive experiments are conducted on NIST SRE10 core condition 5, empirically validating the superiority of JB in term of EM convergence rate and EER performance.

2. JOINT BAYESIAN GAUSSIAN DISCRIMINANT ANALYSIS

Joint Bayesian (JB) Gaussian discriminant analysis was first proposed in [10] for face verification. Its model formulation is similar to the two-covariance model [11] but with different parameterizations and selections of hidden variables in EM training. For speaker

This work is supported by NSFC grant 61473168.

[1] Throughout the paper, we assume the data have zero-mean after a standard centering preprocessing step and omit the global mean in the model.

verification, we use two independent Gaussians to represent speaker identity and intersession residuals respectively. The j-th i-vector of speaker i, denoted by $x_{ij} \in R^d$, is decomposed as:

$$x_{ij} = \mu_i + \varepsilon_{ij}$$

where $\mu_i \sim \mathcal{N}(0, S_\mu)$ is the speaker identity variable, $\varepsilon_{ij} \sim \mathcal{N}(0, S_\varepsilon)$ models the within-speaker variability. The model parameters are $\Theta = \{S_\mu, S_\varepsilon\}$. The extracted m_i i-vectors for speaker i are denoted by $x_i = [x_{i1}; \ldots; x_{im_i}]$. The total hidden variables are stacked as $h_i = [\mu_i; \varepsilon_{i1}; \ldots; \varepsilon_{im_i}]$, which are Gaussian distributed with block diagonal covariance matrix $\Sigma_{h_i} = diag(S_\mu, S_\varepsilon, \ldots, S_\varepsilon)$.

The data likelihood for observed x_i is

$$p(x_i) = \mathcal{N}(0, \Sigma_{x_i}), \Sigma_{x_i} = \begin{bmatrix} S_\mu + S_\varepsilon & S_\mu & \ldots & S_\mu \\ S_\mu & S_\mu + S_\varepsilon & \ldots & S_\mu \\ \vdots & \vdots & \ddots & \vdots \\ S_\mu & S_\mu & S_\mu & S_\mu + S_\varepsilon \end{bmatrix} \quad (1)$$

where the dimension of Σ_{x_i} is $m_i \times d$.

The parameters Θ are estimated by the EM algorithm through iteratively optimizing the expected complete log-likelihood function as follows:

$$\max_{\Theta} \sum_i E_{p(h_i|x_i;\Theta^t)}[logp(h_i; \Theta^{t+1})] \quad (2)$$

where $\Theta^t = \{S^t_\mu, S^t_\varepsilon\}$ are the parameters from the t-th EM update, and Θ^{t+1} the parameters to be updated in the $(t+1)$-th iteration. Under this auxiliary objective function, the terms related to S_μ and S_ϵ are effectively decoupled, resulting in very elegant update equations for S_μ and S_ϵ [10].

In speaker verification testing, we calculate the log-likelihood ratio score to determine whether one set of i-vectors x_1 (including m_1 i-vectors) and another set of i-vectors x_2 (including m_2 i-vectors) are from the same speaker [2] :

$$r(x_1, x_2) = logp(x_1, x_2) - logp(x_1) - logp(x_2) \quad (3)$$

Note that all the data likelihoods $p(x_1, x_2)$, $p(x_1)$ and $p(x_2)$ can be calculated through Eq. 1, which involves matrix inversions.

To accelerate testing, [10] employed SVD to obtain low rank approximations of the matrices appearing in the three log-likelihood terms in Eq. 3, which depend on $m_1 + m_2$, m_1 and m_2 respectively. Therefore, this speedup is more useful under the condition that the number of i-vectors is the same across all subjects, i.e. $m_i = m$. This is often satisfied in the task of face verification and face search.

Here we propose first to do simultaneous diagonalization (SD) of S_μ and S_ε, $\Phi^T S_\mu \Phi = K$ and $\Phi^T S_\epsilon \Phi = I$. Similar to Fisher LDA, we keep the first $s < d$ largest eigenvalues of $S_\mu^{-1} S_\varepsilon$, giving the low-rank diagonal matrix K. Denote by Φ the corresponding low-rank eigenvector matrix. By defining $\Psi = \Phi^{-T}$, we have $S_\mu = \Psi K \Psi^T$, $S_\epsilon = \Psi\Psi^T$, and moreover,

$$\Sigma_{x_i} = \Omega \begin{bmatrix} K + I & K & \ldots & K \\ K & K + I & \ldots & K \\ \vdots & \vdots & \ddots & \vdots \\ K & K & K & K + I \end{bmatrix} \Omega^T$$

where $\Omega = diag(\Psi; \ldots; \Psi)$. Based on this decomposition of Σ_{x_i}, the calculation of data likelihood $p(x_i)$ could be accelerated, if we take Ω to transform the i-vectors via pre-computation. The likelihood calculation then only involves inversion of diagonal matrices, reducing the complexity from $\mathcal{O}(d^3)$ to $\mathcal{O}(d)$. Moreover, it can be seen that this speedup does not depend on m_i and thus has broader applicability. In this paper, we also conduct experiments to compare these two speedup methods over speaker verification accuracy.

3. CONNECTION WITH PLDA

In this section, we investigate the connections between joint Bayesian (JB) [10], Simplified PLDA (SPLDA) [5] and Kaldi-PLDA [12]. We mainly show that different parameterization and selection of hidden variables lead to different behavior of the EM algorithm, and JB is superior to PLDAs in terms of EM convergence. For the advantages of JB in allowing the data to implicitly determine the subspace dimensionality for maximal discrimination and favoring low-rank esimates of S_μ and S_ε, the reader could refer to [10]. Table 1 summarizes the similarities and differences between JB, SPLDA, Kaldi PLDA and the two-covariance model.

3.1. Simplified PLDA (SPLDA)

Basically, SPLDA [5] assumes that j-th i-vector from speaker i obeys the following decomposition :

$$x_{ij} = F z_i + \epsilon_{ij} \quad (4)$$

where the latent speaker factor $z_i \sim \mathcal{N}(0, I)$ and the intersession residual $\epsilon_{ij} \sim \mathcal{N}(0, \Lambda)$ are both Guassians and independently distributed. F is the loading matrix spanning the speaker subspace. The speaker subspace could be full rank, which is also known as the two-covariance model [11].

The parameters $\Theta = \{F, \Lambda\}$ [5] are estimated by the EM algorithm through iteratively optimizing the complete log-likelihood $logp(x_i, z_i; \Theta_{t+1})$ averaged over $p(z_i|x_i; \Theta_t)$ where $\Theta_t = \{\Lambda_t, F_t\}$

$$\max_{\Theta} \sum_i E_{p(z_i|x_i;\Theta_t)}[logp(x_i, z_i; \Theta_{t+1})] \quad (5)$$

Different from Eq. 2 in JB, the hidden variables in SPLDA are only z_i's, excluding ϵ_{ij}'s.[3] Now we analyze the convergence property of the EM updates for SPLDA, analogous to [10]. Note that maximizing Eq. 5 over F_{t+1} is equivalent to minimizing over F_{t+1}

$$\sum_i \sum_j trace(\Lambda_{t+1}^{-1} E[(x_{ij} - F_{t+1} z_i)(x_{ij} - F_{t+1} z_i)^T])$$

It can be seen that

$$\begin{aligned} & E[(x_{ij} - F_{t+1} z_i)(x_{ij} - F_{t+1} z_i)^T] = \\ & (x_{ij} - F_{t+1} E[z_i])(x_{ij} - F_{t+1} E[z_i])^T \\ & + F_{t+1}(I - F_t^T (F_t F_t^T + \Lambda_t)^{-1} F_t) F_{t+1}^T \end{aligned}$$

where

$$E[z_i] = F_t^T (F_t F_t^T + \Lambda_t)^{-1} x_{ij}$$

$$E[z_i z_i^T] - E[z_i] E[z_i]^T = I - F_t^T (F_t F_t^T + \Lambda_t)^{-1} F_t$$

When Λ_t is small, by setting F_{t+1} as F_t, we find that

$$x_{ij} - F_{t+1} E[z_i] \approx x_{ij} - F_{t+1} F_t^T (F_t F_t^T)^{-1} x_{ij} = 0$$

[2] By abuse of notation, here x_1 is not the data corresponding to speaker 1.

[3] Including all hidden variables to derive the EM update for SPLDA is ill-posed under SPLDA's parameterization.

Method	JB	two-covariance	SPLDA	Kaldi PLDA
Observation	$x_i = \{x_{ij}, j = 1, \ldots, m_i\}$			$\bar{x}_i = \frac{1}{m_i}\sum_{j=1}^{m_i} x_{ij}$
Model	$x_{ij} = \mu_i + \varepsilon_{ij}$		$x_{ij} = Fz_i + \varepsilon_{ij}$	$\bar{x}_i = \mu_i + \varepsilon_{i1}$
h_i	$\{\mu_i, \{\varepsilon_{ij}\}\}$	$\{\mu_i\}$	$\{z_i\}$	$\{\mu_i, \varepsilon_{i1}\}$
EM objective function $Q(\Theta_t, \Theta_{t+1})$	$E_{p(h_i\|x_i)}[logp(h_i)]$	$E_{p(h_i\|x_i)}[logp(x_i, h_i)]$		$E_{p(h_i\|\bar{x}_i)}[logp(h_i)]$
Subspace dimensionality setting	loose		strict	loose
EM convergence	fast	slow		fast

Table 1. The summary of the similarities and difference between JB, SPLDA, Kaldi PLDA and the two-covariance model. x_{ij} denotes the j-th i-vector of speaker i. $\mu_i \sim \mathcal{N}(0, S_\mu)$ is the identity variable for speaker i, modeled by the between-class covariance S_μ. $\varepsilon_{ij} \sim \mathcal{N}(0, S_\epsilon)$ is the intersession residual, modeled by the within-class covariance S_ϵ. For SPLDA, $z_i \sim \mathcal{N}(0, I)$ stands for the identity variable.

$$F_{t+1}(I - F_t^T(F_tF_t^T + \Lambda_t)^{-1}F_t)F_{t+1}^T \approx$$
$$F_{t+1}(I - F_t^T(F_tF_t^T)^{-1}F_t)F_{t+1}^T = 0$$

Hence updating F_{t+1} as F_t approximately optimize the M-step and the EM-algorithm stalls upon a single iteration.

Note that theoretically the EM algorithm is only actually guaranteed to produce non-decreasing optimization of data likelihood through a series of parameter updates. Strict convergence to local minima (or stationary points) requires further strong assumptions. Combining this understanding of the EM algorithm and the above analysis of halt upon a single iteration, we could realize that the EM update for SPLDA could be easily stuck into a non-local minimum with small Λ_t. The EM update for JB does not have such problem, since JB has different parameterization and selection of hidden variables. The faster convergence of the EM iterations for JB is also empirically observed in our experiments.

3.2. Kalid PLDA

The Kaldi is a widely used open-source speech recognition toolkit [12]. Here we examine the PLDA implementation in Kaldi code repository [6]. The conceptual starting point for Kaldi PLDA is the SPLDA model as shown in Eq. 4 with full rank F. Next, Kaldi PLDA is only concerned with modeling the average i-vector for each speaker $\bar{x}_i = \sum_{j=1}^{m_i} x_{ij}/m_i$, which is distributed according to

$$p(\bar{x}_i) = \mathcal{N}(0, FF^T + \frac{1}{m_i}\Lambda) \qquad (6)$$

where m_i is the numbers of extracted i-vectors for speaker i.

Eq. 6 is then treated as the data likelihood function. All the extracted i-vectors for each speaker are collapsed as a single sample - the average i-vector, which is assumed to obey the decomposition :

$$\bar{x}_i = \mu_i + \bar{\varepsilon}_i$$

where $\mu_i \sim \mathcal{N}(0, \Gamma)$ models the between-class variability with the covariance $\Gamma = FF^T$ and the average residual $\bar{\varepsilon}_i = \sum_{j=1}^{m_i} \varepsilon_{ij}/m_i \sim \mathcal{N}(0, \frac{1}{m_i}\Lambda)$ models the within-class variability.

The expected complete log-likelihood function for the EM algorithm is optimized to iteratively estimate Γ and Λ, as follows:

$$\max_{\Theta} \sum_i E_{p(\mu_i, \varepsilon_{i1}|x_i;\Theta_t)}[logp(\mu_i, \varepsilon_{i1}; \Theta_{t+1})]$$

The parameterization of Kaldi PLDA is similar to JB, i.e. using two covariance matrices. Hence the EM iterations in Kaldi PLDA can also select the total hidden variables $(\mu_i, \varepsilon_{i1})$, with good convergence. However, the additive decomposition only applies to the average i-vector in Kaldi PLDA. This is helpful for estimating between-class covariance but is detrimental for estimating within-class covariance.

At the testing phase, Kaldi PLDA also performs simultaneous diagonalization of Λ and Γ. However, the significance of computational saving is less than the SD applied in JB, because JB calculates the joint likelihood of a number of i-vectors while Kaldi only calculates the likelihood of a single average i-vector.

4. EXPERIMENTS

4.1. Dataset

We conduct speaker verification experiments with different discriminant analysis techniques on the NIST SRE10 core condition 5, which includes 11982 speakers, 7169 target and 408950 nontarget trials [13]. The DNN used in the experiments is trained on part of the Fisher data including about 600 hours of speech cuts. The i-vector extractor training data comprises 57517 speech cuts of 5767 speakers, which are from Switchboard, Fisher and NIST SRE 04, 06, and 08. Both JB and SPLDA are trained on SRE data, consisting of 36612 speech cuts and 3805 speakers from NIST SRE 04, 06, and 08.

	Fisher	Switchboard	SRE	duration (hours)
DNN-UBM	√			600
i-vector extractor	√	√	√	1890
SPLDA			√	1250
JB			√	1250

Table 2. The data used to train the DNN-UBM, i-vector extractor, SPLDA and JB for speaker verification.

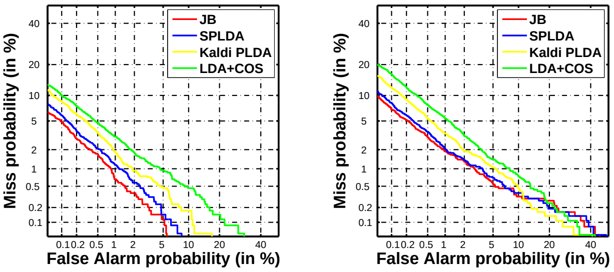
4.2. System Configuration

The features used in the experiments are 40-dimensional Mel Frequency Cepstral Coefficients (MFCCs), including 20-dimensional static features and first-order derivatives. The speaker verification system uses a DNN-UBM with 5 hidden layers, 5335 senones, and a 600-dimensional i-vector extractor. The input of the DNN-UBM is the MFCCs extracted using 21 frames (11 frames before and 9 frames after). For discriminant analysis techniques, we implement three references namely LDA+COS, SPLDA and Kaldi PLDA. We apply the LDA to the i-vectors to obtain 200-dimension features and use cosine distance metric for testing. For SPLDA, we set the dimension of the subspace to 300. For Kaldi PLDA, we use the default configuration of the Kaldi toolkit [6]. The system performances are reported by equal error rate (EER) and minimum decision cost function (DCF) defined in NIST SRE08 and SRE10 [13].

4.3. Results

4.3.1. Speaker Verification Performance

We evaluate LDA+COS, Kaldi PLDA, SPLDA and JB on the NIST SRE10 core condition 5. These four models share the same training and test data. Fig. 1 illustrates the detection error trade-off (DET) curves of LDA+COS, SPLDA, Kaldi PLDA and JB for discriminant analysis with configurations described in Section 4.2. From the results show in Tab. 3, we can conclude that :



(a) SRE10 MALE (b) SRE10 FEMALE

Fig. 1. DET curves for JB, SPLDA, Kaldi PLDA and LDA in SRE10 core condition 5 evaluation.

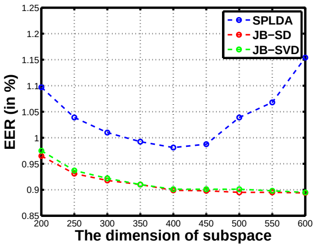


Fig. 2. The influence of subspace dimensionality on JB and SPLDA using NIST SRE10 core condition male test data.

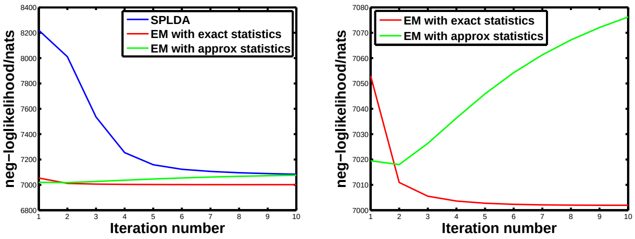
- Compared to distance-based discriminant analysis LDA+COS, probabilistic generative model based methods such SPLDA, Kaldi PLDA and JB achieve better performance on EER.
- In terms of EER, JB improves 13.0% and 45.3% compared to SPLDA and Kaldi PLDA respectively on SRE10 male tests, 9.2% and 30.9% on female tests. This verifies that JB with careful selection of hidden variables achieves better parameter estimation due to efficient EM updates.
- SPLDA achieves better results on EER than Kaldi PLDA, because SPLDA utilizes the joint likelihood of i-vectors rather than the single average i-vector as used in Kaldi PLDA to estimate the parameters.

4.3.2. Subspace Dimensionality

Here we investigate the impact of the sub-space dimensionality of JB and SPLDA for discriminant analysis, which is shown in Fig. 2. Two methods (SVD, SD) are used to reduce the subspace dimensionality of JB in speaker verification testing. It can be seen that: 1) The dimension of the subspace plays an important role in SPLDA that lower may cause under-fitting, while higher may cause over-fitting. 2) The JB performance fluctuates slightly with the change of subspace dimension in testing, but the time complexity reduces from $O(d^3)$ to $O(s^3)$. Loose parameterization for JB makes it more robust since the dimension of the subspace is automatically fitted via data rather than manual defined. 3) SVD and SD have close performances but SD has wider applicability.

4.3.3. Convergence Rate

As discussed before, EM iterations for SPLDA are easier to stall in a single iteration. Fig. 3 shows the neg-loglikelihood curve of SPLDA with the optimal subspace dimensionality and that of JB trained by EM with exact and approximated statistics. From Fig. 3, first we find that JB converges faster than SPLDA with better parameter estimation. Second, we find that JB trained by EM with approximated statistics proposed by [10] will diverge in Fig 3b, while JB trained EM with exact statistics converges nicely.



(a) (b)

Fig. 3. (a) The negative log-likelihood of JB (EM with exact or approximated statistics) and SPLDA during training. (b) The zoom-in of negative log-likelihood convergence curves for JB with exact and approximated EM statistics.

System	SRE10 MALE			SRE10 FEMALE		
	EER	DCF10	DCF08	EER	DCF10	DCF08
LDA+COS	1.905	0.292	0.091	2.619	0.399	0.126
Kaldi PLDA	1.299	0.284	0.079	1.944	0.345	0.102
SPLDA	1.010	0.217	0.055	1.621	0.287	0.079
JB	**0.894**	**0.188**	**0.048**	**1.485**	**0.245**	**0.069**

Table 3. Performance comparison of four different discriminant analysis back-ends on NIST SRE10 core condition 5.

EER	SRE10 MALE	SRE10 FEMALE
JB	0.894	1.485
JB-SVD (dim=403)	0.901	1.513
JB-SD (dim=407)	0.899	1.510
SPLDA (dim=403)	0.981	1.674
SPLDA (dim=407)	0.981	1.673

Table 4. The effect of dimensionality reduction for JB and PLDA. The dimension of subspace for JB-SVD is determined by $dim = rank(A) = 403$ (A is defined in [10]) and the dimension of the subspace for JB-SD is determined by $dim = rank(S_\mu) = 407$.

Tab. 4 shows the differences on EER between SPLDA and JB with or without dimension reduction. It is observed that even with the same dimension of the subspace learned by JB, SPLDA is still worse than JB. This justifies our analysis that model formulation and hidden variable selection of JB leads to better parameter estimation than SPLDA.

5. CONCLUSIONS

In this paper, we propose to apply JB to model i-vectors with careful parameterization and hidden variable selection that benefits EM iterations. Both theoretical derivation and experiments conducted on the NIST SRE10 core condition demonstrate that: 1) the parameterization of JB enables it to learn the intrinsic dimensionality of the identify subspace, which can reduce the system complexity without performance degradation; 2) Hidden variables selection of JB makes EM iterations converge faster with better parameter estimation; 3) The EM with exact statistics performs better than with approximated statistics. For future work, it is interesting to apply data domain adaption [5] and feature compensation [14, 15] and nearest-neighbor discriminant analysis (NDA) [16][17] that have been successfully applied to PLDA to JB to further improve performance.

6. REFERENCES

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

[2] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell M-cLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014.

[3] Simon J.D. Prince and James H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," *IEEE 11th International Conference on Computer Vision*, 2007.

[4] Patrick Kenny, "Bayesian speaker verification with heavy-tailed priors," *Odyssey*, 2010.

[5] Daniel Garcia-Romero and Alan McCree, "Supervised domain adaptation for i-vector based speaker recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014.

[6] "https://github.com/kaldi-asr/kaldi," .

[7] Niko Brümmer and Edward De Villiers, "The speaker partitioning problem," *Odyssey*, 2010.

[8] Sergey Ioffe, "Probabilistic linear discriminant analysis," *European Conference on Computer Vision*, 2006.

[9] Pierre-Michel Bousquet and Jean-Francois Bonastre, "Constrained discriminative speaker verification specific to normalized i-vectors," *Odyssey*, 2016.

[10] Dong Chen, Xudong Cao, David Wipf, Fang Wen, and Jian Sun, "An efficient joint formulation for Bayesian face verification," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 39, pp. 32–46, 2016.

[11] Sandro Cumani, Niko Brümmer, Lukáš Burget, Pietro Laface, Oldřich Plchot, and Vasileios Vasilakakis, "Pairwise discriminative speaker verification in the i-vector space," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 6, pp. 1217–1227, 2013.

[12] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[13] Alvin F. Martin and Craig S. Greenberg, "The NIST 2010 s-peaker recognition evaluation," *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[14] Fred Richardson, Brian Nemsick, and Douglas Reynolds, "Channel compensation for speaker recognition using MAP adapted PLDA and denoising DNNs," *Odyssey*, 2016.

[15] Sandro Cumani and Pietro Laface, "I–vector transformation and scaling for PLDA based speaker recognition," *Odyssey*, 2016.

[16] Seyed Omid Sadjadi, Sriram Ganapathy, and Jason W Pelecanos, "The IBM 2016 speaker recognition system," *arXiv preprint arXiv:1602.07291*, 2016.

[17] K. Fukunaga and J.M. Mantock, "Nonparametric discriminant analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 6, pp. 671–678, 1983.