Interspeech 2018
2-6 September 2018, Hyderabad

Improved Supervised Locality Preserving Projection for I-vector Based Speaker Verification

Lanhua You, Wu Guo, Yan Song, Sheng Zhang

National Engineering Laboratory of Speech and Language Information Processing,
University of Science and Technology of China, Hefei, China

lhyou@mail.ustc.edu.cn, {guowu, songy}@ustc.edu.cn, zs1234@mail.ustc.edu.cn

Abstract

A Supervised Locality Preserving Projection (SLPP) method is employed for channel compensation in an i-vector based speaker verification system. SLPP preserves more important local information by weighing both the within- and between-speaker nearby data pairs based on the similarity matrices. In this paper, we propose an improved SLPP (P-SLPP) to enhance the channel compensation ability. First, the conventional Euclidean distance in conventional SLPP is replaced with Probabilistic Linear Discriminant Analysis (PLDA) scores. Furthermore, the weight matrices of P-SLPP are generated by using the relative PLDA scores of within- and between-speaker pairs. Experiments are carried out on the five common conditions of NIST 2012 speaker recognition evaluation (SRE) core sets. The results show that SLPP and the proposed P-SLPP outperform all other state-of-the-art channel compensation methods. Among these methods, P-SLPP achieves the best performance.

Index Terms: speaker verification, supervised locality preserving projection, probabilistic linear discriminant analysis

1. Introduction

Speaker verification (SV) is used to verify a person's claimed identity from voice characteristics. In recent years, i-vector [1] based speaker verification systems have become very popular because of their good performance and ability to compensate for within-speaker variations. An i-vector is generated by projecting the Gaussian mixture model (GMM) mean shifted supervector onto a low-rank total variability (TV) subspace while retaining the speaker identity. This way, an i-vector can be viewed as a front-end for further modeling.

As the i-vectors are based on one total variability space that contains speaker and channel variability information, compensation techniques are required to limit the effects of channel variability in the i-vector speaker representations. Here, linear discriminant analysis (LDA) [2] is a standard channel compensation approach. The main objective of LDA is to describe the differences between the groups in terms of canonical variants, which are linear combinations of the original variables. Currently, locally weighted LDA (LWLDA) [3] and non-parametric discriminant analysis (NDA) [4] are also employed in the speaker recognition field, and they can alleviate some of the limitations identified for LDA, where the underlying distribution of classes is supposed to be Gaussian and unimodal. In combination with the well-known PLDA [5] backend, the i-vector/PLDA framework dominates the research field of text-independent speaker recognition.

In this paper, we ask the following question: Is LDA, NDA or LWLDA optimal for i-vector based speaker verification? The answer is often neglected. The Gaussian distribution, which is the premise of LDA, cannot be guaranteed, particularly when speech recordings are collected in the presence of noise and channel distortions. To cope with the above noted issue, NDA measures the scatter matrices on a local basis using the K-nearest neighbor (K-NN) rule. However, a neglected problem is that the intrinsic geometry of the data is destroyed after LDA or NDA projection. On the other hand, LWLDA only weighs the within-speaker i-vectors, and the local information of the between-speaker data is neglected. These problems make these methods open to improvement.

In this paper, we explore supervised locality preserving projection (SLPP) [6] for channel variability compensation. Different from the above algorithms, SLPP can perform embedding that preserves more important local information in the data based on the similarity matrices. We evaluate SLPP against LDA, NDA and LWLDA on the five common conditions of NIST 2012 SRE core sets [7]. Experiments show that SLPP outperforms LDA, NDA and LWLDA. Furthermore, a novel PLDA-based SLPP (P-SLPP) algorithm is proposed to learn the speaker identity subspace, where the PLDA scores are integrated into the objective function of the SLPP. In the P-SLPP algorithm, we modify the weight matrices of SLPP with the relative PLDA scores of within- and between-speaker pairs to enhance channel compensation ability. As far as we know, this is the first combination of the channel compensation algorithm and the backend classification algorithm in a speaker verification task. Experiments show that the proposed P-SLPP offers further improvement over conventional SLPP and achieves the best performance among all the abovementioned algorithms.

The remainder of this paper is organized as follows. Section 2 gives a brief review of LDA, NDA and LWLDA. Section 3 presents the application of SLPP for speaker verification, as well as the proposed P-SLPP algorithm. In addition, we interpret the relationships among SLPP, LDA and their variants. Section 4 presents the experimental setup and the results of this study. In Section 5, we summarize our work and discuss future work.

2. Dimensionality reduction methods

2.1. LDA

LDA computes an optimum linear projection to obtain a more discriminative and lower-dimensional representation of the i-vector. It is obtained by maximizing the between-speaker scatter while simultaneously minimizing the within-speaker scatter [2]. Let $\mathbf{S}_w$ and $\mathbf{S}_b$ be the within- and between-speaker scatter matrices.

$$\mathbf{S}_w = \sum_{s=1}^{S} \sum_{i=1}^{n_s} (\mathbf{x}_i^s - \overline{\mathbf{x}}_s)(\mathbf{x}_i^s - \overline{\mathbf{x}}_s)^T, \tag{1}$$

10.21437/Interspeech.2018-41

$$\mathbf{S}_b = \sum_{s=1}^{S} n_s(\overline{\mathbf{x}}_s - \overline{\mathbf{x}})(\overline{\mathbf{x}}_s - \overline{\mathbf{x}})^T, \quad (2)$$

where S is the total number of speakers, n_s is the number of i-vectors corresponding to the s^{th} speaker, $\mathbf{x}_i^s$ is the i^{th} i-vector belonging to speaker s, and $\overline{\mathbf{x}}_s$ and $\overline{\mathbf{x}}$ are the global mean i-vectors for speaker s and all speakers, respectively.

Moreover, as described in [8], the above scatter matrices can further be written in a pair-wise manner:

$$\mathbf{S}_w = \frac{1}{2}\sum_{i,j=1}^{n} W_{ij}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T, \quad (3)$$

$$\mathbf{S}_b = \frac{1}{2}\sum_{i,j=1}^{n} B_{ij}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T, \quad (4)$$

where W_{ij} and B_{ij}, which respectively determine the influence of the scatter of different i-vector pairs on within- and between-speaker scatter matrices, are given by

$$W_{ij} = \begin{cases} \frac{1}{n_s} & z_i = z_j = s \\ 0 & z_i \neq z_j \end{cases}, \quad (5)$$

$$B_{ij} = \begin{cases} \frac{1}{n} - \frac{1}{n_s} & z_i = z_j = s \\ \frac{1}{n} & z_i \neq z_j \end{cases}, \quad (6)$$

where n is the total number of i-vectors and z_i and z_j are the speaker class labels.

After calculating the scatter matrices, the LDA linear projection $\mathbf{A} \in \mathbb{R}^{d\times r}$ $(d > r)$ is defined as:

$$\mathbf{A} = \arg\max_{\mathbf{A}}[\mathrm{tr}((\mathbf{A}^T\mathbf{S}_w\mathbf{A})^{-1}\mathbf{A}^T\mathbf{S}_b\mathbf{A})]. \quad (7)$$

This optimization problem has an analytical solution in which the r columns of $\mathbf{A}$ are the eigenvectors corresponding to the largest eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$ [2]. Through the optimized projection matrix $\mathbf{A}$, the transformed i-vector $\mathbf{y}$ of the input $\mathbf{x}$ can be computed as follows:

$$\mathbf{y} = \mathbf{A}^T\mathbf{x}. \quad (8)$$

2.2. NDA

The NDA approach is the same as LDA except that the global mean i-vectors in (1) and (2) are replaced with local sample averages computed based on the K-NN of individual samples [4]. In more detail, the scatter matrix $\mathbf{S}_b$ is given by

$$\mathbf{S}_b = \sum_{s=1}^{S}\sum_{\substack{l=1\\ l\neq s}}^{S}\sum_{i=1}^{n_s} w_i^{s,l}(\mathbf{x}_i^s - \mathbf{M}_i^{s,l})(\mathbf{x}_i^s - \mathbf{M}_i^{s,l})^T, \quad (9)$$

where $w_i^{s,l}$ is the weighing function, while $\mathbf{M}_i^{s,l}$ is the local mean of K-NN samples for $\mathbf{x}_i^s$ from class l.

$\mathbf{S}_w$ is obtained in a similar fashion to $\mathbf{S}_b$ except that the weighting function is set to 1 and local gradients are computed within each class. After both $\mathbf{S}_w$ and $\mathbf{S}_b$ are obtained, $\mathbf{A}$ can be computed as in LDA.

2.3. LWLDA

LWLDA, which is a localized variant of LDA, computes $\mathbf{S}_w$ and $\mathbf{S}_b$ in the same way as in (3) and (4), except that the weight matrices are calculated as

$$W_{ij} = \begin{cases} \frac{H_{ij}}{n_s} & z_i = z_j = s \\ 0 & z_i \neq z_j \end{cases}, \quad (10)$$

$$B_{ij} = \begin{cases} H_{ij}(\frac{1}{n} - \frac{1}{n_s}) & z_i = z_j = s \\ \frac{1}{n} & z_i \neq z_j \end{cases}, \quad (11)$$

where the affinity matrix $\mathbf{H}$ weighs the within-speaker i-vectors to preserve the complex structure of the within-speaker data. It can be chosen by a Gaussian function that varies with the local density of data samples [3].

Once $\mathbf{S}_w$ and $\mathbf{S}_b$ are computed, the transformed matrix is formed by calculating the eigenvectors of $\mathbf{S}_w^{-1}\mathbf{S}_b$.

3. Supervised locality preserving projection

3.1. SLPP

The conventional LPP [9] uses the K-NN rule to preserve the neighborhood structure of the data set; however, it is an unsupervised method, and hence, the class information is neglected. To overcome this issue, a supervised LPP (SLPP) was proposed by Shen et al. in [6]. This SLPP algorithm constructs both within- and between-speaker K nearest neighbor graphs through which two weight matrices are defined. Based on the within- and between-speaker weight matrices, SLPP attempts to ensure that the nearby within-class data pairs are kept closer, while the nearby between-class data pairs are mapped farther apart. That is different from LWLDA, which only focuses on preserving the local information of the within-speaker i-vectors. The local information of both within- and between-class data are preserved in the SLPP method. Moreover, SLPP only weighs the data pairs which are connected in the neighbor graph so that the unimportant pairs of i-vectors are not considered for optimization.

For more details, given a set of i-vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n]$, the objective function of SLPP is defined as follows:

$$\max \frac{\sum_{ij}(y_i - y_j)^2 B_{ij}}{\sum_{ij}(y_i - y_j)^2 W_{ij}}, \quad (12)$$

where $y_i = \mathbf{a}^T\mathbf{x}_i$ denotes the i^{th} sample after projection, and $\mathbf{a}$ is a transformation vector that needs to be estimated; $\mathbf{B}$ and $\mathbf{W}$, as described in detail later, are symmetrical between- and within-speaker weight matrices, respectively. Through some algebraic derivation [6], the objective function in Eq. (12) can be written as

$$\max \frac{\mathbf{a}^T\mathbf{X}\mathbf{L}^B\mathbf{X}^T\mathbf{a}}{\mathbf{a}^T\mathbf{X}\mathbf{L}^W\mathbf{X}^T\mathbf{a}}, \quad (13)$$

where $\mathbf{L}^B = \mathbf{D}^B - \mathbf{B}$ and $\mathbf{L}^W = \mathbf{D}^W - \mathbf{W}$ are the Laplacian matrices, and $\mathbf{D}$ is a diagonal matrix; $\mathbf{D}_{ii}^B = \sum_j \mathbf{B}_{ij}$, and $\mathbf{D}_{ii}^W = \sum_j \mathbf{W}_{ij}$.

The transformation vector $\mathbf{a}$ that maximizes the objective function is given by the largest eigenvalue solution to the generalized eigenvalue problem:

$$\mathbf{X}\mathbf{L}^B\mathbf{X}^T\mathbf{a} = \lambda\mathbf{X}\mathbf{L}^W\mathbf{X}^T\mathbf{a}. \quad (14)$$

When we map i-vectors into an r-dimensional subspace, the projection matrix $\mathbf{A}$ is composed of the r largest eigenvalues solution to Eq. (14).

3.1.1. Weight Matrices

As shown in objective function Eq. (12), the weight matrices have great influence on the SLPP method. The detailed calculation procedure of the weight matrices is stated as follows:

1) Construct the within-speaker connected graph: suppose that $\mathbf{x}_i$ and $\mathbf{x}_j$ are from the same speaker, $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected if $\mathbf{x}_i$ is among K-NN samples for $\mathbf{x}_j$, or $\mathbf{x}_j$ is among K-NN samples for $\mathbf{x}_i$.

2) Construct the between-speaker connected graph: suppose that $\mathbf{x}_i$ and $\mathbf{x}_j$ are from the different speakers, $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected if $\mathbf{x}_i$ is among K-NN samples for $\mathbf{x}_j$, or $\mathbf{x}_j$ is among K-NN samples for $\mathbf{x}_i$.

3) Compute the within-speaker weight matrix $\mathbf{W}$ based on the within-speaker connected graph:

$$W_{ij} = \begin{cases} \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\tau}) & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are connected} \\ 0 & \text{others} \end{cases} \tag{15}$$

4) Compute the between-speaker weight matrix $\mathbf{B}$ based on the between-speaker connected graph:

$$B_{ij} = \begin{cases} \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\tau}) & \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are connected} \\ 0 & \text{others} \end{cases} \tag{16}$$

where τ is a scaling factor.

As described in Eqs.(15) and (16), the weight matrices measure the similarity between a pair of the connected i-vectors by Euclidean distance. Based on the similarity matrices, SLPP learns a linear projection by making $\mathbf{y}_i$ and $\mathbf{y}_j$ closer if the within-speaker samples $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected, while making them farther part if $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected in the between-speaker graph.

3.2. P-SLPP

In the i-vector/PLDA-based speaker verification system, there is a great gap between the channel compensation algorithm and the final scoring method. In the final scoring step, the similarity score between the speaker model and the test utterance is measured by PLDA scoring, while the Euclidean distance is always adopted in calculating the weight matrices for the conventional SLPP method. Furthermore, the SLPP separately uses the within- and between-speaker similarity in Eqs.(15) and (16), which has a low discriminative ability.

Motivated by this, we replace the Euclidean distance with the PLDA score [10] in Eqs.(15) and (16). More specifically, we integrate the within- and between-speaker similarity into one equation. Through these two important modifications, the channel compensation method is more consistent with the final similarity calculation, and the i-vectors after channel compensation are more discriminative. Suppose $\mathbf{x}_{ik}^W$ and $\mathbf{x}_{ik}^B$ are the k^{th} $(1 \leq k \leq K)$ K-NN samples for $\mathbf{x}_i$ in within- and between-speaker data, respectively, and a relative PLDA score R_{ik} is defined as

$$R_{ik} = plda\_score(\mathbf{x}_i, \mathbf{x}_{ik}^B) - plda\_score(\mathbf{x}_i, \mathbf{x}_{ik}^W), \tag{17}$$

where the PLDA score between a pair of i-vectors is computed as

$$plda\_score = \mathbf{w}_i^T \mathbf{Q} \mathbf{w}_i + \mathbf{w}_j^T \mathbf{Q} \mathbf{w}_j + 2\mathbf{w}_i^T \mathbf{P} \mathbf{w}_j + const, \tag{18}$$

where $\mathbf{w}$ is a centralized i-vector $\mathbf{x}$ after length normalization and matrices $\mathbf{P}$ and $\mathbf{Q}$ are obtained after PLDA training [11].

By the sigmoid function with a scaling factor, we normalize R_{ik} between 0 and 1.

$$G_{ik} = \frac{1}{1 + e^{-R_{ik}/\tau}} \tag{19}$$

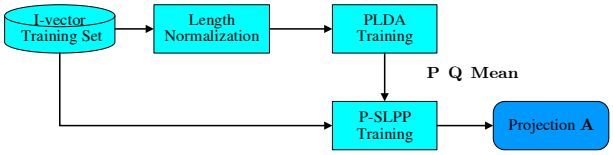

Figure 1: *The training procedure of P-SLPP.*

The normalized weight approaches 0.5 if the within- and between-speaker PLDA scores are equal (i.e., $R_{ik} = 0$). Next, the weight matrices are given by

$$W_{ij}^{'} = \begin{cases} G_{ik} & \text{if } \mathbf{x}_j \text{ is the } k^{th} \text{ within-speaker } K\text{-NN sample for } \mathbf{x}_i \\ 0 & \text{else} \end{cases} \tag{20}$$

$$B_{ij}^{'} = \begin{cases} G_{ik} & \text{if } \mathbf{x}_j \text{ is the } k^{th} \text{ between-speaker } K\text{-NN sample for } \mathbf{x}_i \\ 0 & \text{else} \end{cases} \tag{21}$$

Note that $\mathbf{W}^{'}$ and $\mathbf{B}^{'}$ are not symmetric. To solve this problem, the final weight matrices can be obtained as

$$W_{ij} = \max(W_{ij}^{'}, W_{ji}^{'}), \tag{22}$$

$$B_{ij} = \max(B_{ij}^{'}, B_{ji}^{'}). \tag{23}$$

After the weight matrices are computed, the remaining procedure of P-SLPP is similar to that of SLPP. Figure 1 shows the training procedure of P-SLPP, which could be divided into the following steps:

1) Obtain $\mathbf{P}$, $\mathbf{Q}$ and the mean i-vector through the PLDA offline training.

2) Compute the within- and between-speaker matrices using Eqs. (17)-(23).

3) Calculate the projection matrix $\mathbf{A}$ using Eq. (14).

The above P-SLPP effectively combines the channel compensation algorithm (SLPP) with the backend algorithm (PLDA), both of which adopt the PLDA scores as similarity measures. This preserves the strength of SLPP, which only weighs the connected data pairs. In addition, P-SLPP improves the conventional SLPP by giving a heavy penalty in the objection function if the PLDA score between $\mathbf{x}_i$ and $\mathbf{x}_{ik}^B$ is higher than that between $\mathbf{x}_i$ and $\mathbf{x}_{ik}^W$. Such a penalty mechanism takes both the within- and between-speaker similarity into account at the same time and better separates the between-speaker data pairs.

3.3. Relationship between SLPP and LDA

In this section, we can see that there is a close relationship between SLPP and LDA and that SLPP can also be viewed as a variant version of LDA. From Eq. (14), $\mathbf{X}\mathbf{L}^B\mathbf{X}^T$ can be reformulated as

$$\begin{aligned} \mathbf{X}\mathbf{L}^B\mathbf{X}^T &= \sum_{i=1}^{n} (\sum_{j=1}^{n} B_{ij}) \mathbf{x}_i \mathbf{x}_i^T - \sum_{i,j=1}^{n} B_{ij} \mathbf{x}_i \mathbf{x}_j^T \\ &= \frac{1}{2} \sum_{i,j=1}^{n} B_{ij} (\mathbf{x}_i \mathbf{x}_i^T - \mathbf{x}_i \mathbf{x}_j^T - \mathbf{x}_j \mathbf{x}_i^T + \mathbf{x}_j \mathbf{x}_j^T) \\ &= \mathbf{S}_b. \end{aligned} \tag{24}$$

Similarly, we have $\mathbf{X}\mathbf{L}^W\mathbf{X}^T = \mathbf{S}_w$; hence, the projection matrix $\mathbf{A}$ trained from SLPP is given by the eigenvectors corresponding to the r largest eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$ as well. From the above deduction, we can see that the only difference among LDA, LWLDA and SLPP (also including P-SLPP) is from the weight matrices $\mathbf{W}$ and $\mathbf{B}$. It shows that the weight matrices are the key to these algorithms and further explains the rationality of improving the weight matrices in the P-SLPP algorithm.

4. Experiments and analysis of results

4.1. Experimental data and evaluation metric

Experiments are carried out on the core conditions of the NIST SRE 2012 database [7]. The total trials contain utterances from five common conditions (CC). There are approximately 1.16 million trials, with 21,248 target trials and 1,143,112 non-target trials. All the models are gender-dependent. The data from previous years' evaluations, Switchboard and the Mix5 dataset are used as a training set.

The performance is evaluated in terms of equal error rate (EER) and the minimal detection cost function (minDCF) defined in the NIST 2012 SRE evaluation protocol.

4.2. System configurations

Each speech signal is parameterized by 39-dimensional perceptual linear predictive (PLP) features containing delta and delta-delta coefficients. The i-vector extractor is based on the GMM/i-vector framework [1]. Since the DNN/i-vector system [12] has good complementarity with the GMM/i-vector system, we also report our results using the score fusion of the two systems with equal weights.

4.2.1. GMM/i-vector system

For the GMM/i-vector system, a gender-dependent 1024-component GMM-UBM with diagonal covariance matrices is trained using a subset of the training set. After UBM is trained, two different total variability matrices are trained to meet the test conditions of telephone and microphone channels. The first matrix with dimension 400 is trained using the telephone data, and the second matrix with dimension 200 is trained using the interview data. Then, these two matrices are stacked to form a matrix with dimension 600, which is used to extract the i-vectors for all utterances.

4.2.2. DNN/i-vector system

For the DNN/i-vector system, an eight-layer DNN with 792 input nodes, 2048 nodes in each hidden layer and 6004 output nodes is trained on approximately 300 hours of clean English telephone speech from Switchboard datasets. The input layer of the DNN is composed of 11 (5+1+5) frames, where each frame corresponds to 24-dimensional log Mel-filterbank features plus their first and second order derivatives. Once DNN is trained, the following procedures are the same as those of the GMM/i-vector system except where the posteriors are produced by the DNN.

4.2.3. I-vector transformation and PLDA scoring

After extracting the 600-dimensional i-vectors, one of the mentioned channel compensation technologies is applied to project i-vectors to a low-dimensional subspace. Since the prior G-PLDA model follows the Gaussian distributions, data whitening and length normalization are adopted before PLDA. After this pre-processing, the PLDA algorithm is adopted as the backend classifier for speaker verification, where the sizes of the speaker and channel matrices are 250 and 10, respectively.

Table 1: *Results for GMM/i-vector (EER%/minDCF*1000)*

Methods	**CC1**	**CC2**	**CC3**	**CC4**	**CC5**
LDA	4.01/372	1.65/218	3.41/380	3.30/351	2.57/290
NDA	3.91/361	1.68/228	**3.25**/364	3.25/348	2.53/304
LWLDA	3.85/355	1.55/217	3.33/373	3.34/352	2.42/296
SLPP	3.85/**352**	1.53/214	3.30/359	3.32/343	2.49/296
P-SLPP	**3.84**/362	**1.38/211**	3.37/**341**	**2.73/307**	**2.30/278**

Table 2: *Results for fusion system (EER%/minDCF*1000)*

Methods	**CC1**	**CC2**	**CC3**	**CC4**	**CC5**
LDA	3.53/357	1.04/167	3.36/350	3.16/248	1.70/214
NDA	3.57/346	1.10/174	3.17/354	3.11/250	1.70/218
LWLDA	3.27/349	1.00/163	3.21/352	3.27/249	1.71/208
SLPP	**3.16**/335	**0.87**/172	**3.16**/329	3.03/254	1.62/206
P-SLPP	3.34/**329**	**0.87/160**	3.41/**311**	**2.32/209**	**1.47/199**

4.3. Results and analysis

Table 1 presents the performance of the GMM/i-vector system with different channel compensation methods in NIST 2012 SRE. It can be observed that SLPP outperforms LDA, NDA and LWLDA in most conditions. On average, the proposed P-SLPP gives the best performance among all these methods in terms of EER or minDCF in all common conditions. The performance of the fusion system is also reported in Table 2, which shows a more obvious improvement provided by SLPP and P-SLPP. Specially, in terms of EER, P-SLPP provides a 23.4-29.1% relative improvement over SLPP, NDA, LDA and LWLDA in the case of CC4. We believe this is due to the combination of SLP-P projection and PLDA scoring, which removes the unwanted variations resulting from changes in noise and channel.

5. Conclusions

In this study, we employ SLPP for channel compensation in an i-vector based speaker verification system. Compared with the conventional LDA and its variants (NDA and LWLDA), SLPP can make better use of local information based on within- and between-speaker connected graphs. To meet the PLDA backend, we propose P-SLPP to improve SLPP based on the relative PLDA scores. Moreover, we uncover that the only difference among these methods (except NDA) is from the weight matrices. The experimental result shows that SLPP and the proposed method outperform LDA, NDA and LWLDA, and among these methods, P-SLPP achieves the best performance.

P-SLPP is the effective combination of channel compensation and backend classification. Accordingly, in our future study, we will continue to focus on the combination methods.

6. Acknowledgements

This work was partially funded by The National Key Research and Development Program of China (Grant No.2016YFB1001303).

7. References

[1] N. Dehak, P. 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, 2011.

[2] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic Press, 1990.

[3] A. Misra, S. Ranjan, and J. H. Hansen, "Locally weighted linear discriminant analysis for robust speaker verification," *Proc. Interspeech 2017*, pp. 2864–2868, 2017.

[4] S. Sadjadi, J. Pelecanos, and W. Zhu, "Nearest neighbor discriminant analysis for robust speaker recognition," in *Proc. Interspeech 2014*, 2014.

[5] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Odyssey*, 2010, p. 14.

[6] S. Zhong-Hua, P. Yong-Hui, and W. Shi-Tong, "A supervised locality preserving projection algorithm for dimensionality reduction," *Pattern Recognition and Artificial Intelligence*, vol. 21, no. 2, pp. 233–239, 2008.

[7] NIST, "The nist year 2012 speaker recognition evaluation plan," http://nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf.

[8] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis," *Journal of machine learning research*, vol. 8, pp. 1027–1061, May 2007.

[9] X. He and P. Niyogi, "Locality preserving projections," *Advances in neural information processing systems*, 2003.

[10] I. Salmun, I. Shapiro, I. Opher, and I. Lapidot, "Plda-based mean shift speakers' short segments clustering," *Computer Speech & Language*, vol. 45, pp. 411–436, September 2017.

[11] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Interspeech*, 2011, pp. 249–252.

[12] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1695–1699.