DNN i-vector based Fishervoice and PLDA SVM scoring for NIST SRE 2016

Jinghua Zhong, Helen Meng

Department of Systems Engineering & Engineering Management,
The Chinese University of Hong Kong, Hong Kong SAR of China
{jhzhong, hmmeng}@se.cuhk.edu.hk

Abstract

Our ongoing work that applies Fishervoice to map joint factor analysis (JFA)-mean supervectors[1] into a compressed discriminant subspace has shown that performing cosine distance scoring on the Fishervoice projected vectors outperforms classical JFA. In this paper, we refine Fishervoice for low-dimensional i-vectors by only using the nonparametric between-class scatter matrix to substitute the parametric one in linear discriminative analysis (LDA). The task of 2016 speaker recognition evaluation (SRE16) only has unlabeled in-domain training data and labeled out-of-domain training data for model training. Support vector machine (SVM) scoring can capture the discriminative information embedded in the unlabeled in-domain training data. We perform probabilistic linear discriminant analysis (PLDA) before SVM scoring for inter-session compensation with speaker label information from out-of-domain training data. This approach constitutes CUHK's submission for SRE16. In this paper, we present a detailed analysis of the approaches and the performance gains with refined Fishervoice and PLDA SVM scoring.

Index Terms: DNN i-vector, Refined Fishervoice, PLDA, SVM, SRE16

1. Introduction

The speaker recognition evaluation (SRE) regularly conducted by the National Institute of Standards and Technology (NIST), strongly supports the research on text-independent speaker verification. There were several new challenges in the latest edition SRE16 [1], including more duration variability in test segments, evaluation data collected outside North America, same and different phone number trials, unlabeled in-domain training data and labeled out-of-domain training data. The main focus of SRE16 was testing the robustness to new languages and channels. Since the most important factors that impact the evaluation performance are channel and language mismatch, effective use of unlabeled in-domain training data is key.

In recent years, the Gaussian Mixture Model (GMM) [2] based i-vector [3] has become a popular approach for text-independent speaker verification. It compresses both channel and speaker information into a low-dimensional space called total variability space, and accordingly projects each GMM supervector to a total factor feature vector called the i-vector. Then LDA [4] and Probabilistic LDA (PLDA) were applied in [5] on the i-vectors for inter-session compensation. In [6], a deep neural network (DNN) trained for automatic speech recognition (ASR) replaced the GMM in traditional i-vector computation. It used DNN senone posterior for frame alignment in the i-vector extraction process. The phonetic information provided through senone posteriors succeeded at improving the accuracy of frame alignment and therefore achieved better speaker verification performance.

Based on the JFA-mean supervector [7], we proposed a speaker recognition framework named Fishervoice in [8, 9]. Using nonparametric Fisher's discriminant analysis, the framework mapped JFA-mean supervectors into multiple discriminant subspaces [10, 11]. Such an algorithm can reduce dimensionality through reducing unfavorable intra-speaker variability. It can also exploit the discriminative information such as classification boundaries in the multiple discriminative subspaces. Sadjadi et al. [12, 13] investigated similar application of nonparametric discriminant analysis for robust speaker recognition. In this work, we propose to refine our Fishervoice approach for i-vectors. First, we note that the first step in Fishervoice is principal component analysis (PCA) [14], which may be redundant for low-dimensional i-vectors, especially when the number of training samples is higher than the dimension of i-vectors. Second, in order to alleviate the limitation of Gaussian distribution assumption in LDA , we only enhance the between-class scatter matrix in LDA to extract discriminative speaker class boundaries — this is the most essential part in Fishervoice.

Besides, PLDA has shown to be a good inter-session compensation method for the i-vector framework. However, it needs speaker labels during model training. SVM, which is first applied on JFA-mean supervectors in speaker recognition, do not need detailed speaker labels for model training. Using SVM scoring on i-vectors is not very popular primarily because of its inferior performance compared with PLDA log-likelihood ratio (LR) scoring and even cosine distance scoring when giving labeled in-domain training data. It was studied in [15, 16] that utterance partitioning with acoustic vector resampling (UP-AVR) [17] could overcome the data imbalance between utterances from target speaker and utterances from background speakers in SVM. Besides, Gang et al. [18] proposed a fast universal background support imposter data selection method for SVM based speaker verification. In [19], Cumani et al. proposed pairwise SVM as a pairwise second degree polynominal kernel classifier in the i-vector pairs space.

Since effective use of the unlabeled in-domain training data is most essential for the task of SRE16, we propose to perform classical SVM scoring to derive the discriminative information embedded in the unlabeled in-domain training data by constructing SVM models for every target speaker. The SVM seeks to find a classifying hyperplane that is optimal for separating the i-vectors of a target speaker from the i-vectors of all the background speakers. We also try to perform refined Fishervoice and PLDA beforehand to reduce intra-speaker variability in the i-vectors with speaker label information using the out-of-domain training data.

[1]The JFA-mean supervector of an utterance is a GMM supervector obtained from the JFA model.

2. Refined Fishervoice and PLDA SVM scoring

2.1. Refined Fishervoice

i-vectors model both speaker- and channel- dependent information, and Linear Discriminant Analysis (LDA) is applied for channel compensation. The approach seeks to find a linear transformation that maximizes the ratio of the determinant of the between-class scatter matrix $\boldsymbol{S_b}$ to that of the within-class scatter matrix $\boldsymbol{S_\omega}$. Given samples from the training dataset, let C be the total number of speakers, H_i be the number of samples for the speaker i, $\boldsymbol{x_{i,h}}$ be the h-th sample vector from speaker i, $\boldsymbol{\mu_i}$ be the sample mean of the speaker i and $\boldsymbol{\mu}$ be the sample mean of all the training data. The optimal projection $\boldsymbol{W_{lda}}$ for LDA is calculated as follows:

$$\begin{aligned} \boldsymbol{W_{lda}} &= \underset{\boldsymbol{W}:\|\boldsymbol{w}_i\|=1}{\arg\max} \frac{\left\|\boldsymbol{W^T S_b W}\right\|}{\left\|\boldsymbol{W^T S_\omega W}\right\|} \\ \boldsymbol{S_\omega} &= \sum_{i=1}^{C}\sum_{h=1}^{H_i}(\boldsymbol{x_{i,h}}-\boldsymbol{\mu_i})(\boldsymbol{x_{i,h}}-\boldsymbol{\mu_i})^T \\ \boldsymbol{S_b} &= \sum_{i=1}^{C} H_i(\boldsymbol{\mu_i}-\boldsymbol{\xi})(\boldsymbol{\mu_i}-\boldsymbol{\xi})^T \end{aligned} \tag{1}$$

Since traditional LDA assumes that all classes obey Gaussian distributions with the same covariance matrix, it suffers a fundamental limitation while using the parametric form of scatter matrix. However, i-vectors may not exactly obey Gaussian distribution as shown in [20]. Besides, with only the centers of classes taken into account for computing matrix $\boldsymbol{S_b}$, the approach fails to capture the boundary structure of classes effectively, which is essential in classification. In our previous work, we proposed Fishervoice [8, 9] to enhance performance by extracting discriminant information from the matrices $\boldsymbol{S_\omega}$ and $\boldsymbol{S_b}$ effectively. Based on JFA-mean supervector, the first step of Fishervoice performed PCA to guarantee that $\boldsymbol{S_\omega}$ is non-singular so as to deal with the small sample size problem. Then we enhanced $\boldsymbol{S_\omega}$ by means of whitening transformation [21]. Last but not least, we enhanced $\boldsymbol{S_b}$ by applying nonparametric subspace analysis to capture the boundary structural information. Let $\boldsymbol{v_{i,h}}$ denotes the new sample vector after PCA and whitening transformation, we consider the contribution of $\boldsymbol{v_{i,h}}$ towards the nonparametric between-class scatter matrix $\boldsymbol{S_b'}$ by focusing on its proximity to the boundary which separates speaker i and any other speaker j. Formally, $\boldsymbol{S_b'}$ was computed according to the following equations:

$$\begin{aligned} \boldsymbol{S_b'} &= \sum_{i=1}^{C}\sum_{j=1,j\neq i}^{C}\sum_{h=1}^{H_i} g(i,j,h)(\boldsymbol{v_{i,h}}-\boldsymbol{m_j(v_{i,h})})(\boldsymbol{v_{i,h}}-\boldsymbol{m_j(v_{i,h})})^T \\ \boldsymbol{m_j(v_{i,h})} &= \frac{1}{R}\sum_{r=1}^{R}\boldsymbol{\varphi_{j,r}(v_{i,h})} \end{aligned} \tag{2}$$

where $\boldsymbol{\varphi_{j,r}(v_{i,h})}$ was the r-th vector from speaker j which was among the neighbors of $\boldsymbol{v_{i,h}}$, R was the number of considered nearest neighbors, $\boldsymbol{m_j(v_{i,h})}$ was the mean vector of these R nearest neighbors and $g(i,j,h)$ was a weighing function defined as:

$$g(i,j,h) = \frac{min\{d^\alpha(\boldsymbol{v_{i,h}},\boldsymbol{\varphi_{i,R}(v_{i,h})}), d^\alpha(\boldsymbol{v_{i,h}},\boldsymbol{\varphi_{j,R}(v_{i,h})})\}}{d^\alpha(\boldsymbol{v_{i,h}},\boldsymbol{\varphi_{i,R}(v_{i,h})}) + d^\alpha(\boldsymbol{v_{i,h}},\boldsymbol{\varphi_{j,R}(v_{i,h})})} \tag{3}$$

where the exponential parameter α controls the variation of the weighting function with respect to the distance $d(\boldsymbol{\nu_1},\boldsymbol{\nu_2})$, which was the Euclidean distance between two vectors $\boldsymbol{\nu_1}$ and $\boldsymbol{\nu_2}$. The parameter R was often set as the median value of the total utterances for each speaker in the training data [22]. This step could extract discriminative speaker class boundaries information which was most essential when the samples were not normally distributed.

In this work, we propose to refine the Fishervoice for i-vectors. Since the dimension of i-vector is low compared with the number of training samples, the first step of Fishervoice is redundant. Besides, in order to only alleviate the above-mentioned limitations of LDA, we directly substitute $\boldsymbol{S_b}$ in Eq. 1 with the nonparametric between-class scatter matrix $\boldsymbol{S_b'}$ and then maximize the ratio of the determinant of $\boldsymbol{S_b'}$ to $\boldsymbol{S_\omega}$ for better class separability.

2.2. PLDA SVM scoring

Ioffe [23] and Prince et al. [5] proposed a probabilistic approach called PLDA which applied generative factor analysis modeling to solve the subspace recognition problem. Kenny et al. [20] introduced heavy-tailed PLDA which used Student's t distributions instead of the Gaussian distribution to model the i-vectors. Significant performance improvement was demonstrated, but the system was complicated and computationally demanding. Later, a simple length normalization scheme [24] was proposed to deal with the non-Gaussian behavior of i-vectors, which allowed the use of probabilistic models with Gaussian assumptions. This non-linear transformation simplified the second step of Radial Gaussianization proposed in [25] by scaling the length of each whitened i-vector to unit length. In this way, PLDA with Gaussian assumptions could achieve a performance comparable to that of heavy-tailed PLDA. In this paper, we focus on PLDA with Gaussian assumptions, named Gaussian PLDA.

Suppose each speaker i has H_i utterances. The Gaussian PLDA model assumes that each length-normalized speaker vector $\boldsymbol{\eta_{ih}}$ can be decomposed as

$$\boldsymbol{\eta_{ih}} = \boldsymbol{m} + \boldsymbol{\Phi\beta_i} + \boldsymbol{\Gamma\alpha_{ih}} + \boldsymbol{\epsilon_{ih}} \tag{4}$$

where $\boldsymbol{m}$ is a global offset, the columns of $\boldsymbol{\Phi}$ provide a basis for the speaker-specific subspace (i.e. eigenvoices), $\boldsymbol{\Gamma}$ provides a basis for the channel subspace (i.e. eigenchannels), $\boldsymbol{\beta_i}$ and $\boldsymbol{\alpha_{ih}}$ are the corresponding latent vectors and $\boldsymbol{\epsilon_{ih}}$ is a residual term. Besides, $\boldsymbol{\beta_i}$ and $\boldsymbol{\alpha_{ih}}$ are both assumed to have standard normal distributions, and $\boldsymbol{\epsilon_{ih}}$ follows a Gaussian distribution with zero mean and diagonal covariance matrix $\boldsymbol{\Sigma}$. If $\boldsymbol{\Sigma}$ is assumed to be a full covariance matrix, then the eigenchannels can be absorbed into $\boldsymbol{\Sigma}$ and the modified model becomes:

$$\boldsymbol{\eta_{ih}} = \boldsymbol{m} + \boldsymbol{\Phi\beta_i} + \boldsymbol{\epsilon_{ih}} \tag{5}$$

During PLDA model training, the ML point estimates of the model parameters $\{\boldsymbol{m},\boldsymbol{\Phi},\boldsymbol{\Sigma}\}$ are obtained from a set of model training data using the expectation-maximization (EM) algorithm as in [5]. Then, log-likelihood ratio scoring (LRS) is the most successful method in PLDA verification score.

Besides, with the model parameters $\{\boldsymbol{m},\boldsymbol{\Phi},\boldsymbol{\Sigma}\}$, the Maximum a Posteriori (MAP) values of speaker factor $\boldsymbol{\beta}$ could be estimated for all the training data and evaluation data as follows:

$$\boldsymbol{\beta} = (\boldsymbol{\Phi}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Phi} + \boldsymbol{I})^{-1}\boldsymbol{\Phi}^{-1}\boldsymbol{\Sigma}^{-1}\boldsymbol{\eta} \tag{6}$$

So instead of scoring with log-likelihood ratios, PLDA is treated similarly as i-vector extraction process to extract speaker vector $\boldsymbol{\beta}$. Then, SVM scoring is performed on the speaker vector as a discriminative classifier. In the training phase,

SVM projects the low-dimensional input vectors to a high-dimensional space to find a classifying hyperplane that maximizes the margin between every target speaker's vectors and all background speakers' vectors. Once the SVM model training is done, the structure of the classifying hyperplane is captured with a small subset of support vectors of both positive and negative samples from the training data. The SVM model for every target speaker remains fixed during the recognition phase. Given the SVM of target speaker s, the verification score with test utterance t is given by

$$S_{SVM}(\boldsymbol{\beta^{(t)}}, \boldsymbol{\beta^{(s)}}) = \sum_i \alpha_i^{(s)} y_i K(\boldsymbol{\beta^{(t)}}, \boldsymbol{\beta_i^{(s)}}) + d^{(s)} \quad (7)$$

where $\alpha_i^{(s)}$ is the Lagrange multiplier of the i-th sample and $\boldsymbol{\beta_i^{(s)}}$ is the i-th sample with class label $y_i \in \{-1, 1\}$ during the SVM model training for speaker s. A linear kernel function $K(\cdot, \cdot)$ was used.

3. Experimental Setup

The Call My Net Speech Collection collected by the LDC was used to compile the SRE16 evaluation set, development set and part of the model training set. It consists of telephone conversations collected outside North America, spoken in Tagalog and Cantonese (named as major languages) and Cebuano and Mandarin (named as minor languages). The development set contains data from the two minor languages while the evaluation set contains data from the two major languages.

3.1. Feature Extraction

For the acoustic features in speaker modeling, the first 19 Mel frequency cepstral coefficients and log energy were calculated, together with their first and second derivatives. Hence, a 60-dimensional MFCC feature vector was obtained for each frame. The frame length was 25ms and the frame shift was 10ms. Then energy-based voice-activity detection (VAD) and sliding-window cepstral mean and variance normalization (CMVN) were applied to remove non-speech frames and for feature normalization. Besides, the feature vectors for the DNN were 40-dimensional MFCC features without cepstral truncation. The features were also pre-processed with sliding-window CMVN.

3.2. The baseline system

The DNN for i-vector extraction was trained on the Fisher dataset. We trained a 6-hidden layer p-norm neural networks [26] with power $p = 2$ using the Kaldi toolkit [27]. The p-norm input/output dimensions in DNN were set to 3,500/350 for each hidden layer. The DNN ASR system was based on the multi-splice time delay DNN described in [28, 29]. In the multisplice system, a narrow temporal context of only 2 frames before and after was provided to the first layer, so the input of the DNN consisted 200 nodes. The softmax output layer computed posteriors for 5,545 senones.

In addition to 2,472 in-domain unlabeled training data, we also selected large amount of labeled data as out-of-domain training set, from Switchboard II Phase 2, NIST SRE2004-2012 for speaker model training, including 52,391 utterances from 2,670 speakers, each with not less than 8 utterances. The i-vector extractor, LDA and PLDA with whitening and length normalization were all trained on this out-of-domain training set. The dimension of i-vector was set to 600. The rank of LDA projection matrix was set to 500. The number of eigenvoices in PLDA LR scoring was set to 500.

3.3. Refined Fishervoice and PLDA SVM scoring

Here, refined Fishervoice and PLDA with whitening and length normalization were trained on the out-of-domain training set. The parameter R in Eq. 2 that controls the number of nearest neighbors for constructing $\boldsymbol{S_b'}$ was set to 4, according to the median number of utterances for each speaker. The ranks of refined Fishervoice projection matrix were set to 500. Both unlabeled in-domain training set and labeled out-of-domain training set for training target speaker models were used to construct kernels of support vector machine by using LIBSVM [30]. The SVM model training was speaker-specific. For every target speaker, we took all his/her i-vectors as positive examples and all the model training data as negative examples to train SVM models for every target speaker. During verification, given a test i-vector, we computed the SVM testing score with every SVM model of target speaker. The rank of Fishervoice and refined Fishervoice projection matrices were set to 500. The number of eigenvoices in PLDA SVM scoring was set to 400.

4. Results

The performance of NIST SRE16 is evaluated using the Equal Error Rate (EER) and in terms of the primary cost function defined in [1]. The primary metric, $C_{primary}$, is the average cost at two specific points on the DET curve. The minimal and actual primary cost, min $C_{primary}$ and act $C_{primary}$, is computed for performance measure.

4.1. PLDA SVM scoring

Table 1 shows the performance of various methods for NIST 2016 SRE on both development set and evaluation set. We consider PLDA with LR scoring as the baseline (rows 1 and 3 in Table 1). We show the performance comparison of PLDA LR scoring and PLDA SVM scoring under different conditions, including performing Fishervoice beforehand. There are four pairs of results (rows 1 and 2, 3 and 4, 5 and 6, 7 and 8) comparing the two scoring methods. Performances showed that no matter whether the i-vectors were preprocessed by LDA, Fishervoice, refined Fishervoice or not, SVM scoring can significantly improve the performance of PLDA with log-likelihood ratio scoring. SVM scoring outperformed PLDA LR scoring in unlabeled in-domain training scenarios, because PLDA LR scoring could not directly use unlabeled in-domain training data. The worse act $C_{primary}$ of our system is because of calibration issue.

4.2. Refined Fishervoice

We also evaluated the effectiveness of the refined Fishervoice (defined as rFishervoice in Table 1) versus our pervious Fishervoice method based on JFA-mean supervector. Comparing rows 5 and 7 using PLDA LR scoring or rows 6 and 8 using PLDA SVM scoring, the results suggest that the refining process in Fishervoice significantly improves the performance. When comparing refined Fishervoice with the traditional LDA, refined Fishervoice performed slightly better than LDA for the development set when using PLDA LR scoring (rows 3 and 7). Compared to obvious improvement of Fishervoice over LDA on JFA-mean supervector [31], refined Fishervoice on i-vector gained less improvement over LDA. Besides, traditional LDA

Table 1: *Performance of various methods for NIST 2016 SRE on development set and evaluation set.*

	Method	EER (%)		min $C_{primary}$		act $C_{primary}$	
		Development	Evaluation	Development	Evaluation	Development	Evaluation
1	PLDA + LRS	19.87	18.36	0.8807	0.9837	0.9658	1.0269
2	PLDA + SVM	17.78	13.17	0.7735	0.7570	1.0000	1.0000
3	LDA + PLDA + LRS	19.88	18.25	0.8738	0.9821	0.9589	1.0374
4	LDA + PLDA + SVM	**17.67**	**13.01**	**0.7554**	0.7536	1.0000	1.0000
5	Fishervoice + PLDA + LRS	22.54	20.37	0.8843	0.9909	**0.9530**	1.1986
6	Fishervoice + PLDA + SVM	20.02	14.26	0.8470	0.7938	1.0000	1.0000
7	rFishervoice + PLDA + LRS	19.53	18.26	0.8664	0.9839	0.9580	1.0518
8	rFishervoice + PLDA + SVM	18.01	13.05	0.7766	**0.7525**	1.0000	**1.0000**
9	Fusion	16.04	12.35	0.7515	0.7533	0.7672	0.7604

only gained a little improvement on i-vector with PLDA LR scoring (rows 1 and 3). PLDA seems to be the most essential part for inter-session compensation in the i-vector framework, although it is trained with the out-of-domain training data. Both LDA and refined Fishervoice could gain little improvement when combined with PLDA on i-vector.

We also take the NIST 2010 SRE for performance comparison (see Table 2). The experiment setup is the same as that performed on NIST 2016 SRE. From the table, we can see that we also obtained consistent observations on the NIST SRE 2010 extended core task. Similar results between LDA and refined Fishervoice maybe due to the better Gaussian distributions of i-vectors than JFA-mean supervectors.

Table 2: *Performance of various methods for NIST 2010 SRE on coreext-coreext task of cc5.*

Method	EER(%)	minDCF08
PLDA	1.05	0.0048
LDA + PLDA	1.03	0.0045
Fishervoice + PLDA	1.66	0.0081
rFishervoice + PLDA	**1.00**	**0.0045**

4.3. Score Fusion and Analysis

We fused all the methods in Table 1 (row 1-8) for the final verification scores of Row 9. The Bosaris toolkit [32] was used for score calibration and fusion. Score fusion achieved a significant improvement in terms of act $C_{primary}$. We compared the three main approaches on the evaluation set, LDA with PLDA LR scoring, refined Fishervoice with PLDA LR scoring and refined Fishervoice with PLDA SVM scoring (rows 3, 7 and 8 in Table 1), with the fusion results on DET curve in Figure 1. From the figure, we can see that SVM scoring can significantly improve the performance (the green line and red line) while refined Fishervoice shows similar results with LDA (the red line and blue line).

Finally, we evaluated the fusion scores with respect to gender, language enrollment segments' No. and phone difference for enrollment and test as in Table 3. From the results, we have two observations. First, language, number of enrollment segments and the phone difference highly influenced the performance. Besides, Cantonese trials performed much better than Tagalog trials. Results in the development set also showed that Mandarin trials performed much better than Cebuano trials.

5. Conclusions

This paper presented a detailed analysis of our approaches and performance comparison for SRE16. We proposed refining our

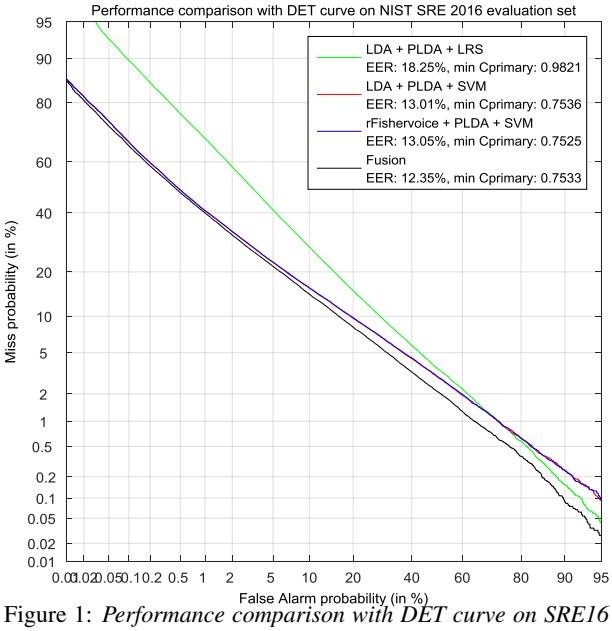

Figure 1: *Performance comparison with DET curve on SRE16 evaluation set.*

Table 3: *Performance comparison of fusion results based on different catalogues for SRE16 evaluation set(EER(%)).*

Catalogue		EER	min $C_{primary}$	act $C_{primary}$
Gender	Male	11.85	0.7329	0.7441
	Female	12.64	0.7684	0.7766
Language	Tagalog	16.76	0.8475	0.8959
	Cantonese	**7.82**	**0.6181**	**0.6248**
Enrollment segments' No.	1	15.05	0.8146	0.8379
	3	**9.47**	**0.6334**	**0.6829**
Phone difference for enrollment and test	Same	**11.01**	**0.7088**	**0.7157**
	Different	15.17	0.8627	0.8727

Fishervoice method for low-dimensional i-vectors and PLDA SVM scoring to effectively use the speaker label information in the out-of-domain training data and discriminative information embedded in the unlabeled in-domain training data. Performance showed that refined Fishervoice gained significant improvement over Fishervoice and showed slightly better performance than LDA. Besides, PLDA SVM scoring could significantly improve the performance. Future work will investigate how to leverage insufficient in-domain data in DNN i-vector by DNN adaptation as in [33]. Also, we will experiment with PLDA adaptation to make use of unlabeled in-domain data.

6. References

[1] "NIST 2016 speaker recognition evaluation plan," https://www.nist.gov/itl/iad/mig/speaker-recognition-evaluation-2016.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1, pp. 19–41, 2000.

[3] 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.

[4] 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, 1997.

[5] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proceedings ICCV*, 2007, pp. 1–8.

[6] Y. Lei, L. Ferrer, M. McLaren *et al.*, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proceedings ICASSP*. IEEE, 2014, pp. 1695–1699.

[7] 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, 2008.

[8] Z. Li, W. Jiang, and H. Meng, "Fishervioce: A discriminant subspace framework for speaker recognition," in *Proceedings ICASSP*, 2010, pp. 4522–4525.

[9] W. Jiang, H. Meng, and Z. Li, "An enhanced Fishervoice subspace framework for text-independent speaker verification," in *Proceedings ISCSLP*, 2010, pp. 300–304.

[10] J. Zhong, W. Jiang, H. Meng, N. Li, and Z. Li, "An integration of random subspace sampling and Fishervoice for speaker verification," in *Proceedings Odyssey*, 2014, pp. 88–93.

[11] J. Zhong, W. Jiang, W. Rao, M.-W. Mak, and H. Meng, "PLDA modeling in the Fishervoice subspace for speaker verification," in *Proceedings Interspeech*, 2014, pp. 1130–1134.

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

[13] S. O. Sadjadi, S. Ganapathy, and J. W. Pelecanos, "The ibm 2016 speaker recognition system," *arXiv preprint arXiv:1602.07291*, 2016.

[14] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[15] W. Rao and M.-W. Mak, "Boosting the performance of i-vector based speaker verification via utterance partitioning," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1012–1022, 2013.

[16] M.-W. Mak and W. Rao, "Likelihood-ratio empirical kernels for i-vector based PLDA-SVM scoring," in *Proceedings ICASSP*. IEEE, 2013, pp. 7702–7706.

[17] ——, "Utterance partitioning with acoustic vector resampling for GMM-SVM speaker verification," *Speech Communication*, vol. 53, no. 1, pp. 119–130, 2011.

[18] G. Liu, J.-W. Suh, and J. H. Hansen, "A fast speaker verification with universal background support data selection," in *Proceedings ICASSP*. IEEE, 2012, pp. 4793–4796.

[19] S. Cumani, N. Brümmer, L. Burget, P. Laface, O. Plchot, and V. 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.

[20] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proceedings Odyssey*, 2010, pp. 14–18.

[21] A. R. Webb, *Statistical pattern recognition*. John Wiley & Sons, 2003.

[22] Z. Li, D. Lin, and X. Tang, "Nonparametric discriminant analysis for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 755–761, 2009.

[23] S. Ioffe, "Probabilistic linear discriminant analysis," in *Proceedings ECCV*, 2006, pp. 531–542.

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

[25] S. Lyu and E. P. Simoncelli, "Nonlinear extraction of independent components of natural images using radial Gaussianization," *Neural Computation*, vol. 21, no. 6, pp. 1485–1519, 2009.

[26] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *Proceedings ICASSP*. IEEE, 2014, pp. 215–219.

[27] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *Proceedings ASRU*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[28] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proceedings Interspeech*, 2015, pp. 3214–3218.

[29] D. Snyder, D. Garcia-Romero, and D. Povey, "Time delay deep neural network-based universal background models for speaker recognition," in *Proceedings ASRU*. IEEE, 2015, pp. 92–97.

[30] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[31] W. Jiang, Z. Li, and H. Meng, "An analysis framework based on random subspace sampling for speaker verification," in *Proceedings Interspeech*, 2011.

[32] N. Brümmer and E. De Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," *arXiv preprint arXiv:1304.2865*, 2013.

[33] J. Zhong, W. Hu, F. Soong, and H. Meng, "DNN i-vector speaker verification with short, text-constrained test utterances," in *Proceedings Interspeech*, 2017, pp. 1507–1511.