Locality Sensitive Discriminant Analysis for Speaker Verification

Danwei Cai† Weicheng Cai* Zhidong Ni† Ming Li*†
* SYSU-CMU Joint Institute of Engineering, School of Electronics and Information Technology,
Sun Yat-Sen University, Guangzhou, China
† SYSU-CMU Shunde International Joint Research Institute, Foshan, China
E-mail: liming46@mail.sysu.edu.cn

***Abstract*—In this paper, we apply Locality Sensitive Discriminant Analysis (LSDA) to speaker verification system for intersession variability compensation. As opposed to LDA which fails to discover the local geometrical structure of the data manifold, LSDA finds a projection which maximizes the margin between i-vectors from different speakers at each local area. Since the number of samples varies in a wide range in each class, we improve LSDA by using adaptive *k* nearest neighbors in each class and modifying the corresponding within- and between-class weight matrix. In that way, each class has equal importance in LSDA's objective function. Experiments were carried out on the NIST 2010 speaker recognition evaluation (SRE) extended condition 5 female task, results show that our proposed adaptive *k* nearest neighbors based LSDA method significantly improves the conventional i-vector/PLDA baseline by 18% relative cost reduction and 28% relative equal error rate reduction.**

I. Introduction

Current speaker recognition systems widely use i-vector modeling due to its excellent performance as well as its small model size [1] [2]. I-vector based speaker verification systems first calculate zero-order and first-order Baum-Welch statistics by projecting the MFCC features onto Universal Background Model (UBM). Then a single factor analysis is used as a front-end to generate a low dimensional total variability space (i.e. the i-vector space) which jointly models language, speaker and channel variabilities [2]. After i-vectors are extracted, Probabilistic Linear Discriminative Analysis (PLDA) is widely adopted as a back-end modeling approach [3][4][5].

Conventionally, in the i-vector framework, the tokens for calculating the zero- and first-order statistics are the MFCC features trained GMM components. Recently, tokens in the i-vector framework for calculating the zero-order statistics have been extended to tied triphone states, tandem or bottleneck features trained GMM components [6][7][8][9]. The features for calculating the first-order statistics have also been extended from MFCC to feature level acoustic and phonetic fused features [8]. The phonetically-aware tokens trained by supervised learning can provide better token alignment, which leads to a significant performance improvement on the text independent speaker verification tasks [6][7][8][9][10].

This research was funded in part by the National Natural Science Foundation of China (61401524), Natural Science Foundation of Guangdong Province (2014A030313123), the Fundamental Research Funds for the Central Universities(15lgjc10) and National Key Research and Development Program (2016YFC0103905)

Within the i-vector space, Linear Discriminative Analysis (LDA) [11] can be performed before PLDA scoring to generate dimensionality reduced and channel compensated features so that we can reduce the dimensions and variabilities in i-vectors. Intrinsically, LDA tries to estimate the global statistics and only seek a linear manifold based on the Euclidean structure. It may fail to discover the structure which lies on linear submanifolds hidden in the total variability space.

Recently, nonparametric discriminant analysis (i.e. Nearest-Neighbor Discriminant Analysis, NDA), has been successfully applied to speaker verification systems for variabilities compensation [12][13][14]. This motivates us to explore other nonparametric discriminant analysis algorithms for i-vector dimension reduction, e.g. Locality Sensitive Discriminant Analysis [15]. LSDA finds k nearest neighbors globally for each sample, constructs within- and between-class graph to model the local geometrical structure. Then it finds a linear transform matrix to map the i-vectors into a subspace in which the margin between i-vectors from different speakers is maximized at each local area. Compared to LSDA, NDA finds k nearest neighbors in each different class so that its computational complexity is much more higher. In order to gain good performance, LSDA with k nearest neighbors requires the data samples in each class to be larger than k or close to k, but we can not guarantee that because the number of i-vectors in each speaker is heterogeneously distributed. Considering this inherent characteristic of the training set, we improve LSDA by using adaptive k nearest neighbors for each speaker. Since the number of i-vectors for each speaker is not the same and sometimes even varies with a wide range, the speakers with fewer i-vectors have little influence in the objective function of LSDA. We further modify LSDA's within-class and between-class weight matrix to handle this issue of unbalanced data.

II. System Overview

The overview of our speaker verification system with LSDA for variabilities compensation is shown in Fig.1.

A. DNN Tandem Feature Extraction

In the system, Deep Neural Network serves as an acoustic modeling network used to extract phonetic level tandem feature. At first, a DNN acoustic model is trained using acoustic features and phonetic label data. Then MFCC feature is given

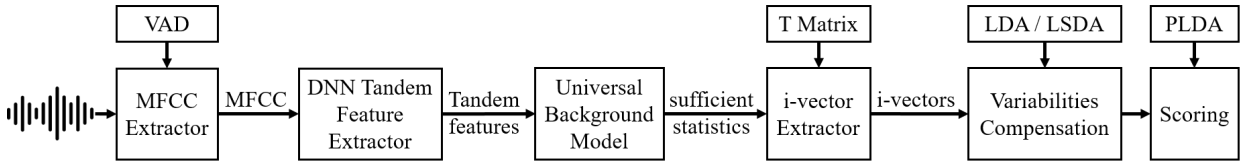


Fig. 1. The overview of speaker verification system with LSDA.

to the DNN model and we can extract tied triphone states phoneme posterior probabilities. We apply PCA on top of it to generate the low dimensional tandem features. Finally tandem feature is concatenated to MFCC feature to generate the hybrid feature [8].

B. i-vector Extraction

The i-vector extractor is to map a sequence of feature vectors (typically MFCC or other features) from a speech utterance to a low dimensional fixed-length vector. It is based on the total variability modeling concept which assumes that speaker-, language- and channel- dependent variabilities reside in a same low-dimensional subspace described by the total variability factor loading matrix $\mathbf{T}$. The supervector $\mathbf{M}(s)$ for a given speech utterance s can be modeled as:

$$\mathbf{M}(s) = \mathbf{m} + \mathbf{T}\mathbf{x}(s) + \epsilon \tag{1}$$

where $\mathbf{m}$ is the global mean supervector, $\mathbf{x}(s) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a normal distributed factor, and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ is a residual noise term to account for the noise that can not be captured by $\mathbf{T}$.

In order to learn the total variability matrix $\mathbf{T}$, Baum-Welch statistics are calculated on GMM Universal Background Model (UBM) as follows to generate supervectors:

$$N_c(s) = \sum_t \gamma_{tc}(s) \tag{2}$$

$$\mathbf{F}_c(s) = \sum_t \gamma_{tc}(s)\mathbf{O}_t(s) \tag{3}$$

where $N_c(s)$ and $\mathbf{F}_c(s)$ denote the zero-order and first-order statistics for speech utterance s. $\gamma_{tc}(s)$ means the posterior probability of the GMM component c given the DNN hybrid feature $\mathbf{O}_t(s)$ at the t^{th} frame .

III. Locality Sensitive Discriminant Analysis

A. Related Work: Linear Discriminant Analysis (LDA)

As stated before, i-vectors model speaker-, language- and channel- dependent information within the same total variability subspace. In order to select the most speaker relevant feature subset for PLDA modeling, LDA can be used to reduce the information irrelevant to the speaker.

Suppose all i-vectors belongs to c speakers and each i-vector $\mathbf{x}_i$ is associated with a specific speaker $l(\mathbf{x}_i) \in \{1, 2, \ldots, c\}$. Linear Discriminant Analysis (LDA) finds an optimum linear projection $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_d) \in \mathbb{R}^{n\times d} : \mathbb{R}^n \mapsto \mathbb{R}^d$. The objective function of LDA is defined as follows:

$$\mathbf{a}_{opt} = \arg\max_{\mathbf{a}} \frac{\mathbf{a}^T S_b \mathbf{a}}{\mathbf{a}^T S_w \mathbf{a}} \tag{4}$$

$$S_b = \sum_{i=1}^{c} m_i(\boldsymbol{\mu}^i - \boldsymbol{\mu})(\boldsymbol{\mu}^i - \boldsymbol{\mu})^T \tag{5}$$

$$S_w = \sum_{i=1}^{c}\sum_{j=1}^{m_i}(\mathbf{x}_j^i - \boldsymbol{\mu}^i)(\mathbf{x}_j^i - \boldsymbol{\mu}^i)^T \tag{6}$$

where $\boldsymbol{\mu}$ is the total mean i-vector, m_i the number of i-vectors in the i-th speaker, and $\mathbf{x}_j^i$ the j-th i-vector in the i-th speaker. We call S_b the between-class scatter matrix and S_w the within-class scatter matrix. The LDA transform is then formed by calculating the eigenvectors of $S_w^{-1}S_b$.

Clearly, LDA assumes the underlying distribution of classes to be Gaussian with a common covariance matrix for all classes. However, the actual distribution of i-vectors may not necessarily be Gaussian [16]. For the NIST SRE type of scenarios, speech recordings come from various sources and are collected in the presence of noise and channel distortions; therefore unimodality of the distributions can not be guaranteed. Moreover, LDA aims to preserve the global class relationship between data points, but it fails to discover the intrinsic local geometrical structure of the data manifold [11]. In speaker verification task, there may not be sufficient training samples and LDA may not be able to accurately estimate the global structure so the local structure becomes more important.

B. Locality Sensitive Discriminant Analysis (LSDA)

Considering the particular goal of maximizing local margin between different classes, a nonparametric discriminant analysis is proposed in [15].

Given m data points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\} \subset \mathbb{R}^n$ sampled from the underlying submanifold $\mathcal{M}$, one can build a nearest neighbor graph G with weight matrix W to model the local geometrical structure of $\mathcal{M}$. For each data sample $\mathbf{x}_i$, we split its k nearest neighbors $N(\mathbf{x}_i) = \{\mathbf{x}_i^1, \ldots, \mathbf{x}_i^k\}$ into two subsets: $N_w(\mathbf{x}_i)$ contains the neighbors sharing the same label with $\mathbf{x}_i$ and $N_b(\mathbf{x}_i)$ contains the neighbors having different labels. Specifically,

$$\begin{aligned} N_w(\mathbf{x}_i) &= \{\mathbf{x}_i^j | l(\mathbf{x}_i^j) = l(\mathbf{x}_i), 1 \le j \le k\} \\ N_b(\mathbf{x}_i) &= \{\mathbf{x}_i^j | l(\mathbf{x}_i^j) \ne l(\mathbf{x}_i), 1 \le j \le k\} \end{aligned} \tag{7}$$

In order to discover both geometrical and discriminant structure of the data manifold, we split the nearest neighbor graph G into within-class graph G_w and between-class graph G_b. The weight matrices of G_b and G_w can be defined as:

$$\begin{aligned} W_{b,ij} &= \begin{cases} 1, & \text{if } \mathbf{x}_i \in N_b(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_b(\mathbf{x}_i) \\ 0, & \text{otherwise} \end{cases} \\ W_{w,ij} &= \begin{cases} 1, & \text{if } \mathbf{x}_i \in N_w(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_w(\mathbf{x}_i) \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{8}$$

LSDA finds a mapping so that connected points of G_w stay as close as possible, while connected points of G_b stay as far away as possible. Let $\mathbf{A}$ be such an optimum linear projection, that is, $\mathbf{y}_i = \mathbf{A}^T\mathbf{x}_i$. The criterion of LSDA is to optimize the following two objects:

$$\min \sum_{ij} (\mathbf{y}_i - \mathbf{y}_j)^T W 3_{w,ij} (\mathbf{y}_i - \mathbf{y}_j) \tag{9}$$

$$\max \sum_{ij} (\mathbf{y}_i - \mathbf{y}_j)^T W_{b,ij} (\mathbf{y}_i - \mathbf{y}_j) \tag{10}$$

By simple reformulation, the objective function (9) and (10) can be reduced to (11) and (12), respectively.

$$\begin{aligned} &\frac{1}{2}\sum_{ij} (\mathbf{y}_i - \mathbf{y}_j)^T W_{w,ij} (\mathbf{y}_i - \mathbf{y}_j) \\ =&\mathbf{a}^T X D_w X^T \mathbf{a} - \mathbf{a}^T X W_w X^T \mathbf{a} \end{aligned} \tag{11}$$

$$\frac{1}{2}\sum_{ij} (\mathbf{y}_i - \mathbf{y}_j)^T W_{b,ij} (\mathbf{y}_i - \mathbf{y}_j) = \mathbf{a}^T X L_b X^T \mathbf{a} \tag{12}$$

where D_w is a diagonal matrix, $D_{w,ii} = \sum_j W_{w,ij}$. $L_b = D_b - W_b$ is the Laplacian matrix of G_b. $X = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$ is a $n \times m$ matrix.

It is clear that D_w provides a natural measure on the data points. If $D_{w,ii}$ is large, then it implies that the class containing $\mathbf{x}_i$ has a high density around $\mathbf{x}_i$ and it is more important. Therefore, we impose a constraint as follows:

$$Y^T D_w Y = 1 \Rightarrow \mathbf{a}^T X D_w X^T \mathbf{a} = 1 \tag{13}$$

Thus, objection function (11) becomes the following:

$$\min_{\mathbf{a}} 1 - \mathbf{a}^T X W_w X^T \mathbf{a} \tag{14}$$

or equivalently,

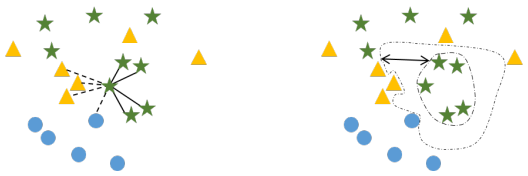
$$\max_{\mathbf{a}} \mathbf{a}^T X W_w X^T \mathbf{a} \tag{15}$$

Finally, the optimization problem is reduced to finding:

$$\arg\max_{\mathbf{a}} \mathbf{a}^T X H X^T \mathbf{a}, \ s.t. \ \mathbf{a}^T X D_w X^T \mathbf{a} = 1 \tag{16}$$

where $H = \alpha L_b + (1-\alpha) W_w$ and α is a parameter to tune the weight between within-class graph and between-class graph, $0 \leq \alpha \leq 1$. The LSDA transform is then formed by calculating the eigenvectors of $(X D_w X^T)^{-1} X(\alpha L_b + (1-\alpha) W_w) X^T$.

Fig. 2 shows the learning procedure. At first, the center star point finds its 8 nearest neighbors. The within-class graph connects nearby 4 points with the same label, while the between-class graph connects nearby 4 points with different labels. After we apply LSDA, the margin between different speakers is maximized.



(a) G_w and G_b (b) Maximize margin

Fig. 2. LSDA learning procedure.

C. LSDA with Adaptive K Nearest Neighbors

LSDA finds the k nearest neighbors to construct the within-class graph and between-class graph. Suppose a special case where there is only one sample in a class. Then the within-class graph of this data sample has nothing, while the between-class graph has the whole k nearest data samples with different labels in it. Looking into the speaker verification tasks, the training dataset has a lot of speakers with a few speech utterances while others may have much more. But LSDA requires the number of data samples from each class in training set to be at the same level so that LSDA can efficiently learn the geometrical structure of the data manifold.

This unbalanced data issue motivates us to use adaptive k nearest neighbors in each class. Instead of finding the whole nearest neighbor graph G then splitting it into between- and within-class graph, we construct these two graphs independently: k nearest neighbors with the same label of a specific data sample are found to construct a within-class graph and βk nearest neighbors with different labels are found to construct a between-class graph, where β is a constant coefficient. Now $N_w(\mathbf{x}_i)$ contains k nearest neighbors sharing the same label with $\mathbf{x}_i$ and $N_b(\mathbf{x}_i)$ contains βk nearest neighbors having different labels.

If the number of samples n_c in class c is less than k, then we choose k to be n_c and construct within-class graph and between-class graph with parameters n_c and βn_c accordingly.

D. Weight Matrix W_b and W_w

As noted before, the numbers of speech utterances are quite different for different speakers in the training data. Small class with fewer speech utterances contributes less in the final objective function (16), while big class contributes more. We want each class to have equal importance in (16), so we modify the within-class weight matrix and between-class weight matrix in (8) as follows:

$$\begin{aligned} W_{b,ij} &= \begin{cases} k/n, & \mathbf{x}_i \in N_b(\mathbf{x}_j) \vee \mathbf{x}_j \in N_b(\mathbf{x}_i) \wedge n < k \\ 1, & \mathbf{x}_i \in N_b(\mathbf{x}_j) \vee \mathbf{x}_j \in N_b(\mathbf{x}_i) \wedge n \geq k \\ 0, & \text{otherwise} \end{cases} \\ W_{w,ij} &= \begin{cases} k/n, & \mathbf{x}_i \in N_w(\mathbf{x}_j) \vee \mathbf{x}_j \in N_w(\mathbf{x}_i) \wedge n < k \\ 1, & \mathbf{x}_i \in N_w(\mathbf{x}_j) \vee \mathbf{x}_j \in N_w(\mathbf{x}_i) \wedge n \geq k \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{17}$$

where n is the number of data samples in $\mathbf{x}_i$'s class.

IV. Experiments

A. Data

We conduct our speaker verification experiments using conversational telephone speech material extracted from datasets released through the Linguistic Data Consortium (LDC) for the NIST 2004-2010 SRE, as well as Switchboard Phase2 Part1, Part2 and Part3 corpora. These datasets contain speech spoken in English, Mandarin, Russain and other languages from a large number of male and female speakers. We conducted our experiment on NIST SRE extended condition 5 (tel-to-tel) female part task. In the evaluation set, there are 2361 enroll models and 379 test segments [17]. In total there are 233077 trials containing 3704 target trials.

B. Speaker Verification System Configuration

For cepstral feature extraction, a 25 ms Hamming window with 10 ms shifts was adopted. Each utterance was converted into a sequence of 36-dimensional feature vectors, each consisting of 18 MFCC coefficients and their first derivatives. We employ the Czech phoneme recognizer [18] to perform the voice activity detection (VAD) by simply dropping all frames that are decoded as silence or speaker noises. Feature warping is applied to mitigate variabilities.

For phonetic feature extraction, we employed an DNN acoustic model and output the frame level phoneme posterior probability. After log, PCA and MVN, the resulted 52 dimensional features are fused with MFCC at the feature level to get the 88 dimensional hybrid tandem feature.

About 1,800 hours of the English portion of Fisher [19] is used to train the DNN front end system. The system is based on multisplice time delay deep neural network (TDNN) [20] in the recipe of the Kaldi toolkit. We use the TDNN structure described in [20] to train the DNN acoustic model.

Switchboard II part1 to part3, NIST SRE 2004, 2005, 2006 and 2008 corpora on the telephone channel are used to learn a 500-dimensional total variability subspace. We first train a gender-dependent 1024-component GMM-UBM model with the extracted 88-dimensional tandem features using NIST SRE 2004 and 2005 corpora. Then zreo-order and first-order Baum-Welch stastics are computed for each recording to extract i-vectors.

After extracting 500-dimensional i-vectors, we use either LDA or LSDA for inter-session compensation by reducing the dimensionality to 350. For LSDA, we find 100 nearest neighbors and the constant α in (16) is set to 0.1. For LSDA with adaptive k nearest neighbors, we set k to be 20 in finding within-class nearest neighbors and $3k$ in finding between-class nearest neighbors. For LSDA with adaptive k nearest neighbors and modified weight matrix, we use the within-class matrix W_w and between-class matrix W_b described in (17). The dimensionality reduced i-vectors are then centered, whitened and unit-length normalized. The Gaussian PLDA model with a full covariance residual noise term is trained on i-vectors extracted from all training data which amounted to 2,790 speakers and 30,600 speech files. The eigenvoice subspace in the PLDA model is assumed to be full-rank.

C. Results and Discussion

Table I shows the experiments results obtained with experimental setup presented above. We evaluate the effectiveness

TABLE I
PERFORMANCE COMPARISON ON NIST SRE 2010

System	minDCF10	minDCF08	EER [%]
Baseline (i-vector/PLDA)	0.2222	0.0681	1.62
LDA	0.2198	0.0672	1.59
NDA	0.2130	0.0667	1.59
LSDA	0.1957	0.0609	1.43
LSDA-adaptive k	0.2048	0.0573	1.19
LSDA-adaptive k-weight	**0.1843**	**0.0538**	**1.16**

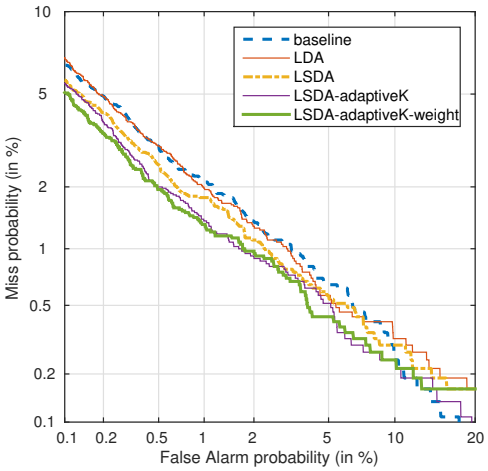


Fig. 3. DET plot comparison on NIST SRE 2010

of the LSDA and its improved adaptive k nearest neighbor version versus LDA for inter-session variability compensation and dimensionality reduction in the i-vector space. It is observed that the LSDA based system with adaptive k nearest neighbors and modified weight matrix outperformed the baseline by 28.4% and 17.1% relative error reduction in terms of EER and norm new minDCF, respectively. We also implement NDA in [14] but experiment result shows the little improvement as LDA. We guess this may due to our implementation as well as the untuned parameters because of the high computation complexity. Furthermore, Figure 3 shows the Detection Error Trade-off (DET) curves of the baseline system and LDA or LSDA based systems. We can find out that the proposed LSDA method achieves significant performance enhancement. This improvement may be because LSDA makes no assumptions regarding the underlying class distribution. What's more, LSDA is good at modeling the local geometrical structure so that boundary information within and across different speakers can be well captured. Our improved version of LSDA gives all the classes the same status in the final objective function (16) so that this geometrical information is further strengthened.

V. CONCLUSIONS

In this paper, we apply the Locality Sensitive Discriminant Analysis (LSDA) to the start-of-the-art i-vector/PLDA speaker verification system. In addition, we propose the adaptive k nearest neighbors idea to LSDA so that it fits the speaker verification task dataset well. The most prominent property of LSDA is the complete preservation of both discriminant and local geometrical structure in the data. Experiments on NIST SRE2010 have been conducted to demonstrate the effectiveness of LSDA in inter-session variability compensation in the i-vector space. In future work, we will try out the adaptive k nearest neighbors idea in NDA to see whether it can achieve the same performance improvement.

REFERENCES

[1] N. Dehak, P. A. Torres-Carrasquillo, D. A. Reynolds, and R. Dehak, "Language recognition via i-vectors and dimensionality reduction," in *Interspeech*, pp. 857–860, 2011.

[2] 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, 2011.

[3] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, IEEE, 2007.

[4] P. Matějka, O. Glembek, F. Castaldo, M. J. Alam, O. Plchot, P. Kenny, L. Burget, and J. Černocky, "Full-covariance ubm and heavy-tailed plda in i-vector speaker verification," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4828–4831, IEEE, 2011.

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

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

[7] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey*, pp. 293–298, 2014.

[8] M. Li and W. Liu, "Speaker verification and spoken language identification using a generalized i-vector framework with phonetic tokenizations and tandem features," in *INTERSPEECH*, pp. 1120–1124, 2014.

[9] L. D'haro, R. Cordoba, C. Salamea, and J. D. Echeverry, "Extended phone log-likelihood ratio features and acoustic-based i-vectors for language recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5342–5346, IEEE, 2014.

[10] M. Li, "Automatic recognition of speaker physical load using posterior probability based features from acoustic and phonetic tokens," in *Interspeech*, pp. 437–441, 2014.

[11] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic press, 2013.

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

[13] S. O. Sadjadi, J. W. Pelecanos, and W. Zhu, "Nearest neighbor discriminant analysis for robust speaker recognition," in *INTERSPEECH*, pp. 1860–1864, 2014.

[14] S. O. Sadjadi, S. Ganapathy, and J. W. Pelecanos, "The ibm 2016 speaker recognition system," *Odyssey*, 2016.

[15] D. Cai, X. He, K. Zhou, J. Han, and H. Bao, "Locality sensitive discriminant analysis," in *IJCAI*, pp. 708–713, 2007.

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

[17] NIST, "The nist year 2010 speaker recognition evaluation plan." www.nist.gov/itl/iad/mig/upload/NIST_SRE10_evalplan-r6.pdf.

[18] P. Schwarz, P. Matejka, and J. Cernocky, "Hierarchical structures of neural networks for phoneme recognition," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, 2006.

[19] C. Cieri, D. Miller, and K. Walker, "The fisher corpus: a resource for the next generations of speech-to-text," in *LREC*, vol. 4, pp. 69–71, 2004.

[20] D. Snyder, D. Garcia-Romero, and D. Povey, "Time delay deep neural network-based universal background models for speaker recognition," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 92–97, IEEE, 2015.