LOCAL FISHER DISCRIMINANT ANALYSIS FOR SPOKEN LANGUAGE IDENTIFICATION

Peng Shen, Xugang Lu, Lemao Liu, Hisashi Kawai

National Institute of Information and Communications Technology, Japan

ABSTRACT

I-vector is a state-of-the-art technique widely used in spoken language identification systems. Since i-vectors include total variability factors, discriminant analysis methods have been introduced to find the most discriminative features while removing the undesired variables for language identification, for example, linear discriminant analysis (LDA) and nonparametric discriminant analysis (NDA). However, these methods either do not consider or use weak local structures of the data. In this study, we introduce a local Fisher discriminant analysis (LFDA) as a post-processing discriminant analysis method to extract the discriminative features from i-vectors. LFDA is a full-rank method which takes the local structure of the data into account for non-Gaussian distribution data, i.e., multimodal. Compared with LDA and NDA, LFDA is a pair-wise local method which enhances the centralization of the distribution of samples in the same class to obtain larger amounts of discriminative features. Experimental results indicate that LFDA is more effective than LDA and NDA for the i-vector-based language identification task.

Index Terms— LFDA, language identification, discriminant analysis, i-vector

1. INTRODUCTION

Spoken language identification (LID) is a task to determine whether the spoken language exists within a speech utterance or not [1, 2]. Recently, an i-vector-based approach widely used in speaker recognition has been introduced to LID [3]. The i-vector paradigm provides an effective way to compress Gaussian mixture model (GMM) supervectors by confining all sorts of variabilities (both language and non-language) to a low-dimensional subspace, referred to as the total variability. I-vector-based approach obtained state-of-the-art performance in many systems for both speaker recognition and language identification tasks [3, 4, 5, 6].

Since i-vector models all sorts of variabilities, such as language, speaker, channel, session, in the same total variability space, a Fisher's linear discriminant analysis (LDA) method [7] was widely used in the i-vector-based approach to determine a number of discriminative vectors by transforming features into a new space with an axis transformation. However, as introduced in several studies [5, 8, 9], there are several disadvantages in the conventional LDA method. Applying LDA on i-vector approach for LID tasks also face some challenges which will limit the performance of the system: (1) LDA method assumes the underlying distribution of classes to be Gaussian with a common covariance matrix for all classes. However, the distribution of i-vectors shows multimodality (samples in the same class belong to several separate clusters) due to short duration utterance and data degradation [10, 11]. (2) LDA can only provide up to $C-1$ (C is the number of classes) discriminative features limited within the rank of between-class scatter matrix. This low dimensional features may not be sufficient for the i-vector-based LID tasks in which the number of target languages is much smaller than the dimensionality of the i-vectors. (3) In LDA analysis, the global average centroid of each class is taken into account in class scatter matrix calculation while ignores the local data structure variation.

To effectively embed multimodal data, a locality-preserving projection (LPP) method [12] was proposed for multimodal data by considering its local structure. However, LPP is an unsupervised method which does not take the label information into account. Nonparametric discriminant analysis (NDA) method [8] was proposed to overcome the above limitations of LDA, by measuring the between-class scatter matrix on a local basis using the k-nearest neighbor (NN) rule. The scatter matrix is generally full-rank, thus loosens the bound on extracted feature dimensionality. Recently, NDA was successfully applied on face recognition [13] and LID task [5], which used the means of the local k-NN rule on between-class scatter matrix. However, NDA method use a weak local structure of the data.

In this study, we introduce a local Fisher discriminant analysis (LFDA) [9] method to overcome the above limitations of LDA for the i-vector-based LID task. LFDA was proposed for dimensionality reduction and it was successfully applied for face recognition tasks [14, 15]. To the best of our knowledge, the LFDA method has not yet been studied for either i-vector-based systems or LID tasks. Different from NDA, LFDA is a pair-wise local method which enhances the centralization of the distribution of samples in the same class to obtain greater amounts of discriminative features. The main contributions of this study are to introduce LFDA for the state-of-the-art i-vector-based LID tasks and analyze its advantages by comparing with LDA and NDA

methods. Detailed discussions of the comparison are given in Section 2.4. Experimental results in Section 3 indicate that LFDA is a more effective method than LDA and NDA methods for the i-vector-based LID tasks.

2. DISCRIMINANT ANALYSIS-BASED METHODS

2.1. Linear discriminant analysis

Linear discriminant analysis (LDA) is a very popular technique for feature selection and dimensionality reduction. It has the advantage of defining new axes by maximizing the discrimination between the variability of different classes while minimizing the intra-class variability.

Let $\mathbf{x}_i \in \Re^d (i = 1, 2, ..., n)$ be d-dimensional samples and $y_i \in \{1, 2, ..., c\}$ be associated class labels, where n is the number of samples and N is the number of classes. Let S_w and S_b be the within- and between-class scatter matrices defined as,

$$S_w = \sum_{i=1}^{c} \sum_{j=1}^{N_i} (\mathbf{x}_j - \boldsymbol{\mu}_i)(\mathbf{x}_j - \boldsymbol{\mu}_i)^T, \quad (1)$$

$$S_b = \sum_{i=1}^{c} N_i (\boldsymbol{\mu}_j - \boldsymbol{\mu})(\boldsymbol{\mu}_j - \boldsymbol{\mu})^T, \quad (2)$$

where N_i is the number and $\boldsymbol{\mu}_i$ is the mean of samples in class i, $\boldsymbol{\mu}$ is the mean of all samples, T denotes the transpose.

The LDA transformation matrix $(\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, ..., \boldsymbol{\varphi}_m)$ (where $\boldsymbol{\varphi}_i \in \boldsymbol{\varphi}$) can be obtained by selecting the generalized eigenvectors associated to the generalized eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_m$ of the following generalized eigenvalue problem:

$$S_b \boldsymbol{\varphi} = \lambda S_w \boldsymbol{\varphi}. \quad (3)$$

As we have discussed in Section 1, LDA is a parametric method and highly depends on the data distribution (Gaussian) and suffers from the relative low dimensional features. To improve LDA, several methods have been developed, such as NDA [8], subclass discriminant analysis (SDA) [16] and kernel Fisher discriminant (KFD) [17].

2.2. Nonparametric discriminant analysis

To overcome the mentioned limitations of LDA, the NDA technique [8, 13] was proposed which focuses on measuring the between-class scatter on a local basis using the k-NN rule. With this local rule, the NDA can model the multimodal data better than LDA. Recently, NDA [5] was successfully applied on LID task. The within-class scatter matrix of NDA has the same form as the LDA (i.e., Eq. 1). The between-class scatter matrix of NDA is defined as,

$$S_b^{\text{NDA}} = \sum_{i=1}^{c} \sum_{j=1, j\neq i}^{c} \sum_{l=1}^{N_i} \omega_l^{ij} (\mathbf{x}_l^i - M_l^{ij})(\mathbf{x}_l^i - M_l^{ij})^T, \quad (4)$$

where $\mathbf{x}_l^i$ denotes the l-th sample from class i, N_i is the sample number in class i, M_l^{ij} is the local mean of k-NN samples (totally K) for $\mathbf{x}_l^i$ from class j, and ω_l^{ij} is a weight function defined as,

$$\omega_l^{ij} = \frac{min\{d^\alpha(\mathbf{x}_l^i, \text{NN}_k(\mathbf{x}_l^i, i)), d^\alpha(\mathbf{x}_l^i, \text{NN}_k(\mathbf{x}_l^i, j))\}}{d^\alpha(\mathbf{x}_l^i, \text{NN}_k(\mathbf{x}_l^i, i)) + d^\alpha(\mathbf{x}_l^i, \text{NN}_k(\mathbf{x}_l^i, j))}, \quad (5)$$

where $\text{NN}_k(X_l^i, j)$ is the k-th nearest neighbor of $\mathbf{x}_l^i$ in class j, $d(.)$ is the Euclidean distance, and α is the parameter ranging from zero to infinity which controls the changing speed of the weight with respect to the distance ratio.

With the weight function ω^{ij}, the boundary samples were emphasized. This is because, for samples near the boundary, it approaches 0.5 and drops off to zero if the samples are far away from the boundary.

2.3. Local fisher discriminant analysis

Similar to NDA method, local Fisher discriminant analysis (LFDA) method [9] was proposed to overcome the poor performance of LDA when samples are in the same class from several separate clusters (i.e., multimodal). LFDA combines the idea of LDA and LPP to evaluate the within- and between-class scatter matrices in a local manner, by which class separability and local structure preservation could be attained at the same time. The within- and between-class scatter matrices of LFDA are defined as,

$$S_w^{\text{LFDA}} = \frac{1}{2} \sum_{i,j=1}^{n} \overline{A}_{i,j}^w (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T, \quad (6)$$

$$S_b^{\text{LFDA}} = \frac{1}{2} \sum_{i,j=1}^{n} \overline{A}_{i,j}^b (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T, \quad (7)$$

where

$$\overline{A}_{i,j}^w = \begin{cases} A_{i,j}/n_c & \text{if } y_i = y_j = c, \\ 0 & \text{if } y_i \neq y_j, \end{cases} \quad (8)$$

$$\overline{A}_{i,j}^b = \begin{cases} A_{i,j}(1/n - 1/n_c) & \text{if } y_i = y_j = c, \\ 1/n & \text{if } y_i \neq y_j, \end{cases} \quad (9)$$

where A is an affinity matrix, and $A_{i,j}$ means the affinity between sample $\mathbf{x}_i$ and $\mathbf{x}_j$ (pair-wise). The local scaling method [18] was used to determine the value of A:

$$A_{i,j} = \exp\left(\frac{-d^2(\mathbf{x}_i, \mathbf{x}_j)}{d(\mathbf{x}_i, \mathbf{x}_K) d(\mathbf{x}_j, \mathbf{x}_K)}\right), \quad (10)$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ is the distance from $\mathbf{x}_i$ to $\mathbf{x}_j$ and $\mathbf{x}_K$ of $d(\mathbf{x}_i, \mathbf{x}_K)$ is the K-th neighbor of point $\mathbf{x}_i$ in class i.

2.4. Comparison of LDA, NDA and LFDA

LDA, NDA and LFDA are both discriminant analysis methods, where LDA is a parametric method which can only extract up to $C - 1$ discriminative features. NDA and LFDA

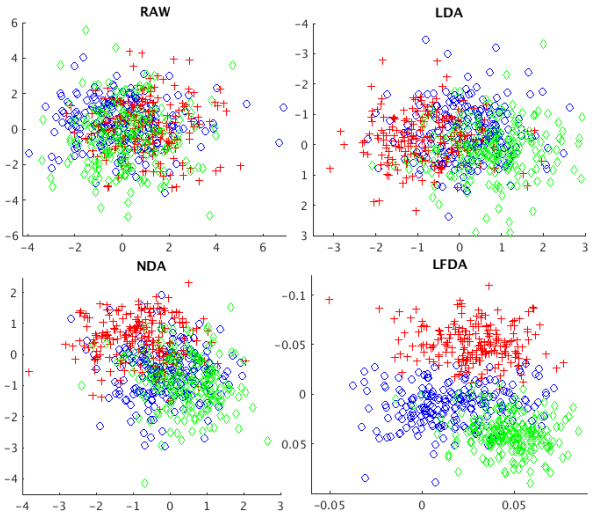

Fig. 1. Distribution of the top two PCA components for the RAW feature and LDA, NDA, LFDA transformed features. (three languages (blue: Arabic, red: Tatar, green: Oromo) of the training data in data set 2)

are full-rank methods which make use of all the samples in the construction of the within- and/or between-class scatter matrices instead of merely using the class centers. Therefore, more discriminative features can be extracted to improve the classification performance than LDA.

Both NDA and LFDA take the local structure of data into account to overcome multimodal data. NDA uses the mean of local k-NN samples to estimate the between-class scatter matrix. Instead of using the simple local mean, LFDA is a pair-wise method which estimates the contribution of the local k-NN samples, respectively. This leads to a more flexible and accurate estimation of the scatter matrices.

In Eqs. 8 and 9 of LFDA, we can see that $1/n - 1/n_c$ is negative while $1/n_c$ and $1/n$ are positive. This implies that if the data pairs in the same class are made close, the within-class scatter matrix becomes "small" and the between-class scatter matrix becomes "large". On the other hand, if the data pairs in different classes are apart from each other, the between-class scatter matrix becomes "large". In NDA, the boundary samples are emphasized with the weight function, however, samples in the same class are not taken into account on the within- and between-class scatter (i.e., $j \neq i$ in Eq. 4). From the above analysis, we can see that LFDA has more powerful capability than NDA on keeping the sample pairs in the same class close to each other.

Figure 1 shows an example of the distribution of RAW features and LDA, NDA, LFDA transformed features for three languages. This figure shows that LDA and NDA enjoy a similar distribution which is different from LFDA. The feature distribution with LFDA tends to be more localized and centralized (e.g., the blue and red points).

Table 1. Experimental data sets.(The number is the utterance number, language number is 50.)

Data set	Training	Dev.	Test
Set1	3738 (utt. length $\geq$ 30s)	2500	2500
Set2	10000 (utt. length $>$ 0)		

3. EXPERIMENT

In this section, experiments are conducted to evaluate the effectiveness of the proposed LID system.

3.1. Data and classifier

The data used in this study are based on the training data of NIST i-vector challenge[1] which were extracted using the same method in [3]. Each i-vector is a vector of 400 components. Segment durations have a very wide range from 1s to more than 800s. To develop our LID system, we reorganized the data to three groups: training, development and test data. These three group data were randomly selected without overlapping with each other. Because most of the LID tasks use longer utterance data as training data, e.g., 30 seconds, we prepared two sets of training data, one (set 1) only includes data with duration longer than 30s (around 70 samples for each class), the other (set 2) includes all the selected samples (200 samples for each class). The two data sets share the same development and test data (50 samples for each class). The details are shown in Table 1.

A discriminative SVM method with linear kernel was used as classifier following the state-of-the-art i-vector-based LID systems [4, 11]. 5-fold cross validation method was used to find the best value of the cost parameter of SVM in the model training step. The scoring metric used for the 2015 NIST i-vector challenge was used. Since, we only focus on the "close-set" problem, we rewrite the scoring metric as,

$$Cost = \frac{1}{c}\sum_{k=1}^{c} P_{error}(k), \tag{11}$$

where $P_{error}(k) = (\#errors\_class\_k/\#trials\_class\_k)$, and c is 50 in this study.

3.2. Parameters investigation

There are several parameters in both NDA and LFDA that need to be carefully selected. For NDA method, the number of k-NN samples K, parameter α in the weight function (Eq. 5) and the dimension of the generated features D need to be considered. For LFDA, K and D need to be selected. The cost parameter of SVM for each experiment was determined with 5-fold cross-validation method, respectively. The development data were used to investigate these parameters. Table

[1] https://lre.nist.gov/

Table 2. System performance (Cost×100) on development data with NDA (D=200, K=11) at different α.

Method&Data	0.5	1	2	3
NDA Set1	26.88	**26.36**	26.88	27.08
NDA Set2	18.76	**18.72**	**18.72**	19.04

Table 3. System performance (Cost×100) on development data with NDA (D=200, α=1) and LFDA (D=200) at different number of nearest neighbor samples K.

Method&Data	3	5	7	9	11	13
NDA Set1	26.48	26.48	**25.60**	26.76	26.36	26.96
NDA Set2	18.48	**18.44**	18.88	18.08	18.72	19.00
LFDA Set1	23.80	23.40	23.28	23.00	**22.84**	23.00
LFDA Set2	16.76	**16.52**	16.68	16.80	16.76	16.80

2 shows the results for parameter α which determines how rapid the decay in the weights of nearest neighbor samples occurs. These experiments were done by fixing D to 200 and K to 11. The results show that both on data set 1 and 2, the best performances were obtained when α was set to 1.

Table 3 shows the investigation on the number of K. Both NDA and LFDA methods are affected by the value of K. For data set 1, NDA and LFDA methods obtained the best results when K is 7 and 11 on data set 1, and 5 on data set 2.

Different from LDA, both NDA and LFDA are full-rank methods, therefore, higher dimensional features can be obtained with these two methods. Table 4 shows the results at different dimensions by fixing K to 11 and α to 1. NDA obtained the best results with 100-dimensional feature on data set 1 and 300-dimensional feature on data set 2, for LFDA, they were 200 and 400, respectively. From these results, we can see that both NDA and LFDA obtained the best results with a higher dimension. However, LDA can only obtain 49-dimensional discriminant feature, which maybe cause information loss, especially in a high-dimensional space with limited training samples.

3.3. Results on test data

Based on the results in Section 3.2, experiments were conducted on test data to compare LFDA with LDA and NDA methods. The parameters K, D and α were confirmed based on the best results in Tables 2, 3 and 4, i.e., (K, D, α) of NDA were (7, 100, 1) and (5, 300, 1) for data set 1 and 2, respectively; (K, D) of LFDA were (11, 200) and (5, 400) for data set 1 and 2, respectively. Cost parameter of SVM was learned with 5-fold cross validation method. Table 5 shows the results of these experiments; the results of original i-vector (RAW) are also included. From these results we can see that the generated features with LDA achieved better results than the RAW features on both data sets. The improvement benefits from the discriminant analysis of LDA by considering the label information which could improve the discriminative capability of the original i-vector feature.

Table 4. System performance (Cost×100) on development data with NDA (K=11, α=1) and LFDA (K=11) at different feature dimensions.

Method&Data	50	100	200	300	400
NDA Set1	27.72	**25.92**	26.36	27.00	26.72
NDA Set2	19.00	17.52	17.64	**17.28**	18.12
LFDA Set1	23.52	23.16	**22.84**	23.32	23.56
LFDA Set2	17.20	16.40	16.76	16.56	**16.08**

Table 5. System performance (Cost×100) with RAW, LDA, NDA and LFDA at best parameters on test data.

Data(Test)	RAW	LDA	NDA	LFDA
Set1	29.44	25.08	24.04	**22.96**
Set2	17.40	16.28	17.48	**15.60**

NDA achieved better results than both LDA and RAW on data set 1. However, no improvements were made on data set 2. Compared with LDA and NDA, LFDA obtained the best results on both data sets. The LFDA method obtained 8.45% and 4.49% relative improvements than the LDA and NDA methods, respectively, on data set 1. Even on data set 2, the LFDA method obtained 4.18% relative improvement than the LDA method. As we have discussed in Section 2.4, LFDA is a full-rank, pair-wise method which takes the local structure of the data into account very effectively. Compared with LDA, the full-rank method is very important for LID tasks because the number of target languages are often limited. The effective LFDA method can specifically bring benefit to the i-vector-based LID systems because the i-vectors suffer multimodal distribution due to short duration utterance or data degradation. Our results show the effectiveness of LFDA method for i-vector-based LID tasks.

4. CONCLUSIONS

The i-vector-based LID system obtained state-of-the-art performance for LID tasks. Since i-vector includes total variability factors, and suffers multimodal distribution due to short duration utterance or data degradation, it is important to apply an effective discriminant analysis method for the i-vector-based LID tasks. In this study, we introduced LFDA for the i-vector-based LID tasks. First, it is a full-rank method which can generate higher dimensional features. Second, it takes the local structure of the data into account with a pair-wise method, which leads to a more flexible and accurate estimation of the scatter matrices. Finally, LFDA has more powerful capability than NDA of keeping the sample pairs in the same class close to each other wisely. Our experimental results showed the effectiveness of LFDA for the i-vector-based LID tasks.

5. REFERENCES

[1] H. Li, B. Ma and K. A. Lee, "Spoken language recognition: From fundamentalsto practice," in *Proc. IEEE*, vol. 101, no. 5, pp. 1136-1159, 2013.

[2] C.-H. Lee, "Principles of spoken language recognition," in *Springer Handbook of Speech Processing and Speech Communication*, 2008.

[3] N. Dehak, P. Torres-Carrasquillo, D. Reynolds and R. Dehak, "Language recognition via ivectors and dimensionality reduction," in *Proc. Interspeech*, pp. 857-860, 2011.

[4] S. Novoselov, T. Pekhovsky and K. Simonchik, "STC speaker recognition system for the NIST i-vector challenge," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2014.

[5] S. O. Sadjadi, J. W. Pelecanos and S. Ganapathy, "Nearest neighbor discriminant analysis for language recognition," in *Int. Conf. ICASSP*, pp.4205-4209, 2015.

[6] Y. Song, B. Jiang, Y. Bao, S. Wei and L.-R. Dai, "I-vector representation based on bottleneck features for language identification," in *Electronics Letters*, vol. 49, no. 24, pp. 1569-1570, 2013.

[7] R. A. Fisher, "The use of multiple measurements in taxonomic problems," in *Annals of Eugenics*, vol. 7, no. 2, pp. 179-188, 1936.

[8] K. Fukunaga and J. Mantock, "Nonparametric discriminant analysis," in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 5, no. 6, pp. 671-678, 1983.

[9] M. Sugiyama, "Local Fisher discriminant analysis for supervised dimensionality reduction," in *Int. Conf. ICML*, pp. 905-912, 2006.

[10] P. Kenny, "Bayesian speaker verification with heavy tailed priors," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, 2010.

[11] M. McLaren, A. Lawson, Y. Lei and N. Scheffer, "Adaptive Gaussian backend for robust language identification," in *Proc. Interspeech*, pp. 84-88, 2013.

[12] X. He and P. Niyogi, "Locality preserving projections," in *NIPS*, vol. 16, 2004.

[13] Z. Li, D. Lin and X. Tang, "Nonparametric discriminant analysis for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 755-761, 2009.

[14] Q. Shen and R. Liu, "Face recognition based on LFDA and LS-SVM," in *Int. Conf. on Education Technology and Training*, pp.136-139, 2009.

[15] L. Wang, H. Ji and Y. Shi, "Face recognition using maximum local Fisher Discriminant Analysis," in *Int. Conf. on Image Processing (ICIP)*, pp.1737-1740, 2011.

[16] M. Zhu and A. M. Martinez, "Subclass Discriminant Analysis," in *PAMI*, vol. 28, no. 8, pp. 1274-1286, 2006.

[17] S. Mika, G. Rätsch and K.-R. Müller, "A Mathematical Programming Approach to the Kernel Fisher Algorithm," in *NIPS*, vol. 13, pp. 591-597, 2000.

[18] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *NIPS*, vol. 17, pp. 1601-1608, 2005.