Between-Class Covariance Correction For Linear Discriminant Analysis in Language Recognition

Abhinav Misra, Qian Zhang, Finnian Kelly, John H. L. Hansen

Center for Robust Speech Systems (CRSS)
Erik Jonsson School of Engineering & Computer Science
The University of Texas at Dallas (UTD), Richardson, Texas, USA
{abhinav.misra, qian.zhang, finnian.kelly, john.hansen}@utdallas.edu

Abstract

Linear Discriminant Analysis (LDA) is one of the most widely-used channel compensation techniques in current speaker and language recognition systems. In this study, we propose a technique of Between-Class Covariance Correction (BCC) to improve language recognition performance. This approach builds on the idea of Within-Class Covariance Correction (WCC), which was introduced as a means to compensate for mismatch between different development data-sets in speaker recognition. In BCC, we compute eigendirections representing the multi-modal distributions of language i-vectors, and show that incorporating these directions in LDA leads to an improvement in recognition performance. Considering each cluster in the multi-modal i-vector distribution as a separate class, the between- and within-cluster covariance matrices are used to update the global between-language covariance. This is in contrast to WCC, for which the within-class covariance is updated. Using the proposed method, a relative overall improvement of +8.4% Equal Error Rate (EER) is obtained on the 2015 NIST Language Recognition Evaluation (LRE) data. Our approach offers insights toward addressing the challenging problem of mismatch compensation, which has much wider applications in both speaker and language recognition.

1. Introduction

Recent developments in language recognition have focused on exploiting Deep Neural Network (DNN) based i-vector extraction methods [1]. However, after i-vectors have been extracted, there remains the need to apply channel compensation techniques prior to the scoring stage.

In this study, we focus on adapting Linear Discriminant Analysis (LDA) based channel compensation to improve overall system performance. In language recognition, LDA aims to compute a reduced set of dimensions onto which i-vectors can be projected, so that variability between same-language samples can be minimized while at the same time maximizing the variability between different-language samples. This is accomplished by maximizing the ratio of between-language covariance to within-language covariance. Sources of within-language variation can be different channels, speakers, acoustic environments or speaking styles. On the other hand, differences between languages occur mainly due to different phonetic contents.

NIST conducted a Language Recognition Evaluation (LRE) in 2015 [2], where they released data corresponding to twenty languages. All the data was from conversational telephone speech and broadcast narrowband speech, resulting in considerable within-language and between-language variability. Furthermore, languages were grouped together based on their phonetic similarities. There were a total of six clusters into which all the twenty languages were divided. The main motivation behind our approach is to use this additional information related to language clusters in order to improve the system performance.

In analyzing the data, we found that the distribution of the full pool of language i-vectors is multi-modal, with each mode corresponding to a separate language cluster. Similar observations have been made in recent studies on speaker recognition. In [3, 4, 5, 6], the authors have shown that based on the source of data, speaker i-vectors have a multi-modal distribution with each mode representing its respective source. In [3, 4], the authors propose a source normalization algorithm to mitigate the effect of this multi-modality over datasets. They compute a separate between-speaker covariance matrix for each distinct data-set and then take average of all the matrices. This essentially reduces the mismatch between the data-sets by centering them around a global mean. In [5], the author proposes Inter-Dataset Variability Compensation (IDVC) technique that removes the mismatch using Nuisance Attribute Projection (NAP). First a subspace is computed representing all different data-sets

This project was funded by AFRL under contract FA8750-12-1-0188 and partially by the University of Texas at Dallas from the Distinguished University Chair in Telecommunications Engineering held by J. H. L. Hansen.

and then NAP is used to remove that subspace as an i-vector pre-processing step. In [6], the authors estimate a between-dataset covariance that is later added to within-speaker covariance as Within-Class Covariance Correction (WCC). This additive term is weighted heavily so that eigendirections representing the data-set shift are completely removed from LDA computation.

In the case of language recognition, we want to increase the separation between different language clusters, rather than removing or reducing it. Hence, in this study, we propose computing the covariance of different language clusters with respect to a global mean, and then adding it during LDA as Between-Class Covariance Correction (BCC).

Additionally, since the focus of LRE 2015 was to separate different languages within the same language cluster, we also computed the covariance of each language with respect to its local cluster mean, and incorporated this as an additional term in BCC. A combination of between-language and within-language additions to BCC resulted in the best performance improvements.

The paper is organized as follows: Section II reviews the LDA algorithm and provides the theoretical framework for BCC, Section III describes the language recognition system used in our study, Section IV analyzes the results and Section V concludes the paper with discussion on future directions.

2. Linear Discriminant Analyses

LDA attempts to maximize the discrimination between different language i-vectors by finding a set of dimensions where between-language covariance is maximum while within-language covariance is minimum. This set of dimensions is obtained with the following procedure: First, between-language and within-language covariance matrices, S_b and S_w respectively, are computed as

$$S_b = \frac{1}{N}\sum_{l=1}^{L} N_l(\mu_l - \mu)(\mu_l - \mu)^t \tag{1}$$

$$S_w = \frac{1}{N}\sum_{l=1}^{L}\sum_{i=1}^{N_l}(\omega_i^l - \mu_l)(\omega_i^l - \mu_l)^t \tag{2}$$

The number of languages (or classes) is L. ω is an i-vector and N_l is the number of i-vectors corresponding to a language l. μ_l is mean of all the i-vectors belonging to language l, while μ is global mean of all the total number of N i-vectors present in the training data-set.

After computation of above scatter matrices, recall that we are looking for a projection that maximizes the ratio of between-class to within-class covariance. This is accomplished by finding a projection matrix that maximizes the following objective function [7]:

$$J(V) = \frac{V^t S_b V}{V^t S_w V} \tag{3}$$

Table 1: *Languages and their corresponding clusters*

Cluster Name	Corresponding languages
`Arabic`	Egyptian, Iraqi, Levantine, Maghrebi, Modern Standard
`Chinese`	Cantonese, Mandarin, Min, Wu
`English`	British, General American, Indian
`French`	West African, Haitian Creole
`Slavic`	Polish, Russian
`Iberian`	Caribbean Spanish, European Spanish, Latin American Spanish, Brazilian Portuguese

The above relationship is a Rayleigh quotient, and hence the solution V is the generalized eigenvectors of the following equation:

$$S_b V = \lambda S_w V \tag{4}$$

The optimal projection matrix is obtained by taking the columns representing the eigenvectors corresponding to the largest eigenvalues. The equation has $L-1$ non-zero eigenvalues, thus the optimal matrix can have a maximum of $L-1$ columns or eigenvectors.

2.1. Between-Class Covariance Correction

The 2015 NIST LRE data contains twenty languages divided into six clusters, as shown in Table 1. The evaluation plan for the LRE focused on distinguishing within-cluster languages, which are closely related, as can be observed from Table 1.

To visualize the relative distribution of languages and language clusters, we use Principal Component Analyses (PCA) [8]. First, we take the full set of training data and extract i-vectors corresponding to languages represented in all the six clusters. We then compute Principal Component Analyses (PCA) using a between-cluster covariance matrix. The top part of Fig 1 shows the language i-vectors projected through first two bases of PCA. It shows clearly different modes or clusters into which language i-vectors are distributed. It is quite apparent that each cluster has its own corresponding i-vector mean. Next, we want to see how LDA affects this distribution. Hence, we subsequently compute LDA and project the i-vectors through its first two eigendirections. We observe similar distribution as PCA, with the exception that some of the clusters start splitting up into bimodal distributions, particularly Chinese and Arabic. This happens as LDA attempts to maximize between-language variation and hence, further separates different languages within a cluster. This observation further motivated us to consider adding within-cluster covariance term to BCC.

In this study, our aim is to maximize the separation between different clusters, and additionally, between languages within a given cluster, so that LDA has more between-language discriminating ability. To accomplish

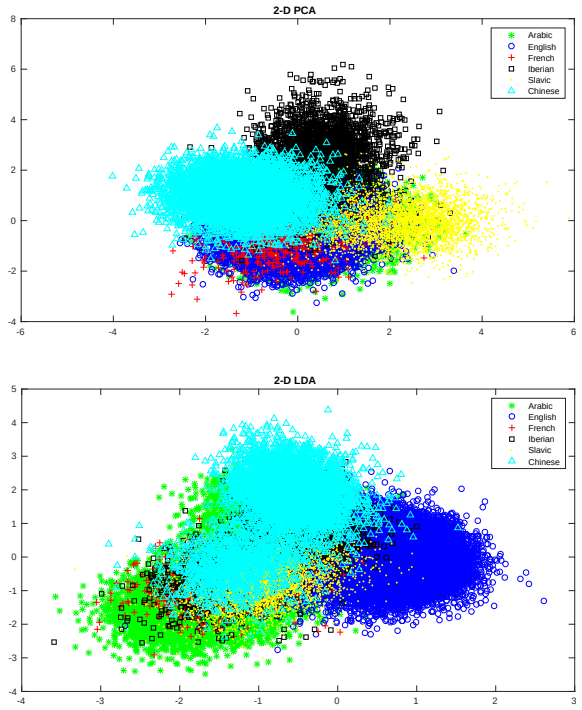

Figure 1: Projection of language i-vectors into the first two bases of PCA and LDA, estimated from between-cluster covariance.

that, we first compute between-cluster covariance matrix as:

$$S_{bcc} = \frac{1}{C}\sum_{c=1}^{C}(\mu_c - \mu)(\mu_c - \mu)^t, \quad (5)$$

where S_{bcc} is the between-cluster covariance, μ_c is the mean of cluster c, C is the total number of clusters, and μ is the global mean of all the language i-vectors.

Next, similar to equation 5, we compute within-cluster covariance matrix, S_{wcc} as:

$$S_{wcc} = \frac{1}{C}\sum_{c=1}^{C}\sum_{i=1}^{N_c}(\omega_i^c - \mu_c)(\omega_i^c - \mu_c)^t, \quad (6)$$

where, N_c is the total number of i-vectors belonging to cluster c.

After computing S_{bcc} and S_{wcc}, they are added to between-class covariance S_b of LDA as:

$$S_b^{new} = S_b + \alpha S_{bcc} + S_{wcc}, \quad (7)$$

where α is a scaling factor by which we weigh S_{bcc}.

We assume that the eigendirections represented by between-cluster covariance matrix has useful between-cluster discriminatory information. Similarly, within-cluster covariance matrix has useful between-language discriminatory information. Therefore, by scaling up and adding both of them to S_b, we make sure the Fisher Ratio in LDA for these directions is substantial. In our experiments, we observe that once α is chosen such that the order of magnitude of values in both S_b and S_{bcc} is the same, maximum improvement is obtained. This value of α was heuristically determined to be 60000. We also observe that the order of magnitude of S_{wcc} is already similar to S_b, so it doesn't need any scaling.

3. System Description

Figure 2, shows the overall diagram of the system used in our study. The following Sections describe the main components.

3.1. Training Data

All the system components use training data provided by NIST LRE 2015 Organizers. The data was provided in four parts as described below:

3.1.1. CALLHOME/CALLFRIEND

The first part consisted of CALLHOME and CALLFRIEND multi-lingual corpora collected by Linguistic Data Consortiium (LDC). It consists of telephone conversations, of fifteen to thirty minutes duration, between callers and their friends/relatives. The corpus contains Egyptian Arabic (95.4 hours), U.S. English (100 hours) and Mandarin Chinese (71.8 hours).

3.1.2. Previous LRE data

The second part of the training corpus consists of recent NIST data collected for LRE purposes, and data from past LRE test sets. It contains both telephone channel conversations as well as segments extracted from broadcast recordings containing narrow-band speech. Table 2, details all the languages present in this part of training corpus.

3.1.3. Switchboard-I

The third part of training data consists of release-2 of Phase-I of Switchboard telephone corpus. It contains telephone conversations between participants speaking U.S. English. There are a total of 2438 conversations of average 6-7 minutes duration, resulting in a total of 270 hours of data approximately.

3.1.4. Switchboard Cellular-II

The final part consists of Switchboard Cellular part-II telephone corpus. It contains a total of 2020 calls with participants talking in U.S. English. Each call is around 6-7 minutes duration, resulting in a total of 225 hours of data approximately. Table 2 shows the duration of data available for each language in training-set.

Table 2: *Details of languages present in NIST LRE15 training data*

Languages	Hours
Egyptian Arabic	95.4
Iraqi Arabic	37.2
Levantine Arabic	41.1
Modern Standard Arabic	3.7
Maghrebi Arabic	38.6
British English	0.5
U.S. English	600 (approx.)
Indian English	8.1
Haitian Creole French	2.7
West African French	7.7
Brazilian Portuguese	0.8
Polish	30.8
Russian	18.0
Carribean Spanish	26.9
European Spanish	8.1
Latin American Spanish	6.9
Mandarin Chinese	71.8
Cantonee Chinese	8.1
Wu Chinese	7.7
Min Chinese	3.4

All the training data files greater than one minute in duration were segmented into shorter segments of 5, 15 and 50 seconds. This was done to reproduce the evaluation data distribution of the LRE15 challenge. After segmentation, the files were divided into train and test sets in the ratio of around 6:4. That is, out of a total of 119,260 files, 73869 were part of train-set ($\sim 60\%$) and 45391 were part of test-set ($\sim 40\%$).

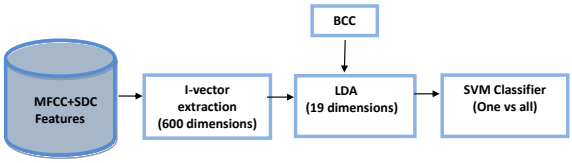

Figure 2: System Diagram showing the addition of Between Covariance Correction (BCC) to LDA .

3.2. Feature Extraction

The Kaldi toolkit [9] is used for the feature and i-vector extraction parts of the system. First, a speech activity detector based on log Mel-energy is applied over the segmented data files. Then, 39-dimensional MFCC features (13 + Δ + $\Delta\Delta$) are extracted using 20 ms analysis window and 10 ms frame shift. Shifted Delta Cepstral (SDC) [10] features are later computed and appended to the MFCC features.

Table 3: *Language recognition Results for each cluster.*

Cluster	EER(%) before BCC	EER(%) after BCC $\alpha = 60000$
Arabic	8.4972	8.068
English	2.5887	2.5354
French	**6.0538**	**4.0359**
Iberian	13.149	12.3589
Slavic	23.4586	23.0075
Chinese	6.4031	6.2245

Table 4: *Overall Language Recognition Results*

Performance Metric (%)	Before BCC	After BCC $\alpha = 60000$
EER	5.6729	5.1927
Accuracy	78.5839	81.276

3.3. I-vector Extraction

A Universal Background Model (UBM) with 256 mixtures is trained using the train-set features as extracted above. Then, using the same train-set features, a Total Variability (TV) matrix is trained. Finally, based on the UBM and TV matrix, 600-dimensional i-vectors are extracted for each utterance. The i-vectors are centered, whitened and length-normalized before LDA is applied to reduce their dimensions to 19 (number of classes -1).

3.4. SVM

For classification, a discriminative Support Vector Machine (SVM) classifier is trained using the reduced dimension i-vectors extracted as above. A 20 class SVM with a radial basis function (RBF) kernel is trained using LIBSVM [11]. Optimal SVM parameters are obtained via cross-validation. The output log-likelihood score is taken as the probability of each test i-vector given the target language class compared with all (19) non-target language classes (one vs all).

4. Results

Table 3 shows the language recognition performance of both the baseline system and the improved system, that is obtained after application of BCC. It can be observed that for the French cluster, there is a significant improvement in language recognition after BCC, with a relative improvement of +29.6% in Equal Error Rate (EER). For other clusters, there is not as significant a change in EER, although positive trends are observed. There is an overall relative improvement in system EER of +8.4%. As can be observed from table 4, system accuracy (ratio of correct language classifications to total number of trials) also improved by relative +3.42%.

A confusion matrix is presented in Figure 3, that

Figure 3: Confusion matrices showing classification counts of languages among different clusters

Table 5: *Results for French cluster*

Languages	Accuracy (%)	
	Before BCC	After BCC
Haitian Creole	93.07	93.84
West African	38.29	43.35

shows the classification counts of all the languages in different clusters. The numbers on the diagonal are correct classification counts, while off-diagonal elements represent misclassification counts. It can be observed that, after applying BCC, there is an absolute 5.89% increase in the correct classification counts for Arabic language cluster and absolute 2.28% increase in the correct classification counts for English language cluster. Additionally, motivated by the improvement obtained in language recognition for the languages corresponding to the French cluster, we also compute the number of misclassification errors for those languages. Table 5 shows the within-cluster errors for French. Both the French languages show an improvement in classification accuracy, as indicated by their cluster's EER performance.

5. Conclusion

In this paper, we used useful information relating to the multi-modal nature of language data to improve recognition performance on an LRE 2015 data-set.

We proposed a method of Between-Covariance Correction (BCC), for which we computed covariance matrices corresponding to between-cluster and within-cluster variability, and observed that by adding them to the Fisher ratio of LDA, an improvement in performance is obtained.

For future work, we intend to compute the eigendirections corresponding to between-cluster and within-cluster variabilities in a more effective manner to get further improvement. Right now, the performance of the system relies heavily on the scaling parameter α. Future work will focus more on finding better ways to optimize the addition of BCC without relying on any scaling parameter.

6. References

[1] F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *Signal Processing Letters, IEEE*, vol. 22, no. 10, pp. 1671–1675, Oct 2015.

[2] "The 2015 NIST language recognition evaluation plan (LRE15)," 2015, Available at `http://www.nist.gov/itl/iad/mig/upload/LRE15_EvalPlan_v23.pdf`.

[3] M. McLaren and D. van Leeuwen, "Source-normalized lda for robust speaker recognition using i-vectors from multiple speech sources," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 3, pp. 755–766, March 2012.

[4] M. McLaren and D. van Leeuwen, "Source-normalised-and-weighted lda for robust speaker recognition using i-vectors," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, May 2011, pp. 5456–5459.

[5] H. Aronowitz, "Inter dataset variability compensation for speaker recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 4002–4006.

[6] O. Glembek, J. Ma, P. Matejka, Bing Zhang, O. Plchot, L. Burget, and S. Matsoukas, "Domain adaptation via within-class covariance correction in i-vector based speaker recognition systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 4032–4036.

[7] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic Press, 1990, ch.10.

[8] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[9] D. Povey, A. Ghoshal, G. Boulianne, Burget L., O. Glembeck, N. Goel, and Hannemann M. et al., "The kaldi speech recognition toolkit," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2011.

[10] Pedro A. Torres-Carrasquillo, Elliot Singer, Mary A. Kohler, Richard J. Greene, Douglas A. Reynolds, and John R. Deller Jr., "Approaches to language identification using gaussian mixture models and shifted delta cepstral features," in *7th International Conference on Spoken Language Processing, ICSLP2002 - INTERSPEECH 2002, Denver, Colorado, USA, September 16-20, 2002*, 2002.

[11] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, 2011.