INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

UTD-CRSS Systems for 2016 NIST Speaker Recognition Evaluation

Chunlei Zhang, Fahimeh Bahmaninezhad, Shivesh Ranjan,
Chengzhu Yu, Navid Shokouhi, John H.L. Hansen

Center for Robust Speech Systems (CRSS), Erik Jonsson School of Engineering,
University of Texas at Dallas, Richardson, Texas, U.S.A.
{chunlei.zhang,fahimeh.bahmaninezhad,john.hansen}@utdallas.edu

Abstract

This study describes systems submitted by the Center for Robust Speech Systems (CRSS) from the University of Texas at Dallas (UTD) to the 2016 National Institute of Standards and Technology (NIST) Speaker Recognition Evaluation (SRE). We developed 4 UBM and DNN i-vector based speaker recognition systems with alternate data sets and feature representations. Given that the emphasis of the NIST SRE 2016 is on language mismatch between training and enrollment/test data, so-called domain mismatch, in our system development we focused on: (i) utilizing unlabeled in-domain data for centralizing i-vectors to alleviate the domain mismatch; (ii) selecting the proper data sets and optimizing configurations for training LDA/PLDA; (iii) introducing a newly proposed dimension reduction technique which incorporates unlabeled in-domain data before PLDA training; (iv) unsupervised speaker clustering of unlabeled data and using them alone or with previous SREs for PLDA training, and finally (v) score calibration using unlabeled data with "pseudo"speaker labels generated from speaker clustering. NIST evaluations show that our proposed methods were very successful for the given task.

Index Terms: NIST SRE, speaker recognition, domain mismatch, i-vector, speaker clustering

1. Introduction

As in previous SREs, the main task for 2016 NIST SRE is speaker recognition (i.e., to determine whether a specified target speaker is speaking during a given segment of speech). Compared with previous SRE challenges, there are some differences: (1) target speaker data is not distributed in advance like in SRE12; (2) fixed condition (using only specified data sets) is introduced, which is intended to encourage cross-system comparisons; (3) more duration variability is introduced in the test data; and finally (4) language mismatch between training (mainly English) and enrollment/test (non-English) data. All these new traits make this SRE very challenging, especially with limited labeled data in the fixed condition [1].

This paper describes how CRSS systems address these new challenges introduced in SRE16. The paper is organized as follows: Sec.2 describes several baseline systems focused on front-end level overview, including both datasets and feature representations; Sec.3 introduces several core techniques that we used in SRE16, including speaker clustering for unlabeled training data, discriminant analysis via support vectors (SVDA) [2] for dimension reduction and domain mismatch compensation, PLDA training with in-domain unlabeled data and "pseudo" speaker labels, score calibration and fusion strategies, etc. Sec.4 and Sec.5 details the configuration of each CRSS sub-system and the formation of CRSS final evaluation submissions to NIST; Sec.6 shows CRSS sub-system performance on development (DEV) and evaluation (EVAL) sets. Finally, conclusion and future work are summarized in Sec. 7.

2. CRSS baselines

We developed 4 baseline systems for SRE16, all consisting of i-vector based systems [3] but with different acoustic modeling (i.e., UBM or different DNN models [4, 5, 6]). For back-end, we mainly use LDA/SVDA to reduce the dimension of the i-vectors and PLDA [7] to calculate likelihood scores.

Table 1 summarizes the number of speakers, speech segments used for training UBM, total variability matrix (TV Matrix), LDA/SVDA and PLDA models as well as the statistics for the DEV and EVAL sets provided by NIST for the system development and evaluation purposes.

2.1. CRSS#1: UBM i-vector

This system is a modification of Kaldi (sre10/v1) [8]. 60 dimensional feature vectors for each frame is adopted here including 20 dimensional MFCC features appended with $\Delta + \Delta\Delta$. Unvoiced parts of the utterances are removed with energy based speech activity detection (SAD). For training 2048-mixture UBM and TV Matrix, SRE2004, 2005, 2006, 2008, telephone data of SRE 2010, Switchboard II phase 2,3 and Switchboard Cellular Part1 and Part2 (SWB) and Fisher English are used. Next, 600 dimensional i-vectors are extracted and their dimensions reduced to 580 with LDA. For training LDA/PLDA, only SRE 04-08 are used; in addition, speakers who have less than 4 utterances are filtered out. Also, unsupervised speaker clustering is performed (see Sec. 3.1 for the details of speaker clustering), 75 "pseudo" speaker labels for unlabeled minor data and 300 for unlabeled major data are generated. This speaker clustered in-domain data is then used separately to train PLDA and also score calibration in order to alleviate domain mismatch. Before PLDA scoring, mean subtraction is also applied. For SRE16 DEV trails, the mean i-vector is generated using only unlabeled minor data, while for SRE16 EVAL, the mean is calculated from unlabeled major data.

2.2. CRSS#2: SWB DNN i-vector

We developed a DNN i-vector system based on Kaldi (swbd/s5 & sre10/v2). In this system, a DNN acoustic model is used to generate the soft alignments for i-vector extraction. The DNN architecture has 6 fully connected hidden layers with 1024 nodes for each layer. A cross-entropy objective function is employed to estimate the posterior probabilities of 3178 senones. The ASR corpus which we used for training the DNN acoustic model is Switchboard. An 11-frame context of 39 dimensional ($\Delta + \Delta\Delta$) MFCC feature are projected into 40 dimensional using fMLLR transform for each utterance, which relies on a GMM-HMM decoding alignment.

The reason we apply the fMLLR feature here is that, by

http://dx.doi.org/10.21437/Interspeech.2017-555

Table 1: *Statistics of data used for modeling DNN, UBM, TV, LDA/SVDA/PLDA, and DEV/EVAL set enrollment and trials data.*

Sub-system	DNN		UBM/TV		LDA/SVDA/PLDA		Enrollment (DEV/EVAL)		Trials (DEV/EVAL)	
	Spkrs	Segments	Spkrs	Segments	Spkrs	Segments	Spkrs	Segments	Target	nonTarget
CRSS1	-	-	38110	89860	2921	37037	80/802	120/1202	4828/1986729	19312/1949666
CRSS2	4878	4878	5767	57517	2921	37037				
CRSS3	-	-	5756	57273	3794	36410				
CRSS4	1239	1239	5756	57273	3794	36410				

speaker normalization, we expect to acquire more accurate phonetic alignment in the following TV matrix training (see more details in [5]). After i-vector extraction, we apply similar strategies for the back-end such as LDA and PLDA, briefly described in the above section (as CRSS#1).

2.3. CRSS#3: UBM i-vector

An alternative UBM i-vector system is also adopted from Kaldi (sre10/v1). Similar to CRSS#1, we extract 60 dimension MFCC features within a 25ms window, with a shift size of 10ms. Non-speech frames are discarded using an energy-based SAD. 2048-mixture full covariance UBM and TV Matrix are trained using data collected from SRE2004, 2005, 2006, 2008 and Switchboard II phase 2,3 and Switchboard Cellular Part1 and Part2. At the back-end level, after extracting i-vectors, the global mean calculated from minor and major unlabeled data is subtracted from all i-vectors. After that, i-vectors are length-normalized [9] and their dimension reduced from 600 to 400 using LDA/SVDA. In some developed systems based on CRSS#3 configuration, we use both LDA and SVDA. First, SVDA reduces the dimension from 600 to 500; then, LDA is used to reduce the dimension to 400. Again, i-vectors are length-normalized. Finally, trial-based mean subtraction is used (the participant i-vectors in a trial are averaged and the value is subtracted from both i-vectors) and scores are calculated using PLDA. The front-end is trained with SWB and SRE04-08; however, the back-end only uses SRE04-08 and unlabeled training data. For back-end development, the MSR [10] toolkit was adopted and modified.

2.4. CRSS#4: Fisher English DNN i-vector

The last baseline is a DNN i-vector system modification of Kaldi (sre10/v2), which is based on the multisplice time delay DNN (TDNN) for acoustic modeling [11]. In this study, TDNN is trained with only a small portion of Fisher English data (1239 utterances). 40-dimensional f-bank features are utilized to train the TDNN model with six layers, the hidden layers have an input dimension of 350 and an output dimension is 3500. The softmax output layer computes the posteriors for 3859 triphone states. More details on the TDNN structure and training procedure are provided in [11]. After TDNN training, 20 MFCCs appended with ($\Delta + \Delta\Delta$) coefficients (overall 60 MFCCs) are employed for training TV matrix.

After 600-dim i-vector extraction, we apply similar strategies in the back-end level, including LDA/SVDA and PLDA, briefly described in the above section (as CRSS#3).

3. Core components in system development

In the fixed condition of SRE16, we are given an extensive amount of out-of-domain data (i.e., previous SREs, SWB, Fisher English etc). Only a small amount of in-domain data is available (without speaker labels), which makes existing techniques very difficult to apply with this so-called domain mismatch. In SRE16, participants were provided with unlabeled training data, which contains two subsets (i.e., unlabeled minor and unlabeled major). The unlabeled minor data set has 200 utterances, while the major set contains 2272 utterances. The minor set has two languages for the purpose of system development, while the major set contains two alternate languages corresponding to the final evaluation.

In this evaluation, several techniques are proposed to address the domain mismatch presented above.

3.1. Speaker clustering of unlabeled data

For compensating the domain mismatch, the use of unlabeled data becomes very important. There are several stages where we can use the unlabeled data, for example, LDA/PLDA training and score calibration, where more in-domain information is very likely to achieve a better result. It is very intuitive to perform speaker clustering of the unlabeled data, and then generate a "pseudo" speaker label for each utterance, similar to the method we used in 2015 NIST LRE i-vector challenge[12]. With these labels, we incorporate the in-domain information from unlabeled data to train LDA and PLDA. In fact, in the experiment, this simple operation improved the LDA/PLDA baseline performance for the DEV set.

To accomplish speaker clustering, we first train a gender identification using previous SRE data before speaker clustering, and then apply a simple K-means algorithm over the gender dependent subsets; finally, we pool two gender dependent subsets together. In the experiment, we found this can provide more accurate speaker clustering and thus more benefits to the following LDA and PLDA training stages.

3.2. Discriminant analysis via support vectors (SVDA)

Discriminant analysis via support vectors (SVDA) is a variation of LDA that only uses support vectors to calculate the between and within class covariance matrices. In contrast to LDA, SVDA captures the boundary of classes, and performs well for small sample size problems (i.e., when the dimensionality is greater than the sample size). The idea of using support vectors with discriminant analysis has been previously introduced in [13] which made significant improvement over LDA. In addition, the effectiveness of SVDA in i-vector/PLDA speaker recognition for NIST SRE2010 was studied in [2] previously for both long and short duration test utterances and achieved consistent improvement.

More specifically, LDA definition of class separation criterion will be optimized by the transformation matrix $\hat{A}$ as [14],

$$\hat{A} = \underset{A^T S_w A = I}{\operatorname{argmax}} [tr(A^T S_b A)], \tag{1}$$

where S_b and S_w are between class and within class covariance matrices. In traditional LDA, every sample of all classes participate in calculating these covariance matrices; however, for SVDA only the support vectors are used. The between class covariance matrix in SVDA is defined as,

$$S_b = \sum_{1 \le c_1 \le c_2 \le C} w_{c_1 c_2} w_{c_1 c_2}^T. \tag{2}$$

where $w_{c_1 c_2}$ is the optimal direction to separate the two classes c_1 and c_2 by a linear SVM (for calculating $w_{c_1 c_2}$ only support vectors of the two classes c_1 and c_2 are participating). If we

Table 2: *Description of the CRSS sub-systems. For systems which use SVDA, training data includes LDA data in addition to minor and major unlabeled data. For sub-systems 3 and 4, for DEV set, minor data is used in PLDA training; for EVAL set, major data is used.*

Sub system	i-vector	LDA/SVDA data	PLDA data	SVDA	LDA	Speaker Clustering	Filtering
1	CRSS 4	SRE 04-08	SRE 04-08	×	✓	×	×
2	CRSS 4	SRE 04-08/ + Minor, Major	SRE 04-08	✓	✓	×	×
3	CRSS 2	SRE 04-08, Minor, Major	SRE 04-08, Minor or Major	×	✓	✓	✓
4	CRSS 1	SRE 04-08, Minor, Major	SRE 04-08, Minor or Major	×	✓	✓	✓
5	CRSS 3	SRE 04-08	SRE 04-08	×	✓	×	×
6	CRSS 3	SRE 04-08/+ Minor, Major	SRE 04-08	✓	✓	×	×
7	CRSS 3	SRE 04-08/+ Minor, Major	SRE 04-08	✓	×	×	×

define $\hat{X} = [\hat{x}_1, \hat{x}_2, ..., \hat{x}_{\hat{N}}]$ to contain all support vectors and $\hat{N}$ to be their total number; then, the within class covariance matrix for SVDA will be formulated as,

$$S_w = \sum_{c=1}^{C} \sum_{i \in \hat{I}_c} (\hat{x}_i - \hat{\mu}_c)(\hat{x}_i - \hat{\mu}_c)^T, \quad (3)$$

with the index for support vectors in class c and their mean are represented by $\hat{I}_c$ and $\hat{\mu}_c$, respectively. Finally, similar to LDA, the optimum transformation $\hat{A}$ will contain the k eigenvectors corresponding to the k largest eigenvalues of $S_w^{-1} S_b$. More details on SVDA are provided in [2].

Two strategies can be adopted here for training the linear SVM in SVDA framework [15]: 1) one-versus-one and 2) one-versus-rest. We use the second approach here as it can use minor and major unlabeled data optimally. More specifically, for training the SVM classifier to separate one class against data from the remaining classes, minor and major unlabeled data are added to the rest class. Therefore, the class labels are not needed here. Details of data used for training SVDA are in Table 2.

3.3. Unlabeled data PLDA

To fully explore the information from the in-domain unlabeled data, we did an interesting experiment, which uses only the in-domain unlabeled data to train PLDA (however, SRE04-08 are used for training the LDA). To do that, we use the "pseudo" labels from speaker clustering. Surprisingly, PLDA with only 75 estimated speakers from 200 minor language i-vectors achieved 20.5% EER on the DEV experiment (using i-vectors from CRSS#1 baseline), this was an encouraging result. Meanwhile, if we add more data (i.e., 75 + 300 estimated speakers from 2472 i-vectors in minor and major languages) for PLDA training, the performance degraded from 20.5% to 26.3%. This observation suggests that out-of-domain language data is not always helpful to train a discriminative classifier, because in the view of DEV enrollment/test data, the major language data is also out-of-domain. We argue that even for a data-driven algorithm such as PLDA, choosing a proper data set to train the classifier is still essential.

Motivated by this, we believe the use of unlabeled data in the SRE16 evaluation data will be more beneficial. Compared with only 200 minor language utterances, the major language set has 2272 utterances. Although the speaker label is not given, we say the estimated label is still useful, and could probably perform better than 20.5% EER in the EVAL set.

3.4. Calibration and fusion

The CRSS calibration and fusion system is mainly based on the BOSARIS toolkit [16]. The PAV algorithm is used to create a calibration transformation matrix. We used two data sets for calibration. The first one is DEV data, where we use all the DEV trials information that NIST provided for system development to train the calibration system. Again, because in this SRE, the DEV and EVAL sets have totally different languages, it is not guaranteed that the calibration will work well for the final EVAL set. For this consideration, we created a new trial list to calibrate evaluation scores, and we used unlabeled data (both minor and major) with estimated speaker labels. We believe the score distribution of unlabeled data will be closer to that of evaluation.

After calibration, we fused our sub-systems for final submission. For system fusion, we employed a simple linear fusion system using logistic regression.

4. CRSS sub-systems

We developed 7 sub-systems from the 4 CRSS baselines that used SREs to train SVDA, LDA and PLDA. Also, as described above, we developed 4 sub-systems with just the unlabeled data PLDA idea. The details of each system concerning data and techniques used are listed in Table 2 and Table 3. More specifically, sub-systems 8, 9, 10, 11 are respectively share the same configuration as sub-systems 4, 3, 6, 2; however, only unlabeled data are used to train PLDA.

Table 3: *Sub-systems using just unlabeled data to train PLDA.*

Sub-system	i-vector	SVDA	LDA
8	CRSS 1	×	✓
9	CRSS 2	×	✓
10	CRSS 3	✓	✓
11	CRSS 4	✓	✓

Table 4: *CRSS submission for NIST SRE2016.*

Submission	sub-systems	Calibration Data	Fusion
Primary	1-7	DEV+Unlabeled	LR
Contrastive1	1-7	DEV	LR
Contrastive2	1-11	DEV+Unlabeled	LR

5. CRSS submissions

The final submissions of CRSS is the fusion of several sub-systems. In the final submission, we tried different system combinations as well as different calibration strategies. We submit a 1-7 sub-systems fusion with DEV+Unlabeled data for score calibration as our primary submission. To test our hypothesis that PLDA training using only unlabeled data will benefit for the EVAL set, we submitted this as a contrastive submission. All these combinations make our final submissions to SRE16.

6. Performance of CRSS submissions on SRE16 data

Table 5, 6, and 7 show the equal error rate (EER), minimum Cprimary (min-Cprimary) and actual Cprimary (act-Cprimary) costs for single systems and fusion systems using NIST scoring software on both DEV and EVAL set.

Table 5: *Single systems scores. The equalized and unequalized scores are separated with "/". Three values for act-Cprimary use DEV, Unlabeled, and DEV+Unlabeled data for calibration respectively.*

sub-system	DEV EER	DEV min-Cprimary	DEV act-Cprimary	EVAL EER	EVAL min-Cprimary	EVAL act-Cprimary
1	17.14 / 18.7	0.768 / 0.779	0.768/0.779 0.881 / 0.891 0.812 / 0.822	14.93 / 14.93	0.846 / 0.859	1.286/1.282 0.858 / 0.878 0.931 / 0.946
2	17.05 / 18.87	0.755 / 0.757	0.768 / 0.769 0.794 / 0.8 0.777 / 0.775	12.94 / 13.26	0.766 / 0.777	0.799 / 0.848 0.776 / 0.813 0.854 / 0.909
3	17.84 / 18.41	0.754 / 0.734	0.754 / 0.734 0.834 / 0.826 0.787 / 0.766	15.08 / 14.37	0.837 / 0.833	0.999 / 1.109 0.838 / 0.879 0.902 / 0.984
4	17.17 / 17.5	0.719 / 0.694	0.722 / 0.694 0.82 / 0.812 0.769 / 0.746	14.15 / 13.52	0.826 / 0.816	1.29 / 1.433 0.831 / 0.867 0.905 / 0.973
5	15.59 / 16.08	0.701 / 0.671	0.709 / 0.671 0.812 / 0.813 0.746 / 0.726	12.42 / 12.68	0.797 / 0.806	1.593 / 1.683 0.819 / 0.859 0.998 / 1.061
6	15.58 / 15.95	0.679 / 0.629	0.688 / 0.629 0.744 / 0.735 0.694 / 0.647	10.66 / 10.95	0.698 / 0.697	0.933 / 0.999 0.7 / 0.746 0.813 / 0.873
7	15.53 / 16.63	0.685 / 0.658	0.686 / 0.658 0.775 / 0.77 0.697 / 0.672	10.91 / 11.25	0.719 / 0.719	0.83 / 0.892 0.733 / 0.768 0.788 / 0.846

Table 6: *Scores for single systems that only use unlabeled data for training PLDA. The equalized and unequalized scores are separated with "/". For the calibration DEV+Unlabeled data are used.*

sub-system	DEV EER	DEV min-Cprimary	DEV act-Cprimary	EVAL EER	EVAL min-Cprimary	EVAL act-Cprimary
8	29.72 / 26.15	0.898 / 0.908	0.92 / 0.933	20.93 / 21.05	0.895 / 0.895	0.902 / 0.971
9	29.48 / 26.84	0.901 / 0.9	0.917 / 0.914	21.66 / 22	0.918 / 0.927	0.919 / 0.942
10	26.48 / 24.96	0.943 / 0.954	0.956 /0.96	20.34 / 20.82	0.956 / 0.959	0.957 / 0.961
11	27.01 / 26.22	0.921 / 0.933	0.935 / 0.945	21.41 / 22.22	0.96 / 0.97	0.963 / 0.972

Table 7: *Fusion scores for submitted systems as well as scores when only Unlabeled data are used for calibration. The equalized and unequalized scores are separated with "/".*

Submission	DEV EER	DEV min-Cprimary	DEV act-Cprimary	EVAL EER	EVAL min-Cprimary	EVAL act-Cprimary
Primary	14.24 / 14.98	0.59 / 0.562	0.612 / 0.58	9.37 / 9.43	0.646 / 0.638	0.708 / 0.806
Contrastive1	13.81 / 14.66	0.585 / 0.554	0.589 / 0.559	9.41 / 9.49	0.675 / 0.663	0.869 / 0.993
Contrastive2	14.13 / 14.89	0.601 / 0.562	0.617 / 0.577	9.36 / 9.42	0.647 / 0.638	0.702 / 0.807
OnlyUnlabeled	14.27 / 15.04	0.592 / 0.562	0.618 / 0.589	9.35 / 9.43	0.645 / 0.638	0.686 / 0.777

As we can see from Table 5, the UBM i-vector with SVDA (sub-system 6) consistently outperforms other configurations on both DEV and EVAL set. The result indicates the effectiveness of SVDA in compensating for domain mismatch, especially when SVDA incorporates unlabeled data to find the support vectors discriminatively.

Although single systems using unlabeled data PLDA did not achieve good performance on DEV set, reasonable results (if not top level) can be found on EVAL (Table 6), and fusion with CRSS Primary submission slightly improves the overall performance shown as CRSS Contrastive2 in Table 7. This encourages us to explore more on unlabeled data to benefit the speaker recognition performance.

Because of the domain mismatch in SRE16, score calibration becomes essential to achieve a low actual cost. In Table 7, we can see that calibration using the in-domain score is important (CRSS Contrastive1), which leads to a good performance on the DEV set. However, for the EVAL set, DEV score calibration is no longer effective (also because of domain mismatch). As we proposed in Sec.3.4, calibration using scores from unlabeled data is very effective to achieve a low actual cost. The result can be found in CRSS Primary and Contrastive2 (score calibration with DEV+Unlabeled data). One additional step is that, we find that score calibration with only unlabeled data can be more beneficial to find a low actual cost. We post this result as a post-evaluation analysis.

7. Conclusion and future work

In this paper, we described single and fused systems submitted from CRSS to NIST SRE 2016 challenge. Domain mismatch is the dominant challenge introduces in this task. We proposed different strategies to use unlabeled in-domain data. Performance on both DEV and EVAL sets prove that our proposed methods are contributing an important role for compensating domain mismatch. In continuing this work, we will focus more on using unlabeled in-domain data. For SVDA, we just used a 1-vs-rest strategy; replacing that with 1-vs-1 strategy could be helpful for unbalanced problem. In our unsupervised clustering approach the number of estimated speakers were not close to the actual number of classes; therefore, more accurate clustering approaches can help in calibration and PLDA training. In addition, introducing duration and noise uncertainty for the i-vectors may improve the performance of the system further.

8. Acknowledgement

We would like to thank Dr. Kong Aik Lee for organizing I4U meetings, and other I4U group members for sharing ideas and insights during I4U meetings. We would like to thank Qian Zhang, Abhinav Misra, Dr. Finnian Kelly and other CRSS colleagues for their insights and helpful discussions in the development of the systems. We want to thank Dr. Gang Liu previously with CRSS, now with Alibaba Group for his helpful advice.

9. References

[1] "NIST 2016 speaker recognition evaluation plan," https://www.nist.gov/sites/default/files/documents/itl/iad/mig/SRE16_Eval_Plan_V1-0.pdf, 2016.

[2] F. Bahmaninezhad and J. H. Hanesn, "i-vector/PLDA speaker recognition using support vectors with discriminant analysis," in *IEEE ICASSP*, 2017.

[3] 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.

[4] 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)*. IEEE, 2014, pp. 1695–1699.

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

[6] S. Sadjadi, J. Pelecanos, and S. Ganapathy, "The ibm speaker recognition system: Recent advances and error analysis," in *Proceedings of INTERSPEECH*, 2016.

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

[8] 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 *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE, 2011.

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

[10] S. O. Sadjadi, M. Slaney, and L. Heck, "Msr identity toolbox v1. 0: A matlab toolbox for speaker-recognition research," *Speech and Language Processing Technical Committee Newsletter*, vol. 1, no. 4, 2013.

[11] 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)*. IEEE, 2015, pp. 92–97.

[12] C. Yu, C. Zhang, S. Ranjan, Q. Zhang, A. Misra, F. Kelly, and J. H. Hansen, "Utd-crss system for the nist 2015 language recognition i-vector machine learning challenge," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5835–5839.

[13] S. Gu, Y. Tan, and X. He, "Discriminant analysis via support vectors," *Neurocomputing*, vol. 73, no. 10, pp. 1669–1675, 2010.

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

[15] C. Chang and C. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

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