The CRSS systems for the 2010 NIST speaker recognition evaluation

Yun Lei, Taufiq Hasan, Jun-Won Suh, Abhijeet Sangwan, Hynek Bořil,
Gang Liu, Keith Godin, Chi Zhang and John H. L. Hansen

Center for Robust Speech Systems (CRSS)
Erik Jonsson School of Engineering & Computer Science
The University of Texas at Dallas, USA

***Abstract*—This document briefly describes the systems submitted by the Center for Robust Speech Systems (CRSS) from The University of Texas at Dallas (UTD) in the 2010 NIST Speaker Recognition Evaluation. Our systems primarily use factor analysis as feature extractor [1] and support vector machine (SVM) classification framework. Our main focus in the evaluation is on the telephone trials in the core condition and 10 second train-test condition. Novel elements in our system include a supervised probabilistic principal component analysis (SPPCA) based approach for factor analysis, and an algorithm for optimal selection of the negative samples for training the SVM.**

I. System components

In this section, we describe the specific blocks used for building our systems. Later, we will discuss how these parts were joined together to form our sub-systems.

A. Feature Extraction

The acoustic features used in this submission were identical for all the subsystems. A 60-dimension feature (19 MFCC with log energy $+ \Delta + \Delta\Delta$) using a 25 ms analysis window with 10 ms shift, filtered by feature warping using a 3-s sliding window is employed [2]. To remove the silence frames, a Hungarian phoneme recognizer [3] and an energy based voice activity detection (VAD) method were used. A block diagram of our feature extraction system is shown in Fig.2.

B. UBM Training

Two gender dependent UBMs with 1024 mixtures were trained on the NIST 2004, 2005, 2006 SRE enrollment data. We used the HTK toolkit for training. 20 iterations per mixture split was used. These UBMs were later used for factor analysis training and the joint factor analysis (JFA) based system.

C. Factor analysis

We used two different modeling approaches for our factor analysis training, probabilistic principal component analysis (PPCA) and supervised probabilistic principal component analysis (SPPCA). For both methods, the Switchboard II Phase 2 and 3, Switchboard Cellular Part 1 and 2, and the NIST 2004, 2005, 2006 SRE enrollment data were used as the training data. In total, 400 factors were used.

1) PPCA method: This is the classical probabilistic principal component analysis (PPCA) approach for the factor analysis model [4], as utilized in [5], [6], [1], and was employed here as one of the two techniques.

2) SPPCA method: The supervised probabilistic principal component analysis (SPPCA) model [7] is proposed to integrate the speaker label information into the factor analysis approach using PPCA. The latent factor from the proposed model is believed to be more discriminative than the one from the PPCA model. We have performed extensive experiments on this model, in combination with different types of inter-session compensation techniques in the back-end for this evaluation.

D. Channel Compensation

We have used three different channel compensation techniques. In most of the cases, they were applied in pairs. They are discussed below.

1) Linear discriminant analysis (LDA): LDA is a common technique for dimensionality reduction and widely used in pattern recognition applications. NIST 2004, 2005, 2006 SRE enrollment data are used as the training data for LDA.

2) Nuisance attribute projection (NAP): The NAP algorithm [8] is used to find a projection matrix intended to remove the nuisance direction from the feature vectors. The NAP matrix was also trained using the same factor analysis dataset, obtained from the NIST 2004, 2005, 2006 SRE enrollment data.

3) Within Class Covariance Normalization (WCCN): The WCCN method [9] is based on linear separation between target and impostor speakers using one versus all decision. NIST 2004, 2005, 2006 SRE enrollment data are used for training the WCCN matrix.

E. Support Vector machine (SVM) training

The SVMs were trained using the SVM-light toolkit [10]. The background dataset consists of NIST SRE 2004, 2005, 2006, and the Switchboard II Phase 2 and 3, Switchboard Cellular Part 1 and 2, with a total of 12,763 utterances. We have used a novel algorithm for finding the best negative examples for one of our SVM based systems. A similar idea is considered in [11], [12] where a certain number of negative examples were chosen based on system performance evaluation. The novel aspect of our method is that, the difference of two SVMs trained on different number of background speakers is measured for each enrollment speaker [13]. Using this difference information, the best speakers are selected as

the background data for each model. This method, unlike that in [11], [12], is not dependent on the system performance and thus can be applied for unseen data.

F. Score Normalization

NIST SRE 2005 data was used for t-norm to normalize the decision score obtained with the SVM system [14]. The t-norm model is trained with a leave-one-out method, and the same speaker utterances are excluded to train its' own t-model. No z-norm was used in the SVM case.

G. Score Fusion

Two methods were investigated for training the weights in a linear score fusion technique. Score fusion software based on Brummer et. al.'s FoCal toolkit[1] implemented the linear logistic regression (LLR) method to train the fusion weights, as well as a direct mean and variance-normalization method. The score fusion software was also designed to automate the process of choosing a fusion method and fused systems for the best DCF value.

II. The sub-systems

In this section, we describe the subsystems that were used in our submission. In total, we have developed five subsystems, four of which are SVM based and one of them is GMM based. All of the SVM systems use the factor analysis front-end. A brief description of the subsystems are given below.

A. SVM-SPPCA-LDA

This sub-system uses the factor analysis front-end features as the input to the SVM classifier [1]. SPPCA algorithm for training the factor analysis, LDA and WCCN was used for channel compensation and t-norm for score normalization.

B. SVM-PPCA-LDA

This sub-system uses the factor analysis front-end features as the input to the SVM classifier [1]. PPCA algorithm is used for training the factor analysis and LDA and WCCN was used for channel compensation.

C. SVM-SPPCA-NAP

Similar to the SVM-SPPCA-LDA system except this system uses NAP instead of LDA for channel compensation. NIST 04 and 05 data were used for impostors for the SVM training. The impostor selection algorithm was not used in this case.

D. SVM-PPCA-NAP

Similar to the previous SVM-PPCA-LDA system except this one uses NAP in place of LDA for channel compensation. NIST 04 and 05 data were used for impostors for the SVM training. Also, the impostor selection algorithm was not used in this case.

[1] http://www.dsp.sun.ac.za/~nbrummer/focal/index.htm

E. GMM-UBM-JFA

The joint factor analysis (JFA) system is a commonly used framework for speaker verification [5]. In this system, 300 speaker factors and 100 channel factors was used. Eigenvoice matrix V was trained on Switchboard II, Phases 2 and 3; Switchboard Cellular, Part 1 and 2; NIST 2005 and 2006 data. Eigenchannel matrix U was trained on NIST 2004, 2005, and 2006 data; diagonal matrix D was trained on NIST 2004 data.

F. SVM-PPCA-LDA-BG

This sub-system uses the factor analysis front end-features as the input to the SVM classifier [1]. SPPCA algorithm for training the factor analysis, LDA and WCCN was used for channel compensation and t-norm for score normalization. The new background speaker selection algorithm was used in this subsystem for SVM training.

G. Other developments

We have also implemented an ASR based system for this evaluation. Following [15], ASR trained on Switchboard is used to generate MLLR transform matrices for speaker verification tokens. The ASR employs PLP front-end and feature warping [2]. A global MLLR transform and broad phone-group transforms are estimated by the system. PCA is applied to reduce the MLLR features' dimension. MLLR features are then use as input to the SVM classifier. We also explored PMVDR [16] features for speaker recognition in a GMM-MAP framework [17]. Due to lack of time and the magnitude of the SRE 2010 evaluation we could not submit results for these sub-systems.

III. Development strategy

In order to incorporate the new DCF parameters in our system, we have generated new trial lists consistent with the SRE 2010 trials. In this years evaluation, the P_{target} parameter was set to 0.001 instead of 0.01 as in SRE 2008. Thus it is more meaningful to use a trial set that has a much fewer number of target trials compared to nontarget trials. We ran extensive experiments to find optimal parameters for our sub-systems, including LDA dimension and number of impostors (selected using our new algorithm) for SVM training. The newly generated trials were used in these experiments.

IV. Results

In this section we present some of the results that we have obtained in the SRE 2008, tel-tel, core condition. The results are shown in Table I. The DCF values are computed using the new parameters, $C_{\text{Miss}} = 1$, $C_{\text{FA}} = 1$ and $P_{\text{target}} = 0.001$ and normalized with the value $C_{\text{Default}} = 0.001$ as required.

V. The CRSS submissions

This section describes the system results that were actually submitted. NIST allows 3 submissions per train-test condition. These are the submissions that were delivered.

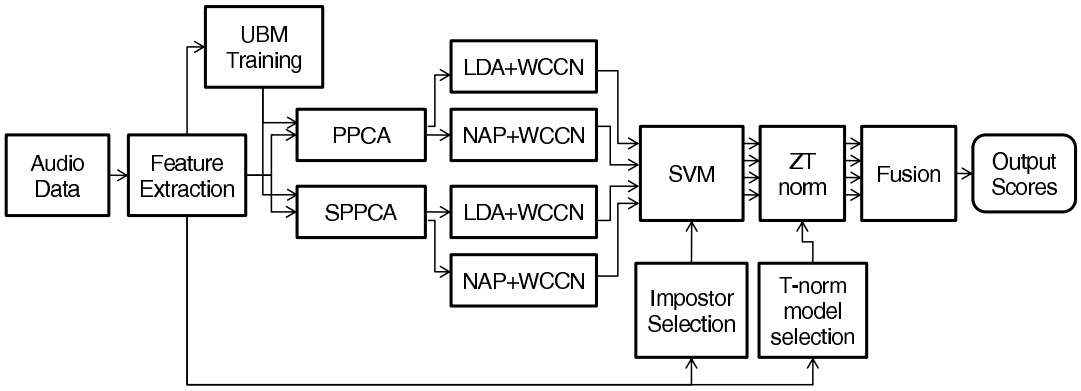

Fig. 1. A conceptual block diagram of the CRSS core system submissions. This shows a fusion of four of our SVM based sub-systems.

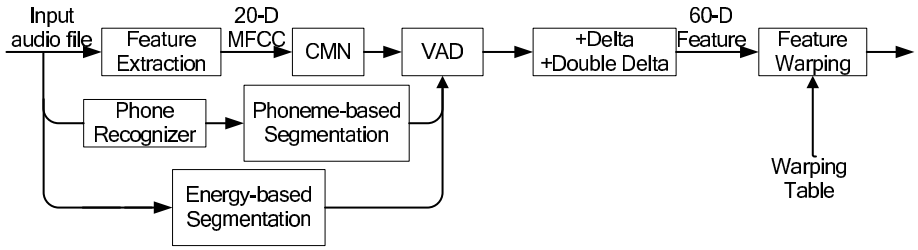

Fig. 2. A block diagram of the feature extraction block of the CRSS systems.

TABLE I
EER AND DCF PERFORMANCE OF THE SUBSYSTEMS IN THE NIST 2008 SRE, 5M TRAIN-TEST, TEL-TEL SUB-CONDITION TRIALS.

	System	Male		Female	
		EER (%)	DCF	EER (%)	DCF
1	SVM-PPCA-LDA	3.1235	0.262	3.2739	0.336
2	SVM-SPPCA-LDA	3.1256	0.309	3.2970	0.534
3	SVM-PPCA-NAP	3.1430	0.321	3.6406	0.386
4	SVM-SPPCA-NAP	3.1235	0.345	4.5486	0.511
5	GMM-UBM-JFA	3.1256	0.682	5.2024	0.638
6	SVM-PPCA-LDA-BG	3.1256	0.231	3.2366	0.351

1) CRSS Primary-Core System 1: This is a fusion of subsystems SVM-PPCA-LDA, SVM-PPCA-NAP and in Section II submitted as CRSS_1_core_core_primary_llr. We used linear logistic regression for training the weights for fusion and the FOCAL toolkit was used.

2) CRSS Alternate-Core System 2: This is a fusion of subsystems SVM-PPCA-LDA, SVM-PPCA-NAP, SVM-SPPCA-LDA, SVM-SPPCA-NAP, GMM-UBM-JFA and SVM-PPCA-LDA-BG. This is submitted as CRSS_2_core_core_primary_llr. We used linear logistic regression for training the weights for fusion. A conceptual block diagram for this fusion system is given in

3) CRSS Alternate-Core System 3: This is a fusion of subsystems SVM-PPCA-LDA, SVM-PPCA-NAP, SVM-SPPCA-LDA, SVM-SPPCA-NAP and SVM-PPCA-LDA with background selection for SVM. This is submitted as CRSS_3_core_core_primary_llr. We used linear logistic regression for training the weights for fusion.

4) CRSS Primary-10sec System 1: This is the SVM-PPCA-LDA system run on the 10sec train and test condition. Submitted as CRSS_1_10sec_10sec_primary_llr.

5) CRSS Alternate-10sec System 2: This is a fusion of the SVM-PPCA-LDA and SVM-PPCA-NAP systems run on the 10sec train and test condition. Fusion was performed using the FOCAL toolkit. Submitted as CRSS_2_10sec_10sec_alternate_llr.

VI. COMPUTATIONAL RESOURCES

The speaker ID system was implemented on our CRSS high-performance Rocks computing cluster running the CentOS Linux distribution. The cluster comprises 18 HP Intel Quad-Core Xeon 2.33 GHz CPU's, yielding 72 CPU cores. A total of 126 GB RAM is available internally on the system. A 4 TB external RAID disk array is attached to the cluster by means of the storage area network (SAN). The array is connected with the cluster nodes through a 1 Gbit Ethernet switch.

VII. CPU EXECUTION TIME

The CPU execution times for the SVM systems are considerably fast assuming that the UBM and factor analysis matrices are trained beforehand. Time required for training on a 5 minute utterance is 6.2771 minutes assuming a single CPU, which gives a real time factor (RTF) of 1.2554. For testing each 5 minute segment, it took 4.6034 minutes which is gives an RTF of 0.9207.

REFERENCES

[1] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front-end Factor Analysis for Speaker Verification," *submitted to IEEE Transaction on Audio, Speech and Language Processing*.

[2] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. 2001: A Speaker Odyssey*, 2001, pp. 213–218.

[3] P. Schwarz, P. Matejka, and J. Cernocky, "Hierarchical structures of neural networks for phoneme recognition," in *Proc. IEEE ICASSP 2006*, vol. 1, May 2006, pp. I–I.

[4] M. Tipping and C. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural computation*, vol. 11, no. 2, pp. 443–482, 1999.

[5] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 4, pp. 1435–1447, May 2007.

[6] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 980–988, July 2008.

[7] Y. Lei and J. H. L. Hansen, "Speaker recognition using supervised probabilistic principal component analysis," in *Proc. Interspeech'10 (Submitted)*, 2010.

[8] W. Campbell, D. Sturim, D. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP 2006*, vol. 1, Toulouse, France, May 2006.

[9] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Ninth International Conference on Spoken Language Processing*. Citeseer, 2006.

[10] T. Joachims, "Making large-scale support vector machine learning practical, Advances in kernel methods: support vector learning," 1999.

[11] M. McLaren, B. Baker, R. Vogt, and S. Sridharan, "Improved SVM speaker verification through data-driven background dataset collection," *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4041–4044, 2009.

[12] ——, "Exploiting Multiple Feature Sets In Data-Driven Impostor," *Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4434–4437, 2010.

[13] J. Suh, Y. Lei, and J. H. L. Hansen, "Best background data selection in svm speaker recognition for new diverse evaluation data sets," in *Proc. Interspeech'10 (Submitted)*, 2010.

[14] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 42–54, 2000.

[15] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman, "MLLR transforms as features in speaker recognition," in *Ninth European Conference on Speech Communication and Technology*. ISCA, 2005.

[16] U. Yapanel and J. Hansen, "A new perceptually motivated MVDR-based acoustic front-end (PMVDR) for robust automatic speech recognition," *Speech Communication*, vol. 50, no. 2, pp. 142–152, 2008.

[17] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models,," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19 – 41, 2000.