The CRSS systems for the 2010 NIST speaker recognition evaluation

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

***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 Workshop. Our systems primarily use factor analysis as feature extractor [1] and the support vector machine (SVM) classifier. 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 optimal set of negative sample selection algorithm for training the SVM.**

I. System components

In this section, we will describe the different 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 tool for training. 20 iterations per mixture split was used. These UBMs were later used for factor analysis training and the JFA 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 was 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].

2) SPPCA method: The supervised probabilistic principal component analysis (SPPCA) model 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 extensively experimented on this model, in combination with different types of intersession 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 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 [7] 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, that is the NIST 2004, 2005, 2006 SRE enrollment data.

3) Within class covariance normalization (WCCN): The WCCN method [8] 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 SVMlite toolkit. The background dataset consists of NIST SRE 2004, 2005, 2006, and the Switchboard II Phase 2 and 3, Switchboard Cellular Part 1 and 2 as total of 12,763 utterances. We have used a novel algorithm for finding the best negative examples for our SVMs. A similar idea is considered in [9], [10] where a certain number of negative examples were chosen based on system performance evaluation. In our method, the difference of two SVMs trained on different number of background speakers is measured for each enrollment speaker which is Using this difference information, the best speakers are selected as the background data for each model. This method, unlike in [9], [10], is not dependent on the system performance and thus can be applied in unseen data.

F. Score normalization

NIST SRE 2005 data was used for t-norm to normalize the decision score obtained with the SVM system [11]. The t-norm

model is trained with leave-one-out method, and same speaker utterances are excluded to train 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 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. Best impostor selection algorithm was incorporated in this subsystem and t-norm for score normalization.

C. SVM-SPPCA-NAP

Similar to the SVM-SPPCA-LDA system except this system uses NAP 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 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.

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. Other developments

We have also implemented an ASR based system for this evaluation. Following [12], 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. Due to lack of time and the magnitude of the SRE 2010 evaluation we could not submit results for this 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. 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 we have made.

1) CRSS Primary-Core: This is a fusion of all the subsystems (1-5) mentioned 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 Primary-10sec: This is the SVM-PPCA-LDA system run on the 10sec train and test condition. Submitted as CRSS_1_10sec_10sec_primary_llr.

V. Computational resources

The speaker ID system was implemented on the 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.

VI. 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.

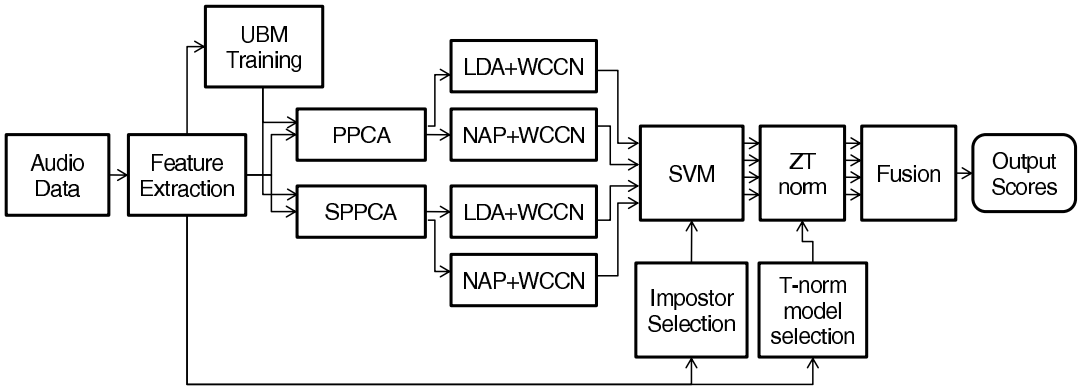

Fig. 1. A block diagram of the CRSS primary submission. This is a fusion of the four SVM systems.

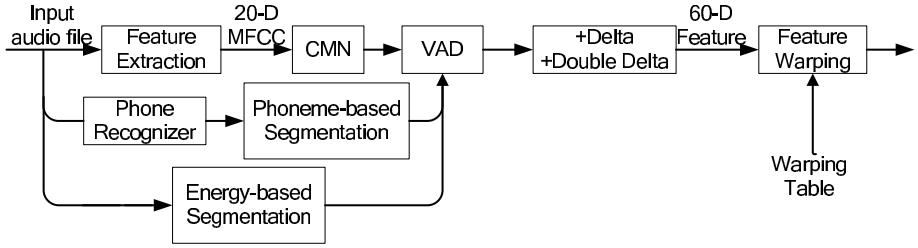

Fig. 2. A block diagram of the feature extraction block of the CRSS systems.

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] 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.

[8] 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.

[9] 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.

[10] ——, "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.

[11] 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.

[12] 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.