IDIAP RESEARCH REPORT

IDIAP SUBMISSION TO THE NIST SRE 2016 SPEAKER RECOGNITION EVALUATION

Srikanth Madikeri Subhadeep Dey Marc Ferras
Petr Motlicek Ivan Himawan

Idiap-RR-32-2016

DECEMBER 2016

Centre du Parc, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

IDIAP SUBMISSION TO THE NIST SRE 2016 SPEAKER RECOGNITION EVALUATION

Srikanth Madikeri[1], *Subhadeep Dey*[1,2], *Marc Ferras*[1], *Petr Motlicek*[1] *and Ivan Himawan*

[1] Idiap Research Institute, Martigny, Switzerland
[2] École polytechnique fédérale de Lausanne, Lausanne, Switzerland
{msrikanth, sdey, mferras, pmotlic, ihimawan}@idiap.ch

ABSTRACT

Idiap has made one submission to the fixed condition of the NIST SRE 2016. It consists of two gender-dependent i-vector systems that use posteriors from a Universal Background Model and a Deep Neural Network, respectively, whose scores have been fused via logistic regression. Both systems use Linear Discriminant Analysis (LDA) for i-vector post-processing and Probabilistic LDA for inference. The entire system was implemented using the Kaldi toolkit. The speech/non-speech senone posteriors from a DNN forward pass were used to segment the data for Voice Activity Detection. The gender-dependent PLDA models were trained on a subset of past SRE data and unsupervisedly adapted to the unlabelled development data provided for the SRE'16.

1. INTRODUCTION

Our systems are developed based on the state-of-the-art i-vector framework for speaker recognition [1], targetting the fixed condition of the NIST SRE 2016 protocol only. We follow the standard chain of blocks in both the front-end and back-end of the system. The inter-speaker variability of i-vectors is retained and/or other variabilities removed using techniques such as Linear Discriminant Analysis (LDA), Within Class Covariance Normalization (WCCN) and PLDA that provide better discriminability amongst speakers [2]. After applying such techniques, i-vectors are assumed to represent the speaker information in the original recording.

In our system, two versions of the standard i-vector framework are employed: an i-vector system that uses the conventional UBM/GMM as implemented in [1, 3, 4, 5] and another i-vector system that computes sufficient statistics based on the DNN system trained for Automatic Speech Recognition as presented in [6]. The two systems are described in Section 2. The data splits to train the systems are then given in Section 3. The results on the NIST SRE 2016 development set are provided in Section 4.

2. I-VECTOR PLDA SYSTEM

The i-vector extractor projects Gaussian mean supervectors on a low-dimensional subspace called *total variability space* (TVS) [1]. The variability model underlying i-vector extraction is

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{w}, \tag{1}$$

where $\mathbf{s}$ is the supervector adapted with respect to a Universal Background Model-Gaussian Mixture Model (GMM-UBM) from a speech recording. The vector $\mathbf{m}$ is the mean of the supervectors,$\mathbf{T}$ is the matrix with its columns spanning the total variability subspace and $\mathbf{w}$ is the low-dimensional i-vector representation. In the above given model, the i-vector is assumed to have standard Normal distribution as prior.

The i-vectors obtained from a speech utterance are further projected onto a discriminative space using LDA, WCCN [7, 1], mean subtraction, length normalization ([8]) and PLDA [2, 9], which together form the back-end of the i-vector system. Using PLDA parameters two i-vectors can be compared as belonging to the same class or as belonging to two different classes, thus generating a simple log likelihood ratio to score a pair of speech utterances.

The standard i-vector system was recently implemented for the Kaldi toolkit [10]. An implementation of PLDA based on [9] available in the Kaldi toolkit [11] is used. The PLDA model parameters are later adapted unsupervisedly to the NIST SRE 2016 development set.

Similar to the I-vector PLDA system, we also used a DNN trained for ASR to model the acoustic subspaces usually modelled with a GMM. That is, to estimate the sufficient statistics to extract an i-vector for a speaker, we utilize the posteriors at the output of the forward pass of a DNN. The UBM parameters are estimated based on these posteriors [6]. The rest of the front-end and back-end remain the same as in the system described in 2.

3. SYSTEM DEVELOPMENT

As our submissions target only the fixed condition, our entire development set was restricted to: Fisher English Corpus

(Parts I and II), Switchboard Cellular (Parts I and II), Switchboard Phase II (Parts I, II and III), NIST SREs 04, 05, 06, 08 and 10. For simplicity, we will refer to the NIST SREs mentioned earlier as *preSRE16*. In addition, we also used the unlabelled part of NIST SRE 2016 to optimize our system performances on the labelled part of the same dataset.

For the GMM-UBM i-vector system, the entire Fisher, Switchboard and *preSRE16* corpora were used to train the UBM and T-matrix. The PLDA was trained on only the *preSRE16* and was adapted to the unlabelled SRE16 corpus.

The DNN for ASR was trained on both parts of the Fisher English corpora. Speaker adaptation techniques such as fMLLR was not used.

All our systems are gender dependent. To train a gender identification engine, we developed an i-vector system by pooling all data from the Fisher corpus. This system does not use any discriminative training technique in the back-end. The i-vectors for the unlabelled SRE 2016 data are then clustered using K-means cluster with K=2. The i-vectors belonging to the two clusters are then averaged. The averaged i-vectors are compared against the labelled i-vectors in the SRE 2016 set and the gender labels are assigned accordingly. Further, these gender-labelled averaged i-vectors are used on the evaluation set to identify the speaker's gender.

3.1. Feature configuration

The front-end used 20 MFCC features along with delta and acceleration parameters, extracted every 10 ms using a window of 30 ms (as used by systems such as [12]). They were further processed through a short term Gaussianization module ([13]) with a context of 300 frames. All systems presented use the same feature configuration.

3.2. Baseline GMM-UBM I-vector system

Gender-dependent GMM-UBM with 2048 components and i-vector extractors of 500 dimensions were trained. The i-vector dimension was reduced to 350 after LDA, followed by mean subtraction and length normalization before being scored using PLDA. Means were separately obtained for the *preSRE16* and SRE 16 data sets. For the latter, the unlabelled SRE 16 set was used for system optimization and the entire SRE 16 development set was used for system evaluation.

The Kaldi toolkit [11] was used for LDA training, PLDA training and adaptation. A standard i-vector extractor was implemented for Kaldi as well [10], based on the baseline system described in [3].

3.3. DNN I-vector system

The DNN system trained for ASR was bootstrapped from a HMM/GMM system trained on the Fisher datasets. The input to the DNNs are 540 dimensional vectors obtained by stacking 9 MFCC feature vectors and the output classes are senone probabilities. We used the Kaldi toolkit to train a DNN with 6 hidden layers with 2'000 sigmoid units per layer and softmax units at the output. The DNN parameters were initialized with stacked Restricted Boltzmann Machine (RBM) that are pretrained in a greedy layer-wise fashion [14, 15]. The number of senone states was automatically derived by the tree-clustering algorithm that was constrained to have around 2k states in order to be comparable with the number of GMM-UBMcomponents.

Table 2. Time taken to extract and compare two i-vectors for the GMM-UBM and DNN i-vector based systems.

System	Time taken (s)
GMM-UBM	12.6
DNN i-vector	22.6
Fusion system	35.2

4. EXPERIMENTS

In this section, we report our results on the part of the NIST SRE 16 development set available for system optimization. We also report the time taken to evaluate each trial on an average.

4.1. System performance

We first report the performance of individual systems followed by the performance of the fused systems. All results are presented in Table 1. In general, the performance of the male systems are significantly better than that of the female systems. The DNN-based systems perform poorer than the GMM-UBM system. However, the fusion of the GMM-UBM and the DNN-based systems provide significant improvements. For the male system, system fusion reduces the equalized EER from 10.1% of the GMM-UBM system to 9.2%, which is equivalent to a relative improvement of 8.9%. Similarly, the relative improvement for the female system is 5.3%. Combining the results from the female and male systems reduces the overall unequalized actual DCF.

4.2. Time requirements

In this section, we report the time required to estimate an i-vector and compare two i-vectors in the two types of i-vector systems presented. The time taken for each stage of the i-vector extraction process are presented in Table 2. The system time information reported by the $time$ command in the current Linux systems are reported. The values are averaged over 100 experiments performed on the NIST SRE 2016 development set. The system is run with a single thread on a Intel Core i7 5930K system with 32 GB RAM with Network File System mounted disks.

Table 1. Results on the development set of NIST SRE 2016 dataset for all systems presented. EER: Equal Error Rate, DCF: Decision Cost Function, minDCF: minimum DCF

System	Equalized			Unequalized		
	EER (%)	minDCF	actual DCF	EER (%)	minDCF	actual DCF
GMM-UBM male	10.1	0.6075	0.8087	11.6	0.5626	0.8588
GMM-UBM female	16.8	0.7381	0.7785	17.4	0.7238	0.8956
SI-DNN male	10.3	0.5751	0.7576	11.6	0.5572	0.8768
SI-DNN female	20.4	0.8311	0.8703	21.0	0.8275	0.9364
Fusion male	9.2	0.5441	0.5661	10.0	0.5062	0.7595
Fusion female	15.9	0.7350	0.7504	16.6	0.7037	0.8652
Fusion male+female	12.6	0.6485	0.6582	13.2	0.6100	0.6247

5. SUMMARY

The Idiap submission to the NIST SRE 2016 evaluation was presented. Two gender-dependent systems based on the state-of-the-art i-vector speaker recognition framework were used: a GMM-UBM based system and a DNN-based system. The GMM-UBM system performed considerably better than the DNN-based system. Significant improvements were obtained after fusing the two systems.

6. ACKNOWLEDGEMENT

This work was supported by project Diarizing Massive Amounts of Heterogeneous Audio (DIMHA) and EU FP7 project Speaker Identification Integrated Project (SIIP). The authors would like to thank Mathew Magimai-Doss for his valuable comments on the paper.

7. REFERENCES

[1] Najim Dehak, Patrick Kenny, Rda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Tran. on Audio, Speech and Language Processing*, pp. 788–798, 2011.

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

[3] O Glembek et al., "Simplification and optimization of i-vector extraction," 2011, pp. 4516–4519, In Proc. of ICASSP.

[4] Srikanth R Madikeri, "A hybrid factor analysis and probabilistic pca-based system for dictionary learning and encoding for robust speaker recognition.," in *Odyssey*, 2012, pp. 14–20.

[5] Petr Motlicek, Subhadeep Dey, Srikanth Madikeri, and Lukas Burget, "Employment of subspace gaussian mixture models in speaker recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4445–4449.

[6] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1695–1699.

[7] Richard O Duda, Peter E Hart, and David G Stork, *Pattern classification*, John Wiley & Sons, 2012.

[8] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," August 2011, pp. 249–252, In Proc. of Interspeech.

[9] Sergey Ioffe, "Probabilistic linear discriminant analysis," in *Computer Vision–ECCV 2006*, pp. 531–542. Springer, 2006.

[10] Srikanth Madikeri, Subhadeep Dey, Petr Motlicek, and Marc Ferras, "Implementation of the standard i-vector system for the kaldi speech recognition toolkit," Idiap-RR Idiap-RR-26-2016, Idiap, 10 2016.

[11] D. Povey, A. Ghoshal, et al., "The kaldi speech recognition toolkit," in *In Proc. of ASRU 2011*, December 2011.

[12] Pavel Matejka, Ondrej Glembek, Ondrej Novotny, Oldrich Plchot, Frantisek Grezl, Lukas Burget, and Jan Cernocky, "Analysis of dnn approaches to speaker identification," *Proc. IEEE ICASSP, Shanghai, China*, pp. 5100–5104, 2016.

[13] Jason Pelecanos and Sridha Sridharan, "Feature warping for robust speaker verification," 2001.

[14] George E Dahl, Dong Yu, Li Deng, and Alex Acero, "Context-dependent pre-trained deep neural networks

for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30–42, 2012.

[15] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio, "Why does unsupervised pre-training help deep learning?," *The Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.