IDIAP RESEARCH REPORT

IDIAP SUBMISSION TO THE NIST SRE 2018 SPEAKER RECOGNITION EVALUATION

Srikanth Madikeri Seyyed Saeed Sarfjoo
Petr Motlicek Sébastien Marcel

Idiap-RR-17-2019

NOVEMBER 2019

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 2018 SPEAKER RECOGNITION EVALUATION

Srikanth Madikeri, Seyyed Saeed Sarfjoo, Petr Motlicek, and Sebastien Marcel

Idiap Research Institute, Martigny, Switzerland
{msrikanth, ssarfjoo, petr.motlicek, marcel}@idiap.ch

ABSTRACT

Idiap has made one submission to the fixed condition of the NIST SRE 2018. It consists of three systems: a gender-dependent i-vector system, a gender-independent x-vector system and a gender-independent Deep Neural Network (DNN)/i-vector system. The acoustic model for the DNN/i-vector system was trained using Lattice-Free Maximum Mutual Information (LF-MMI) criterion. The back-end for all the systems consists of the conventional Linear Discriminant Analysis (LDA) projection followed by Probabilistic LDA (PLDA) scoring for inference. The PLDA was also adapted unsupervisedly using the unlabelled data provided as a part of the development set. The entire system was implemented using the Kaldi toolkit.

1. INTRODUCTION

Our systems are developed based on two frameworks for speaker recognition: the i-vector framework [1] and the x-vector framework. In both frameworks, the back-end of the systems remains the same. Two versions of the standard i-vector framework are employed: an i-vector system that uses the conventional UBM/GMM as implemented in [1, 2, 3, 4] and another i-vector system that computes sufficient statistics based on the DNN system trained for Automatic Speech Recognition as presented in [5]. The two systems are described in Section 2 and 3. The x-vector system setup is given in Section 4. The results on the NIST SRE 2018 development set are provided in Section 5.

2. I-VECTOR PLDA SYSTEM

The i-vector PLDA systems have been part of the state-of-the-art speaker recognition since its introduction in [1]. In NIST SRE 2016 evaluations, GMM-UBM based i-vector systems performed better than the DNN/i-vector systems [6] primarily due to language mismatch [7]. As the scenario in the current evaluation is similar, we trained a gender-dependent GMM-UBM i-vector. However, as opposed to identifying the gender of the enrolled and test speaker, we obtain scores from both systems (male and female) and fuse them together with the scores from other systems. Both i-vector based systems used in the paper were trained using our implementation of the standard i-vector framework implemented for the Kaldi toolkit [8]. PLDA models are trained using the Kaldi toolkit as well [9]. The PLDA model parameters are then adapted in an unsupervised way to the NIST SRE 2018 unlabelled development set.

The front-end used 20 Mel Frequency Cepstral Coefficient (MFCC) features along with delta and acceleration parameters with 30 ms window and 10 ms frame shift [4]. Short term Gaussianization was applied with a context of 300 frames [10].

Gender-dependent GMM-UBMs with 2'048 components and i-vector extractors of 500 dimensions were trained. The ivector dimension was reduced to 350 after LDA, followed by mean subtraction and length normalization before being scored using PLDA. The following data was used to train the GMM-UBM and i-vector extractors: Fisher English, Switchboard Cellular, NIST SRE 2004, 2005, 2006 and 2008. In addition, NIST SRE 2010 was used to train LDA and PLDA.

3. DNN I-VECTOR PLDA SYSTEM

As stated in Section 2, the primary system developed at Idiap for Speaker Recognition (SRE) NIST 2018 is built around the i-vector approach, modeling a speech recording by projecting its acoustic features onto a low-dimensional representation.

As an interesting extension of conventional i-vector framework, in [5], the GMM-UBM components were replaced by the output states of a DNN-based hybrid Automatic Speech Recognition (ASR) system. The sufficient statistics were estimated using the context-dependent phone posterior probabilities estimated at the output of the DNN. The work has demonstrated that a well-defined acoustic space can significantly improve speaker modeling, as opposed to unsupervised training of conventional GMM-UBM components. It is assumed that the improvements in modeling are due to: (i) well-defined nature of DNN output states, and (ii) high alignment accuracy of the discriminative DNN classifier.

In our recent work [11], we experimented with different types of acoustic modeling techniques (primarily developed for ASR tasks), such as GMM-HMM, SGMM-HMM, DNN-HMM and its variants to generate better alignments when building the UBM model. Also employment of a Language

Model (LM) to more reliably estimate phone posteriors at the output of the ASR decoder was explored. Since the SRE performance is related to the sparsity of the phone posteriors estimated at the output of ASR decoder, due to the violation of the Gaussianity assumptions in LDA and PLDA space, a countermeasure was proposed based on adjustment of different scale factors involved in ASR decoding.

This section describes a state-of-the-art acoustic modeling technique applied in ASR - Lattice-Free (LF) version of the Maximum Mutual Information (MMI) criterion - to develop a Time-Delay Neural Network (TDNN) based neural network. The phone posteriors estimated using this network are then used to compute sufficient statistics to build the TDNN-UBM based i-vector system for NIST 2018 SRE task.

Similar to the original LF-MMI work [12], we used the HMM topology allowing to traverse the HMM in one frame. The transition probabilities are set to a constant value (0.5). We used only 408 PDFs in the final tree, which makes the subsequent UBM training fast. As the denominator in MMI criterion, an un-pruned 4-gram phone-level LM is used. The subsampled TDNN is used with a configuration specified in [12]. The input is composed by set of 13 dimensional MFCCs and 100 dimensional online i-vectors, followed by 1'024 dimensional ReLU components. The output is interpreted as the log of pseudo-likelihoods w.r.t. 408 PDF classes. The training did not apply any speed-perturbation technique (i.e. to augment training data as proposed in [13]), as we used relatively large amount of training resources. We also did not use high-resolution features in the TDNN training.

3.1. Experimental setup and datasets

For acoustic modeling, we used 13 dimensional MFCCs, standardly extracted every 10 ms using Kaldi framework [14]. The MFCCs were extended by 100 dimensional kaldi-online-ivectors extracted at the same frame-rate. Estimated posterior probabilities (obtained through a forward-pass) were converted into the Kaldi alignment archives.

The second feature stream was used for UBM/T-matrix computation. We used 20 dimensional MFCCs along with delta and acceleration coefficients extracted every 10 ms using a window of 30 ms generated using HTK toolkit. The 60 dimensional MFCCs were further processed through a short-term Gaussianization module [10] with a context of 300 frames. This type of MFCCs was shown to significantly improve SRE performance although its use for ASR acoustic model training is less efficient than conventional MFCCs [11].

The male and female part of Fisher English Part I and II data was used to train the TDNN model and subsequent TDNN-UBM and T-matrix under LF-MMI criterion. Similar to Section 2, estimated i-vectors using TDNN-UBM model were projected using LDA and passed to the back-end classifier (PLDA). Both LDA and PLDA parameters were trained with the datasets similar to Section 2. Voice activity detector to segment the development data was trained on Fisher English Part I.

The TDNN model was trained using a standard CMU dictionary with around 42 K words.

4. X-VECTOR SYSTEM

This system is mostly based on the x-vector system described in [15]. X-vector is an DNN embedding which models the speakers in both frame and segment level. For sequence modeling in frame level, this model uses a small temporal context centered at the current frame. It assumes for deeper layers, with splicing the output of the previous layers as input, that this architecture can model the sequence of frames from the larger context. In this structure after three layers, the total context of 15 frames is observed. The number of hidden units in each frame-level layer is 512, except layer 5 which is of zie 1'500. Layers 4 and 5 are fully connected layers and do not use any temporal context. To reach segment-level layer from the frame-level layers, statistical pooling is used which aggregates all T frame-level outputs in the fifth layer, computing the mean and standard deviation to reach the 3'000 dimension segment-level vector. Segment-level architecture contains two fully connected layers with 512 hidden units following the softmax layer with N dimension which corresponds to the number of target speakers (i.e. in the training data). The activation functions in this architecture are all rectified linear units (ReLUs). Final x-vector is extracted before applying the activation function in the first segment-level layer.

Similar to Section 2, the x-vectors obtained for each speech utterance are centered, and projected using LDA [1]. LDA of dimension 150 was used, based on tuning the parameters on the training set. After the dimensionality reduction, the x-vector representations are length-normalized [16] and modeled by PLDA [17]. For score normalization, although adaptive s-norm [18] showed significant improvement in NIST SRE 2016 set [19], based on the result on development set of NIST SRE 2018, s-norm was used as score normalization method.

4.1. Datasets

Majority of training data is in English language which consists of telephone, microphone, and audio from video recordings. All wide-band audio was downsampled to 8 kHz. For training the x-vector model, Switchboard dataset (SWBD), main NIST dataset (SRE), and Voxceleb dataset (VCELEB) were used. SWBD contains Switchboard 2 Phases 1, 2, and 3 as well as Switchboard Cellular parts 1, and 2. In total, the SWBD dataset contains about 28 K recordings from 2.6 K speakers. The SRE dataset consists of NIST SREs from 2004 to 2010 along with Mixer 6 and contains about 63 K recordings from 4.4 K speakers. VCELEB contains data

from Voxceleb 1, and 2. Both datasets consist of videos from celebrity speakers. Voxceleb 1 consists of 153'516 utterances from 1'251 speakers and Voxceleb 2 consists of 112'8246 utterances from 6'112 speakers.

To increase the amount and diversity of the existing training data, SRE and SWBD datasets were augmented with additive noise and reverberation. For reverberation and noise, RIR, and MUSAN datasets were used, respectively. RIR is the collection of room impulse responses measured in the different room sizes. The MUSAN dataset, consists of over 900 noise samples, 42 hours of music from various genres and 60 hours of speech from twelve languages. Both MUSAN and RIR datasets are freely available[1]. The strategy for augmenting the data is similar to the ideas mentioned in an original x-vector paper [15]. In addition to clean speech samples, the augmented version of the speech samples mixed with some noise, randomly chosen from four different categories, is added to the dataset. These noise categories contain *babble, music, noise,* and *reverb* which are speech, music, noise, and room impulse response, respectively. In the first three categories, the selected noises from MUSAN dataset are added to the original speech in different SNR levels. In the last category the training recording is artificially reverberated via convolution with simulated RIRs.

4.2. Experimental Setup

After down-sampling the speech data to 8 kHz, 23 dimensional mel frequency cepstral coefficients (MFCCs) were extracted with 25 ms window of speech data with 10 ms frameshift. Band-pass filtering was applied between 20 to 3700 Hz. Log of energy was added to the feature vector and these features were mean-normalized over a sliding window of up to 3 seconds. Energy-based voice activity detection (VAD) was used to removing the non-speech frames. For training the x-vector, chunk size of speech frames were chosen between 200 to 400 frames. In extraction time, chunk size of 100 seconds (10'000 frames) with minimum size of 250 ms was used, while for longer utterances, the average x-vector from input chunks was computed.

In these experiments, as the VCELEB dataset contains more than 1.2 M utterances, we did not perform data augmentation. The x-vector system was trained on combination of VCELEB and augmented version of SWBD and SRE datasets. The PLDA classifiers were trained on just augmented version of SRE. In addition to the aforementioned datasets, the unlabeled part of development set of NIST SRE 2018 was used for PLDA adaptation and score normalization. The rest of the development set was used for initial evaluation and selecting the score normalization method.

[1] http://www.openslr.org

5. EXPERIMENTS

In this section, we report our results on the part of the NIST SRE 18 development set available for system optimization. We also report the time taken to evaluate each trial on an average.

5.1. System performance

As mentioned above, all systems are evaluated on the test set provided with NIST SRE 2018 development. The same test set is used to tune the results, tune the fusion weights and calibrate our systems. The results are presented in Table 1. In terms of EER, the GMM-UBM male system performed the best on VAST condition and the x-vector system performed the best on the CMN2 condition. In terms of minDCF, the x-vector system performed the best overall. The systems are fused at the score level. The fusion of all the individual systems provided the best results in terms of minDCF. However, there was 0.8% absolute degradation in EER from the GMM-UBM system on the VAST condition after fusion. The significance of this degradation is unclear as the size of the test data is severely limited.

5.2. Performance and Processing Requirements

The infrastructure used to train x-vector and LF-MMI acoustic models contains 16 GPU GeForce GTX 1080 Ti with 11 GB memory per GPU. The i-vector extraction for enrollment and probing is done on CPU, Intel(R) Core(TM) i7-5930K CPU @ 3.50GHz with a memory of 32 GB. The average execution time for a single threaded when computed on speech (i.e. without silence) was approximately 1.2s to compute the i-vector for the UBM-GMM system. The execution time for the TDNN i-vector system is also approximately 1.2 s. It is to be noted that for the TDNN i-vector system the posteriors were extracted on the GPU while the i-vector was estimated on the CPU. For both systems, the maximum memory usage for modelling is related to the size of the i-vector extractor, which is 81 MB for the TDNN i-vector system and 501 MB for the UBM-GMM system. The time taken for 10'000 trials in both cases is approximately 0.25 s. The extraction of x-vector for enrollment and probing is done on CPU, Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz, with a total memory of 32 GB. The execution time of x-vector extraction process in a single thread when computed only on detected speech is of 21.43 times faster than real time (FRT). For the whole recordings including silence, it would be 33.4 FRT using 2 GB of memory. x-vector averaging time for enrollment and scoring time is negligible with respect to the x-vector extraction time.

Table 1. Results on the development set of NIST SRE 2018 dataset for all systems presented as provided by the NIST toolkit. EER: Equal Error Rate, minDCF: minimum Decision Cost Function.

System	VAST		CMN2	
	EER (%)	minDCF	EER (%)	minDCF
GMM-UBM male	8.2	0.704	11.7	0.673
GMM-UBM female	11.1	0.67	15.1	0.83
DNN I-vector	12.8	0.815	13.5	0.737
x-vector	11.11	0.597	8.81	0.583
Fusion	9.05	0.630	7.7	0.537

6. ACKNOWLEDGEMENT

This work was partially supported by (1) CTI project ("Shaped") registered under the Commission for Technology and Innovation (Switzerland) and by (2) the European H2020 project TeSLA.

7. REFERENCES

[1] Najim Dehak, Patrick Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," 2011, vol. 19(4), pp. 788–798, IEEE Tran. on Audio, Speech and Language Processing.

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

[3] 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.

[4] 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.

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

[6] Srikanth Madikeri, Subhadeep Dey, Marc Ferras, Petr Motlicek, and Ivan Himawan, "Idiap submission to the nist sre 2016 speaker recognition evaluation," Tech. Rep., Idiap, 2016.

[7] Srikanth Madikeri, Subhadeep Dey, and Petr Motlicek, "Analysis of language dependent front-end for speaker recognition," *Proc. Interspeech 2018*, pp. 1101–1105, 2018.

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

[9] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[10] Jason Pelecanos and Sridha Sridharan, "Feature warping for robust speaker verification," 2001, pp. 213–218, In Proc. of Speaker Odyssey.

[11] Srikanth Madikeri, Petr Motlicek, Marc Ferras, and Subhadeep Dey, "Analysis of posterior estimation approaches to i-vector extraction for speaker recognition," Tech. Rep. Idiap-RR-15-2018, Idiap, 10 2018.

[12] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," pp. 2751–2755, 09 2016.

[13] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," *Proceedings of Interspeech*, 2015.

[14] Daniel Povey et al., "The Kaldi speech recognition toolkit," in *Automatic Speech Recognition and Understanding*, 2011.

[15] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *Submitted to ICASSP*, 2018.

[16] 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.

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

[18] Douglas E Sturim and Douglas A Reynolds, "Speaker adaptive cohort selection for tnorm in text-independent speaker verification," in *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.* IEEE, 2005, vol. 1, pp. I–741.

[19] Pavel Matejka, Ondrej Novotnỳ, Oldrich Plchot, Lukáš Burget, and JH Cernockỳ, "Analysis of score normalization in multilingual speaker recognition," in *Proceedings of Interspeech*, 2017.