INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

A Speaker Recognition System for the SITW Challenge

Oleg Kudashev [1,2], *Sergey Novoselov* [1,2], *Konstantin Simonchik* [1,2], *Alexandr Kozlov* [2]

[1] ITMO University, St.Petersburg, Russia
[2] Speech Technology Center Ltd., St. Petersburg, Russia
{kudashev, novoselov, simonchik, kozlov-a}@speechpro.com

Abstract

This paper presents an ITMO university system submitted to the Speakers in the Wild (SITW) Speaker Recognition Challenge. During evaluation track of the SITW challenge we explored conventional universal background model (UBM) Gaussian mixture model (GMM) i-vector systems and recently developed DNN-posteriors based i-vector systems. The systems were investigated under the real-world media channel conditions represented in the challenge. This paper discusses practical issues of the robust i-vector systems training and performs investigation of denoising autoencoder (DAE) based back-end when applied to "in the wild" conditions. Our speaker diarization approach for "multi-speaker in the file" conditions is also briefly presented in the paper. Experiments performed on the evaluation dataset demonstrate that DNN- based i-vector systems are superior to the UBM-GMM based systems and applying DAE-based back-end helps to improve system performance.

Index Terms: SITW, i-vector, DNN, PLDA, DAE.

1. Introduction

The Speakers in the Wild (SITW) Speaker Recognition Challenge [1, 2] deals with the task of speaker detection in the unconstrained real-word conditions. The SITW Speaker Recognition Challenge provides database [1] with speech recorded in such conditions. These recordings contain samples of media channels with natural characteristics of the original audio samples such as different noise, reverb, compression and other artifacts. Such varying conditions are expected to be difficult for speaker recognition and the main goal of the challenge is to explore new ideas for solving major problems still faced by current speaker recognition technology and to apply them to the real-world data.

Besides "in the wild" recording conditions of the audio data there are several other important aspects of the challenge. The SITW evaluation had:

- Two tracks: evaluation and exploratory.
- Three enroll conditions: core, assist, assistclean.
- Two test conditions: core and multi.
- Development set: approx. 120 speakers

Detailed challenge description is presented in [2].

For many participants the small amount of the 'in-domain' media channel development data leads to the necessity of solving the domain mismatch problem in the challenge. The reason is that typically large datasets like NIST SRE datasets of microphone and telephone channels are used for the speaker recognition system training. The recording conditions of these datasets differ a lot from those for datasets of a media channel provided in the SITW challenge.

The application of the DNN-based i-vector extraction framework [3, 4, 5] for the speaker recognition task leads to significant performance improvements in comparison to conventional UBM-GMM-based systems in telephone channel conditions. However, application of DNN posteriors based systems in case of domain mismatch conditions (e.g. between microphone and telephone channels) comes with its own set of issues [4, 5]. These issues result in overfitting of the system to the specific training conditions. It leads to performance degradation of the system. The UBM-GMM-based approach can thus be more convenient in unconstrained conditions of media channels [4].

This work presents the development of different approaches based on UBM-GMM and DNN when applied to the challenge dataset. Significant attention is paid to practical issues of robust i-vector systems training. The influence of using artificially noised training data for minimization of the mismatch between train and evaluation conditions is studied. In addition to conventional PLDA, a novel back-end based on DAE-PLDA scheme [6, 7] is investigated.

In order to solve a speaker recognition task in "assist" and "assistclean" enrollment conditions we proposed an algorithm that applies a speaker diarization framework to extract speech segments of the target speaker based on a small amount of manually annotated material.

The final ITMO system for the evaluation track of the SITW challenge is a fusion of different subsystems with prior score stabilization with respect to test and enroll speech segments durations.

The paper is organized as follows. A detailed description of the ITMO speaker verification subsystems is given in Section 2. Section 3 describes the training dataset preparation. Section 4 presents our final experiments on the test dataset of the SITW Challenge. Section 5 concludes the paper.

2. System description

In this section we provide a description of all speaker recognition subsystems used in our work. We reviewed a number of existing speaker identification frameworks in order to determine efficient and promising approaches to speaker identification "in the wild" conditions.

2.1. UBM-GMM i-vector systems

The UBM/i-vector framework is a well-known framework in the speaker recognition field. During the SITW challenge we decided to explore two different UBM based i-vector extrac-

http://dx.doi.org/10.21437/Interspeech.2016-1197

tors. One of them was the freely available i-vector extractor proposed by Voice Biometry Standardization (VBS) initiative. The detailed description of the VBS extractor can be found in [8]. This system is denoted *UBM_{VBS} /i-vector*. The second system was implemented in-house and trained on NIST SRE datasets. The system is labeled *UBM_{ITMO}/i-vector*.

To train our $\mathrm{UBM}_{\mathrm{ITMO}}$ /i-vector extractor we used 1024 diagonal components GMM and 600-dimentional total variability space. Feature vectors consist of 13 Mel-Frequency Cepstral Coefficients (MFCC) calculated using 20 filter banks in the range of 300– 3400 Hz as well as their first- and second-order derivatives. Non-speech feature vectors were removed according to the energy-based Voice Activity Detection (VAD). Finally, the cepstral mean normalization was applied.

2.2. DNN –based i-vector system

Among the alternatives, state-of-the-art *DNN/i-vector* framework provides the best speaker recognition performance in "clean" speech conditions [4, 5, 14].

In the *DNN-based i-vector* framework the Deep Neural Network substitutes the UBM in calculation of Baum-Welch statistics followed by the total variability factor analysis. Alternatively, DNN can be used for bottleneck (BN) features extraction. Appending these BN features with MFCCs and using them in *UBM/i-vector* framework also provides impressive speaker recognition performance [5].

In our work for the SITW challenge we applied only DNN posterior based i-vector extraction procedure. For this purpose, a DNN was trained on the Switchboard corpus using the KALDI speech recognition toolkit [9]. Outputs of the DNN correspond to the set of 2700 speech triphone states as well as 20 non-speech states (noise, silence, laughing, etc.). Only 2700 speech-related outputs were used for the calculation of statistics. That prevented us from using any stand-alone VAD. The reader can refer to [7] for more DNN implementation details. 20 MFCC's (including log energy) were calculated using 23 filter banks in the range of 20– 3700 Hz with their first- and second-order derivatives. Mean and variance normalization was consequently applied. This system was named *DNN/i-vector*.

2.3. PLDA back-end

In the case of the simplified PLDA verification system we used the following model:

$$i_r(s) = \boldsymbol{m}_0 + \boldsymbol{V}y(s) + \varepsilon_{r,s}, \tag{1}$$

where $i_r(s)$ is an *f*-dimensional i-vector from set $\{i_1, \ldots, i_R\}$ obtained from *R* utterances belonging to the speaker ***s***, and ***y,*** $\varepsilon_{r,s} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ are hidden speaker factors and Gaussian noise, respectively, $\boldsymbol{V}$- is an eigenvoices matrix

Given a pair of i-vectors i_1 and i_2, assuming zero mean and skipping the scalar term, the commonly used PLDA verification score can be written as [10, 15]:

$$Score = i_1^T \boldsymbol{Q} i_1^T + i_2^T \boldsymbol{Q} i_2^T + 2 i_1^T \boldsymbol{P} i_2^T, \tag{2}$$

where square matrices ***P*** and ***Q*** can be expressed in terms of ***V*** and **Σ**

2.4. DAE-based back-end

Aside from the standard PLDA we studied the application of a denoising autoencoder (DAE) based back-end [6, 7] to SITW data "in the wild" conditions.

The DAE training starts from generative supervised training of the denoising RBM (Figure 1, left). This RBM has a binary hidden layer and a Gaussian visible layer, taking a concatenation of two real-valued vectors as an input. The first vector $\mathrm{i(s,h)}$ is an i-vector extracted from the h-th session of the s-th speaker, the second vector $\mathrm{i(s)}$ is the average over all sessions of this speaker. $\mathrm{i(s)}$ can be viewed as the maximum likelihood estimate in the following model of within-speaker variability: $\mathrm{i(s,h)} \sim \mathcal{N}(\mathrm{i(s)}, \Sigma_\mathrm{W})$, where $\mathcal{N}(\cdot)$ is the Gaussian distribution with mean $\mathrm{i(s)}$ and covariance Σ_W.

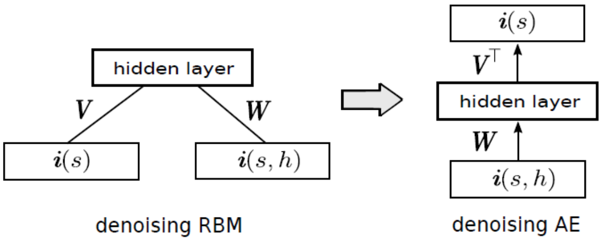

Figure 1: *Learning denoising transform. $i(s,h)$ is the i-vector representing h-th session of s-th speaker. $i(s)$ is the mean i-vector for speaker s. RBM parametrs are used to initialize denoising neural network.*

Then we "unfold" the trained RBM to form the neural network which we refer to as denoising autoencoder (DAE) [7] (Figure 1, right). DAE is discriminatively trained (fine-tuned) to minimize within-speaker variability, defined in the following way:

$$\sum_s \sum_h \left\| i(s) - f\big(i(s,h)\big) \right\|_2^2 \to \min \tag{3}$$

where $f(x) = \boldsymbol{V}^{\boldsymbol{T}} \sigma(\boldsymbol{W}x)$ – denoising transform, $\sigma(\cdot)$ – logistic function.

We used standard PLDA (see Section 2.4) to compute similarity measure for DAE projections. During our experiments [7] we found out that the best performance of the DAE system was obtained when using the set of PLDA parameters estimated on i-vectors passed through the RBM instead of DAE.

2.5. LDA-SVM back-end

In our work we also tried to use discriminative SVM method at the back-end of the speaker verification system. Similar to [10] SVM was applied to i-vectors after LDA projection. We used SVM with linear kernel and implemented *s*-normalization of the scores. The whole development set was used as impostors for the SVM.

2.6. Calibration and Fusion

It is well-known that there is a dependency between the value of the minDCF threshold of a verification system and the duration of speech segments that were used for i-vectors extraction. This is caused by a shift in target and impostor score distributions depending on the test and enroll speech segments duration. This effect can be compensated using Quality Measure Function (QMF):

$$\hat{s} = w_0 + w_1 s + Q(t_{test}, t_{enroll}, w_2 \dots), \tag{4}$$

where s and $\hat{s}$ are the raw and calibrated scores; $w_0, w_1, w_2 \dots, Q(.)$ - calibration parameters and a function, trained on the development dataset, t_{test} and t_{enroll} represent speech segments duration.

Assuming a Gaussian distribution of scores the means of target and impostor scores distributions can be represented as functions of test and enroll speech segments duration. Thus, the scores stabilization procedure can be performed using approximations of those dependencies. For simplification, let us suppose that the target and impostor scores distribution variances are the same and independent of the speech segments duration:

$$\sigma_{tar}^2 = \sigma_{imp}^2 = \sigma^2 \tag{5}$$

Then the score stabilization formula is:

$$\hat{s} = y_0 + y_1 s, \tag{6}$$

where

$$y_0 = -\frac{1}{2\sigma^2}(\mu_T^2 - \mu_I^2) \tag{7}$$

$$y_1 = \frac{1}{\sigma^2}(\mu_T - \mu_I) \tag{8}$$

where μ_T - target-scores mean and μ_I - impostor-scores mean. To compensate the scores shifting we used an approximation of μ_T and μ_I according to the formula:

$$\mu_T = f(\tau_1, \tau_2, \boldsymbol{C}^{tar}), \mu_I = f(\tau_1, \tau_2, \boldsymbol{C}^{imp}),$$

where $f(\cdot,\cdot,\boldsymbol{C})$ is a square symmetric polynomial function with parameters $\boldsymbol{C} = \{C_1, C_2 \dots C_6\}$; $\tau_1 = \sqrt{log(t_{test} + 1)}$ and $\tau_2 = \sqrt{log(t_{enroll} + 1)}$ where t_{test} and t_{enroll} are speech durations (in sec.). The function $f(\cdot,\cdot,\boldsymbol{C})$ takes the form:

$$f(\tau_1, \tau_2, \boldsymbol{C}) = C_1\tau_1^2\tau_2^2 + C_2(\tau_1^2\tau_2 + \tau_2^2\tau_1) + C_3(\tau_1^2 + \tau_2^2) + C_4\tau_1\tau_2 + C_5(\tau_1 + \tau_2) + C_6 \tag{9}$$

The parameters $\boldsymbol{C}^{tar}$ и $\boldsymbol{C}^{imp}$can be estimated on some development set using mean square error (MSE) minimization for the polynomial approximation.

After compensation of duration-dependent score shifting the BOSARIS fusion toolkit [12] was used for final subsystem calibration and fusion. DCF threshold $\log((1 - \mathrm{P_{tar}})/\mathrm{P_{tar}}) = 4.59$ was used for calibration, where $\mathrm{P_{tar}} = 0.01$.

2.7. Speaker diarization

Audio records used in “assist” and “assistclean” enroll conditions contain more than one speaker. The only available information is a hand-marked time interval on the audio records, which contains speech segments of the target speaker only. To find all speech segments of the target speaker we applied i-vector-based speaker diarization framework. After that, resulting speech segments were prepared as concatenation of all speech segments from the hand-marked time interval and speech segments belonging to the speaker that has the largest intersection with the hand-marked interval.

Speaker diarization framework based on i-vectors includes the following steps: 1) splitting the audio record into short speech segments (up to 1 sec.). 2) extracting i-vectors from these speech segments; 3) clustering i-vectors based on Variational Bayesian Analysis (VBA) and PLDA model, which was trained on short speech segments (up to 3 sec.). For further details the reader can refer to [11].

We used gender-independent UBM with 1024 diagonal Gaussians and total variability matrix of dimension 100 to extract i-vectors. Twenty MFCCs (without energy) were calculated using 27 filter banks in the range of 100–3700 Hz without derivatives and without normalization.

UBM, total variability matrix and PLDA model were trained using NIST’s 1998-2010 dataset without artificial data augmentation.

3. Training dataset preparation

As mentioned above, evaluation data provided in the SITW challenge greatly differs from commonly used training datasets like NISTs, Switchboard, Fisher and so on. Different distortions are present in the challenge’s dataset including all kinds of additive noise and reverberation. Additive noise usually includes babble noise, which causes major difficulties for speaker modeling. Reverberation time RT_{60} can be as long as 1 sec., and SNR values can be as low as 5 dB.

In this challenge, NIST’s 1998-2010 dataset was used to train $\mathrm{UBM_{ITMO}}$/i-vector and DNN/i-vector systems as well as PLDA and DAE back-ends for all systems. The prepared dataset is gender-balanced and consists of 8800 microphone and 20800 telephone sessions of 3000 speakers.

To reduce the mismatch between train and evaluation conditions a “noised” train data were generated using a MATLAB tool provided in the REVERB challenge [13]. In contrast to “clean” data, the “noised” data were obtained by distorting 50% of audio records. Additive babble noise and reverberation were added to match SITW conditions as close as possible.

4. Results

Tables 1 to 5 summarize the results of our evaluation of DNN/i-vector and UBM/i-vector systems under different training conditions and with different back-ends. Several performance measures were used to evaluate system performance during the SITW Challenge. In our investigations we focused on some of them: equal error rate (EER), minimum decision cost function (minDCF) with $C_{miss} = C_{fa} = 1$ and $P_{tar} = 0.01$ and corresponding primary actual DCF metric.

These results demonstrate that current state-of-the-art speaker identification system performance degrades when applied to “in-the-wild” datasets. The EER of such PLDA-based systems reaches 10%, while EER of the best DNN/i-vector based system is less than 2% on telephone dataset (NIST 2010 test, det 5 protocol). According to the Table 1, the DNN/i-vector system fine-tuned to the “clean” dataset conditions appears to be less robust to the unconstrained recording conditions compared to the conventional UBM/i-vector system.

It is possible to compensate the mismatch degree between the train and test datasets by using an artificially augmented training dataset. This leads to improved speaker detection results (see Table 2, 4, 5). Here the DNN-based system provides better performance than a UBM-based one.

Table 1. *The Evaluation results for the systems trained on "clean" data and with PLDA as a back-end*

Extractor name	EER ,[%]	min DCF	act DCF
UBM$_{VBS}$ /i-vector	11.18	0.774	0.780
UBM$_{ITMO}$ /i-vector	11.56	0.750	0.769
DNN/i-vector	10.83	0.803	0.816

Table 2. *The evaluation results for the systems trained on "noised" data and with PLDA as a back-end*

Extractor name	EER ,[%]	min DCF	act DCF
UBM$_{VBS}$ /i-vector	11.13	0.720	0.733
UBM$_{ITMO}$ /i-vector	11.50	0.728	0.732
DNN/i-vector	10.63	0.700	0.710

Tables 3 and 4 also demonstrate that employing a nonlinear DAE+PLDA back-end improves the performance of all systems under consideration. This improvement is especially noticeable for the DNN/i-vector system when tuned on "noised" train data. For this case a value of 0.678 at the actDCF point was obtained and the EER moved 18% down (relative).

Table 3. *The evaluation results for the systems trained on "clean" data and with DAE+PLDA as a back-end*

Extractor name	EER ,[%]	min DCF	act DCF
UBM$_{VBS}$ /i-vector	11.20	0.746	0.749
UBM$_{ITMO}$ /i-vector	11.19	0.738	0.739
DNN/i-vector	8.96	0.781	0.786

Table 4. *The evaluation results for the systems trained on "noised" train data and with DAE+PLDA was used as a back-end*

Extractor name	EER ,[%]	min DCF	act DCF
UBM$_{VBS}$ /i-vector	11.86	0.737	0.746
UBM$_{ITMO}$ /i-vector	12.25	0.737	0.738
DNN/i-vector	8.88	0.672	0.678

Table 5. *The evaluation results for the systems trained on "noised" data and with DAE+PLDA as a back-end with scores stabilization*

Extractor name	EER ,[%]	min DCF	act DCF
UBM$_{VBS}$ /i-vector	11.21	0.738	0.739
UBM$_{ITMO}$ /i-vector	11.07	0.735	0.730
DNN/i-vector	8.41	0.669	0.675

Table 5 demonstrates the effectiveness of applying scores stabilization suggested in section 2.6.

For the LDA-SVM based systems the results for different extractors were roughly the same, reaching about 0.75 in terms of minDCF. This value is comparable to the results in Table 1, due to low amount of speakers in the development set that were used as impostors for the SVM training.

For the SITW Challenge we decided to submit the several subsystems fusion results. Some of them are presented in the Table 6.

Table 6. *The evaluation results for different subsystem fusions. (EER /minDCF /actDCF)*

Subsystem names	Evaluation protocol		
	core-core	assist-core	assist-clean-core
UBM$_{ITMO}$/i-vector(DAE_PLDA) DNN /i-vector(DAE_PLDA)	0.081/ 0.645/ 0.647	0.063/ 0.543/ 0.555	0.062/ 0.486/ 0.505
UBM$_{ITMO}$/i-vector(DAE_PLDA) UBM$_{VBS}$ /i-vector(DAE_PLDA) DNN /i-vector(DAE_PLDA)	0.077/ 0.641/ **0.650**	0.064/ 0.532/ **0.538**	0.059/ 0.466/ **0.469**
UBM$_{ITMO}$ /i-vector(DAE_PLDA) UBM$_{VBS}$ /i-vector(DAE_PLDA) DNN /i-vector(DAE_PLDA) UBM$_{VBS}$ /i-vector(LDA_SVM)	0.078/ 0.645/ 0.691	0.075/ 0.534/ 0.540	0.079/ 0.456/ 0.471

As seen in the Table 6, minDCF values for various systems are close to each other. The last system is badly calibrated, possibly because too small development set was used for SVM training.

It should be noted that the system corresponding to the second line in the Table 6 was used for SITW challenge, and it took the 3rd place on the "core-core" conditions.

5. Conclusions

This paper presented an ITMO university speaker recognition system for the Speakers in the Wild (SITW) Speaker Recognition Challenge. UBM/i-vector and DNN/i-vector based systems were investigated. When trained on "clean" dataset the DNN-based system proved to be less robust than the UBM-based one in unconstrained record conditions. We demonstrated that artificial augmentation of training data can reduce speaker detection error in these conditions. DNN-based systems greatly benefit from this approach and demonstrates a substantial performance improvement. Application of the denoising autoencoder at the back-end level and scores stabilization allow to further improve speaker detection quality.

6. Acknowledgments

This work was partially financially supported by the Government of Russian Federation, Grant 074-U01.

7. References

[1] Mitchell McLaren, Luciana Ferrer, Diego Castan, Aaron Lawson, "The Speakers in the Wild (SITW) Speaker Recognition Database", Submitted to Interspeech 2016.

[2] Mitchell McLaren, Luciana Ferrer, Diego Castan, Aaron Lawson, “The 2016 Speakers in the Wild Speaker Recognition Evaluation”, Submitted to Interspeech 2016.

[3] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically aware deep neural network,” in Proc. 2014 IEEE Int. Conf. Acoust., Speech, Signal Process., pp. 1695–1699.

[4] Y. Lei, L. Ferrer, M. McLaren, and N. Scheffer, “A deep neural network speaker verification system targeting microphone speech,” in Proc. Interspeech, 2014.

[5] McLaren, Mitchell, Yun Lei, and Luciana Ferrer. "Advances in deep neural network approaches to speaker recognition." Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. IEEE, 2015.

[6] Sergey Novoselov, Timur Pekhovsky, Oleg Kudashev,Valentin Mendelev, and Alexey Prudnikov, “Non-linear PLDA for i-vector speaker verification,” in INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015, 2015, pp. 214–218.

[7] T. Pekhovsky, S. Novoselov, A. Sholohov, O. Kudashev, “On autoencoders in the i-vector space for speaker recognition”, in Proc. Odyssey 2016.

[8] Voice Biometry Standardization Initiative [Online]. Available: http://voicebiometry.org

[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] S. Novoselov, T. Pekhovsky, and K. Simonchik, “STC Speaker Recognition System for the NIST i-Vector Challenge” presented at Odyssey 2014: *The Speaker and Language Recognition Workshop* [Online]. Available: http://cs.uef.fi/ odyssey2014/program/pdfs/25.pdf.

[11] O. Kudashev, T. Pekhovsky, “Speaker Diarization System Based On Probability Linear Discriminant Analysis”, tech. rep., 2014. Available: https://www.researchgate.net/publication/283714986_SPEAKER_DIARIZATION_SYSTEM_BASED_ON_PROBABILITY_LINEAR_DISCRIMINANT_ANALYSIS

[12] Brümmer N., de Villiers E. “The bosaris toolkit: Theory, algorithms and code for surviving the new dcf” // arXiv preprint arXiv:1304.2865. – 2013.

[13] The REVERB challenge [Online]. Available: http://reverb2014.dereverberation.com/

[14] Daniel Garcia-Romero and Carol Y Espy-Wilson, “Analysis of i-vector length normalization in speaker recognition systems.,” in Interspeech, 2011, pp. 249–252.

[15] Patrick Kenny, “Bayesian speaker verification with heavy tailed priors,” in Odyssey 2010: The Speaker and Language RecognitionWorkshop, Brno, Czech Republic, June 28 - July 1, 2010, 2010, p. 14.