INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

The I2R's Submission To VOiCES Distance Speaker Recognition Challenge 2019

Hanwu Sun, Kah Kuan Teh, Ivan Kukanov, Huy Dat Tran

Institute for Infocomm Research, A*STAR, Singapore

hwsun@i2r.a-star.edu.sg, tehkk@i2r.a-star.edu.sg, hdtran@i2r.a-star.edu.sg, ivan_kukanov@i2r.a-star.edu.sg

Abstract

This paper is about the I2R's submission to the VOiCES from a distance speaker recognition challenge 2019. The submissions were based on the fusion of two x-vectors and two i-vectors subsystems. Main efforts have been focused on the frontend de-reverberation processing, PLDA backend design, score normalization and fusion studies in order to improve the system performance on single channel distant/far-field audio, under noisy conditions. We contribute to the fixed condition task under specific training and development data set. The experimental results showed that the de-reverberation approach can achieve 5% to 10% relative improvement on both EER and DCF for all subsystems and more than 10% improvement in the final fusion system on the Dev dataset and more than 15% relative improvement on the final evaluation dataset. Our final fusion system achieved about 2% EER rate and 0.240 *min*DCF on the Development Dataset.

Index Terms— Speaker recognition, benchmark evaluation, reverberation, system description, x-vector, i-vector, PLDA, LDA.

1. Introduction

This paper describes the systems developed by Institute for Infocomm Research (I2R) team participating in the VOiCES (Voices Obscured in Complex Environments Settings) from a Distance Challenge 2019 [1, 2, 3], which is designed to promote research in the area of speaker recognition on single channel distant/far-field audio, under noisy conditions. The main objectives of this challenge include benchmark state-of-the art technology and support the development of new ideas and technologies in the area of speaker recognition [1].

The distance audio mic recording and audio reverberation in the different size rooms are the major challenge in this evaluation. VOICES dataset were captured by choosing audio data files from the existing available corpora based on close-talking mic speech and then retransmitting the audio data through a high-quality loudspeaker in a number of different size rooms [1]. The subset of the relative small room #1 and room #2 collected datasets were used for the VOICES Dev dataset. The dataset collected from the larger rooms #3 and #4 are used in the VOICE evaluation dataset [1].

We submitted three systems to VOiCES challenge in Fixed Training Condition based on different fusion strategies. This paper presents the technical details of our team approach, including: an approach of frontend de-reverberation processing, adopted x-vector and i-vector systems, data augmentation, PLDA backend, score normalization, and fusion strategies. The paper is organized as follows. Section 2 descripts the development and training dataset usages. A de-reverberation strategy for the audio pre-processing [4, 5, 6] is introduced in Section 3. Section 4 provides LDA, PLDA backend design. Score normalized (s-norm) method is descripted in Section 5. Primary metrics for VOiCES performance measure and our system fusion strategies are presented in Section 6. The experimental Development and Evaluation results are presented and analysed in Section 7. Conclusions are given in Section 8.

2. Train and Development Set

I2R team contributed three submissions in the fixed condition for VOiCES SID challenge. We build up our system by using four separated Kaldi recipes [7]: SITW [8, 9] and Voxceleb [10] 16khz x-vector and i-vectors. The four set vectors are generated: two set x-vectors and two set i-vectors, naming SITW-xvector, SITW-ivector, Voxceleb-xvector and Voxceleb-ivector. The aim of using both x-vector and i-vector has twofold. Firstly, we can compare the performance differences between x-vector and i-vector systems on VOiCES dataset. Meanwhile, we also like to know any complement benefit to add the i-vector result into the final fusion system.

In addition, both the SITW and Voxceleb Kaldi recipes use the same Voxveleb 1 & 2 dataset to train models, but the selected speaker utterances and noises are not exactly same. We hope that two set x-vectors and two set i-vectors generated from Kaldi SITW and Voxceleb recipes may add some extra benefits for the final system fusion.

From development (Dev) dataset provided by the VOiCES challenge [1], we found that 93 speaker related utterances (total 7680) do not have the targeted training models or trials. We extract these speaker's utterances from the dataset and use them for the "in-domain" PLDA backend design. The remained trials are used as the Dev Subset. The summaries of the redesigned development dataset are shown in Table 1.

Table 1: *VOiCES redesigned Development Dataset.*

	No. Spks	No. utts	Total Trials	Target Trials	Imposter Trials
Dev Set	196	15904	4005888	20096	3985792
Dev Subset	103	8224	2039808	20096	2019712
For PLDA	93	7680	-	-	-

We found such design can minimize the differences between the Dev set and Dev subset DCF and EER scores, since only some imposter trials or utterances are removed from the original Dev set. Both datasets have the same number of the targeted trials (20096). Meanwhile, the Dev subset still has more than 2 million imposter trials.

http://dx.doi.org/10.21437/Interspeech.2019-1997

We used the following training data for the fixed condition evaluation: VoxCeleb 1 & 2 [10], MUSAN [11] (Noise and music dataset) and RIRS noise [12] for x-vector and i-vector model training. In addition, The Voxceleb 1 & 2 datasets are also used to train LDA model and out-of-domain PLDA backend. And 93 speaker related utterances extracted from VOiCES Dev dataset are used to design in-domain PLDA model backend. We also used the Voxceleb 1 dataset for system score normalization. The details usages of datasets are summarized in Table 2.

Table 2: *Data Usages for the Model Training.*

Usages	Dataset
x/i-vector models	Voxceleb 1&2 MUSAN (noise and music), RIRS.
LDA, out-of-domain PLDA	Voxceleb 1 & 2
in-domain PLDA	93 speakers utterances extracted from VOiCES Dev set
Score normalization	Voxceleb 1

3. Weighted Prediction Error De-Reverberation

Far-field speech often includes reverberation. A reverberation is created when a sound or signal is reflected causing a large number of reflections or dispersions (wave propagation) on the surface of objects in the space. The late reverberation is the main cause for decrease in the accuracy of speech recognition or speaker recognition. Reverberation is generally modeled as the convolution of a Room Impulse Response (RIR) with the original signal denoted by:

$$y[n] = h[n] * x[n] \qquad (1)$$

where $x[n]$ is the source signal, $y[n]$ is the signal received at the microphone at time n, and $h[n]$ represents the impulse of the channel from the desired source to the microphone.

We adopted the de-reverberation based on the Weighted Prediction Error (WPE) [4, 5, 6] as frontend processing. This method is based on robust blind deconvolution using long-term linear prediction, with the motivation of reducing the effects of the late reverberation. This method receives speech signal in the time domain follow by complex STFT to compute the coefficients of the finite impulse response (FIR) linear prediction filters with taps w iteratively. Finally, a de-reverberated time waveform is obtained by subtracting it from the observed signal denoted by:

$$\hat{y}[n] = y[n] - \sum_{k=0}^{N-1} \widehat{w}\,[k]y[n-k-1] \qquad (2)$$

where the $\widehat{w}$ taps of the filter are computed by minimizing the Euclidean norm of the prediction error by:

$$\widehat{w} = \min_{w} \sum_n |y[n] - \sum_{k=0}^{N-1} \widehat{w}\,[k]y[n-k-1]|^2 \qquad (3)$$

A STFT with a window size of 64ms and a shift of 16ms is used. For WPE, we used the following parameters: 10 filter taps, a delay of 3 frames, 5 iterations of WPE algorithm and no using clean speech Power Spectral Density (PSD) context.

This de-reverberation processing is only applied to the VOiCES Dev dataset and evaluation dataset, not other background dataset.

4. LDA, PLDA and In-domain PLDA

In our four subsystems, we choose Voxceleb 1 & 2 dataset to train the LDA model. Then we apply the LDA reduces x-vector and i-vector dimensions into 175.

For PLDA backend, Voxceleb 1 & 2 datasets are used to do the out-of-domain PLDA model.   The 93 speakers extracted from VOiCES's Dev set and the speakers from Voxceleb 1 & 2 datasets are combined to build up the in-domain PLDA. Since the number of the 93 speakers extracted from VOiCES Dev set is much less than Voxceleb 1 & 2, we have repeated these 93 speakers 8 times to increase the VOiCES dataset weighting in the mixed PLDA model.

5. Score Normalization

Symmetric score normalization is conducted for all the four subsystem. The symmetric normalization (s-norm) with adaptive cohort selection scheme followed by top score selection (top s-norm). Cohort sets were selected from VoxCeleb1, which x/i-vector related audio utterances are first merged into long utterances from all small utterances under the same sections (or folders). For the score $T_{k,j}$ under given the speaker k and test utterance j, S-norm score, $T'_{k,j}$, can be computed as:

$$T'_{k,j} = \frac{1}{2}\left(\frac{T_{k,j}+m_k(\cap_j)}{\emptyset_k(\cap_j)} + \frac{T_{k,j}+m_j(\cap_k)}{\emptyset_j(\cap_k)}\right) \qquad (4)$$

where, m_k and $\emptyset_k$ are the mean and standard deviation of speaker k against the cohort Voxceleb1 dataset $\cap_j$. And this cohort set is selected based on the test utterance j. Only top N scores are selected to compute the mean and standard deviation instead of whole cohort sets.

6. Performance Measures and Fusion

6.1. Primary Metric

For the VOiCES challenge, the primary measure metric [1] follows the metric defined in the NIST SRE 2010 [13], but with slight different parameters settings (P_{Tar} is 0.01 instead of 0.001 in NIST's core condition). The primary metric for the VOiCES challenge is computed by the following Detection Cost Function (DCF):

$$C_{DCF} = C_{miss} \times P_{miss} \times P_{Tar} + C_{fa} \times P_{fa}(1 - P_{Tar}) \qquad (5)$$

where the parameters C_{miss}, C_{fa} and P_{Tar} are setting as shown in Table 3.

Table 3: *Detection Cost Parameters for VOiCES Challenge.*

C_{miss}	C_{fa}	P_{Tar}
1.0	1.0	0.01

Besides the above defined DCF cost function, we also used the well-known Equal Error Rate (EER) to indicate system performance.

6.2. System Fusion

We have used two type fusions in our submission: the linear fusion and Blend based non-linear fusion.

6.2.1. *Linear Fusion*

We adopt the Bosaris toolkit [14] linear fusion method to do the score calibration and fusion. The DCF settings follow the parameters given in Table 3. We use the VOiCES Dev subset trials for the parameters tuning. The linear fusion score output S is computed by

$$S = \sum_{i=1}^{N} S_i \times W_i + \theta \quad (6)$$

where S_i is i^{th} subsystem scores, W_i is the i^{th} subsystem weighting factor by minimizing the DCF cost. And N is number of the subsystems and θ is the fusion score offset value.

6.2.2. *Blend Based Non-Linear Fusion*

Besides the linear fusion, we propose the "Blend" model training method to conduct non-linear score fusion, which is partly inspired by the deep super learner [15]. The fusion system is trained on the VOiCES Dev subset, dividing it in proportion of [70%, 15%, 15%] for training/test/validation datasets.

We split training part (70%) of sub development dataset on stratified 10-folds: every fold has training and test part. On every fold, we train six models: XGBoost, random forest and gradient boosting classifiers with maximum tree depth of 5 and 6. Each classifier has 100 estimators and the entropy criteria are used for training each classifier. After training six classifiers on each fold, we produce six sets of prediction scores from each classifier on test part of each fold. And the evaluation dataset is forwarded through each classifier trained on each fold and averaged across folds. Therefore, we get six sets of predictions for sub development dataset and evaluation dataset. This is the first layer of classification of the super ensemble. Further, we train the random forest classifier with the maximum tree depth of 4 and number of estimators of 50 on the attained predictions (six sets) from the first layer. Finally, we train a calibration on the output scores from the random forest. The isotonic regression [16] calibrations techniques are used

7. Experimental Results

In this Section, we provided the detailed experimental results analysis. Firstly, we compare the EER and DCF differences between VOiCES Dev set and our redesigned Dev subset in Section 7.1. Next, the effects of WPE de-reverberation on the four subsystems are analyzed in Section 7.2. Typical results by using out-of-domain PLDA, in-domain PLDA and s-norm are summarized in Section 7.3. Finally, in Section 7.4, we present the I2R submission performances on both development (Dev) dataset and evaluation (Eval) dataset.

7.1. Differences between Dev Set and Dev Subset

Based on the redesigned VOiCES dev subset, we compared the results on SITW-xvector subsystem. Table 4 showed the EER and *min*DCF values by using out-of-domian and in-domain PLDA for both VOiCES Dev set and the new designed Dev subset.

From the results, we observed that the differences between EER and *min*DCF between Dev set and Dev subset are not very significant under the same PLDA conditions, especially for *min*DCF values by using both out-of-domian and in-domain PLDA backend.

Table 4: *Comparison of Dev Set and Dev Subset.*

Sitw-xvector	Dev Set		Dev Subset	
PLDA	EER*(%)*	*min*DCF	EER*(%)*	*min*DCF
Out-of-domain	3.67	0.410	3.73	0.419
+In-domain	3.02	0.353	3.22	0.359

7.2. De-Reverberation (WPE) Effects

Table 5 and 6 showed the four x/i-vector subsystem EER and *min*DCF results under the same conditions before and after applying the WPE de-reverberation preprocessing on both Dev Subset and Eval set. The PLDA backend only used Voxceleb 1 & 2 dataset and the s-norm is not applied. After applying WPE, we observed that both EER and *min*DCF are consistently improved over all the four subsystems, around 5% to 10% relative improvement for the Dev set and 10% to 15% relative improvement on Eval set. The WPE de-reverberation is more effective on the Eval set than Dev set. It may also indicate that the evaluation dataset has stronger reverberation than the development dataset.

Table 5: *Performances of WPE on Dev Subsystems.*

VOiCES DEV SUB SET	Without WPE		With WPE	
	EER*(%)*	*min*DCF	EER(%)	*min*DCF
SITW-ivector	7.29	0.621	6.88	0.595
SITW-xvector	3.99	0.433	3.67	0.410
Voxceleb-ivector	7.38	0.639	6.84	0.627
Voxceleb-xvector	4.38	0.406	4.04	0.388

Table 6: *Performances of WPE on Eval Subsystems.*

VOiCES Eval Set	Without WPE		With WPE	
	EER*(%)*	*min*DCF	EER(%)	*min*DCF
SITW-ivector	14.99	0.916	13.20	0.813
SITW-xvector	8.60	0.679	7.85	0.615
Voxceleb-ivector	15.08	0.918	13.12	0.818
Voxceleb-xvector	8.56	0.718	7.57	0.626

In addition, from Tables 5 and 6, we also observed that DNN embedding [17, 18] x-vector systems have a significant better performances (about relative 50%) over i-vector systems on both Dev and Eval sets. These results are also consistent with the results report in [19].

7.3. Results of PLDA and S-Norm

Based on SITW-xvector and Voxceleb-xvector subsystems, we have summarized the results in Table 7 by applying out-of-domian, in-domain PLDA backends, as well as s-norm. Where the out-of-domian PLDA model uses Voxceleb 1 & 2 and in-domain PLDA model adds the extra 93 speakers utterances extracted from VOiCES Dev set. S-norm uses the cohort sets from Voxceleb 1.

From Table 7, the in-domain PLDA model has significantly improved EER by about relative 20% and DCF by about 15%. S-norm further boost both EER and DCF for relative 5% to 10%.

Table 7: *System Performances Using Out-of-Domain/In-Domain PLDA and S-norm on VOiCES Dev Subset.*

VOiCES DEV Subset	SITW-xvector		Voxceleb-xvector	
	EER*(%)*	*min*DCF	EER*(%)*	*min*DCF
Out-domain PLDA	3.74	0.4347	4.11	0.4056
+in-domain PLDA	2.86	0.3692	3.36	0.3379
+S-norm	2.63	0.3452	3.12	0.3186

In order to demonstrate the complementary effects for our four subsystems, Figure 1 shows the Eval set and Dev subset (EER) performance progresses starting from single SITW-ivector subsystem by step-by-step adding another 3 vector subsystem scores, as well as applying in-domain PLDA and the s-norm. The linear fusion Bosaris toolkit [14] is used here.

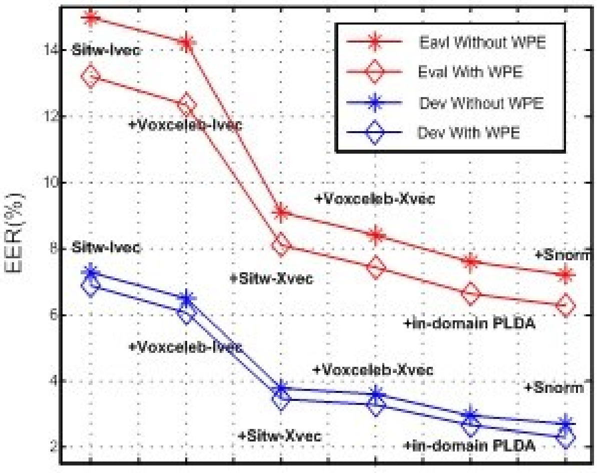

Figure 1: *Dev and Eval System EER Progresses By adding Subsystems, and Applying In-domain PLDA, S-norm.*

From Figure 1, WPE de-reverberation processing can consistently improve the combined system performances at all the stages for both Dev and Eval sets. Meanwhile, we also observed that four set subsystems have noticeably complementary effects in each fusion stage. Applying in-domain PLDA backend can significantly improve the fusion result. S-norm for the individual system score has also shown its effect in the final fusion stage.

7.4. I2R Final Dev Sets and Submission Eval Sets Results

The I2R final submission consists of three fixed condition fusion systems. System01 is based on linear fusion by using Bosaris toolkit [14]. The four x/i vectors subsystem scores are generated by adding in-domain PLDA backend and applying s-norm. Both Dev and Eval sets are pre-processed by WPE de-reverberation. System02 is based on Blend non-linear fusion. System03 is simple linear fusion again by using System01 and System02 outputs. Table 8 and Table 9 shows three submission system results on the Dev and Eval sets, respectively.

From Table 8, the Blend based non-linear fusion achieved significant better performances on both EER and DCF than the linear fusion. Fusion of System01 and System02 can further improve performances on the Dev set and its EER can reach as low as 1.53%. For Eval set, the best results is from the linear fusion System01 and has worst performance in System03, which indicated the further linear fusion of System01 and System02 cause the overfitting of the System03.

We observed that fusion Systems01 on Dev set, achieved 2.42% EER, 0.2433 *min*DCF and 0.2447 *act*DCF. But, Eval set only achieved 6.28% EER, 0.4975 *min*DCF and 0.4992 *act*DCF. One of reasons is that the evaluation dataset are more noisy and short utterances than development dataset. It was recorded in the larger rooms and more serious reverberation was introduced. We also observed that the SNRs of the evaluation dataset are much lower than the development dataset.

Table 8: *Performances of Fusion System on Development Dataset.*

Dev Set	EER*(%)*	*min*DCF	*act*DCF
System01	2.42	0.2433	0.2447
System02	1.76	0.2264	0.2265
System03	1.53	0.2230	0.2234

Table 9: *Performances of Fusion System on Evaluation Dataset.*

Eval Set	EER*(%)*	*min*DCF	*act*DCF
System01	6.28	0.4975	0.4992
System02	6.57	0.4932	0.4995
System03	7.69	0.4939	0.5024

8. Conclusions

The I2R team contributed three submissions for VOiCES 2019 challenge on the SID fixed condition evaluation. Based on development dataset provided, we generated four sets x/i vectors subsystems. Linear fusion and Blend based non-liean fusions were conducted. The experimental demonstrated that DNN embedding SITW/Voxceleb xvectors subsystems have about relative 50% better performances than the i-vector subsystems. The de-reverberation (WPE) method can also consistently improve all the subsystem EER and DCT for around 5 to 10% relative improvement for the Dev set and more than 10% to 15% on the Eval set. Our final fusion achieved 2% EER on Dev set. We also achieved very close actual DCF scores with *min*DCF on the Eval set.

Although, Eval set results are much worse than the Dev set, we still observed that the performance progress trend on the Dev set is very consistent with Eval set.

9. References

[1] M. K. Nandwana, J. Van Hout, M. McLaren, C. Richey, A. Lawson, M Barrios, "The VOiCES from a Distance Challenge 2019 Evaluation Plan," arXiv:1902.10828 [eess.AS], March 2019.

[2] C. Richey, M. Barrios, Z. Armstrong, C. Bartels, H. Franco, M. Graciarena, A. Lawson, M. K. Nandwana, A. Stauffer, J. van Hout, P. Gamble, J. Hetherly, C. Stephenson, K. Ni, "Voices Obscured in Complex Environmental Settings (VOICES) corpus," in ISCA *INTERSPEECH-2018*, pp. 1566-1570, 2018.

[3] M. K. Nandwana, J. van Hout, M. McLaren, A. Stauffer, C. Richey, A. Lawson, and M. Graciarena, "Robust speaker recognition from distant speech under real reverberant environments using speaker embeddings," in *INTERSPEECH-2018*, 2018, pp. 1106–1110.

[4] Weighted Prediction Error for speech dereverberation, Available: https://github.com/fgnt/nara_wpe.

[5] T. Yoshioka, T. Nakatani, M. Miyoshi and H. Okuno, "Blind separation and dereverberation of speech mixtures by joint optimization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 69-84, 2011.

[6] T. Yoshika, X. Chen and M. Gales, "Impact of single-microphone dereverberation on dnn-based meeting transcription systems," *in Proc. of ICASSP*, Florence, Italy, 2014.

[7] Kaldi ASR tools kits, Available: https://github.com/kaldi-asr/kaldi.git.

[8] SITW Dataset, Available: http://www.speech.sri.com/projects/sitw/.

[9] M. McLaren, L. Ferrer, D. Cast´an, and A. Lawson, "The speakers in the wild (SITW) speaker recognition database," *in Interspeech-2016*, San Francisco, USA, 2016, pp. 818–822.

[10] Voxceleb data, http://www.robots.ox.ac.uk/ vgg/data/voxceleb/

[11] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *CoR*R, vol. abs/1510.08484, 2015.

[12] RIRS noise, http://www.openslr.org/resources/28/rirs_noises.zip.

[13] "The NIST Year 2010 Speaker Recognition Evaluation Plan" Available:https://www.nist.gov/sites/default/files/documents/itl/iad/mig/NIST_SRE10_evalplan-r6.pdf.

[14] N. Brummer and E. de Villiers, "The Bosaris toolkit," Available: https://sites.google.com/site/bosaristoolkit/.

[15] S. Young, T. Abdou, and A. Bener, "Deep super learner: A deep ensemble for classification problems," *CoRR*, 2018.

[16] M. J. Best and N. Chakravarti, "Active set algorithms for isotonic regression: A unifying framework," *Mathematical Programming*, vol. 47, no. 1-3, pp. 425–439, may 1990.

[17] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text independent speaker verification," *in INTERSPEECH-2017*, 2017, pp. 999–1003.

[18] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2018*, April 2018, pp. 5329–5333.

[19] M. K. Nandwana, J. van Hout, M. McLaren, A. Stauffer, C. Richey, A. Lawson, and M. Graciarena, "Robust speaker recognition from distant speech under real reverberant environments using speaker embeddings," *in INTERSPEECH-2018*, 2018, pp. 1106–1110.