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

The NEC-TT 2018 Speaker Verification System

Kong Aik Lee[1], *Hitoshi Yamamoto*[1], *Koji Okabe*[1], *Qiongqiong Wang*[1], *Ling Guo*[1], *Takafumi Koshinaka*[1], *Jiacen Zhang*[2], *Koichi Shinoda*[2]

[1]NEC Corporation, Japan
[2]Tokyo Institute of Technology, Japan

{k-lee@ax,h-yamamoto@bc,k-okabe@bx,q-wang@ah,l-guo@bp,koshinak@ap}.jp.nec.com
{jiacen@ks.cs,shinoda@c}.titech.ac.jp

Abstract

This paper describes the NEC-TT speaker verification system for the 2018 NIST *speaker recognition evaluation* (SRE'18). We present the details of data partitioning, x-vector speaker embedding, data augmentation, speaker diarization, and domain adaptation techniques used in NEC-TT SRE'18 speaker verification system. For the speaker embedding front-end, we found that the amount and diversity of training data are essential to improve the robustness of the x-vector extractor. This was achieved with data augmentation and mixed-bandwidth training in our submission. For the multi-speaker test scenario, we show that x-vector based speaker diarization is promising and holds potential for future research. For the scoring back-end, we used two variants of probabilistic linear discriminant analysis (PLDA), namely, the Gaussian PLDA and heavy-tailed PLDA. We show that correlation alignment (CORAL) and CORAL+ unsupervised PLDA adaptation are effective to deal with domain mismatch.

Index Terms: speaker recognition, benchmark evaluation

1. Introduction

Speaker recognition refers to the task of determining the identity of a speaker from that person voice [1, 2]. It has been shown useful, for example, in logical and physical access control [3], forensics [4], and diarization [5]. The series of speaker recognition evaluation (SRE) organized by the National Institute of Standards and Technology (NIST) has served as a major driving force advancing speaker recognition technology in the past decades. This paper describes the NEC-TT submission to the 2018 edition of NIST SRE (SRE'18). We focus on new technical advances and our solutions to major challenges in SRE'18.

Since its inception in 1996, NIST SREs have been focusing on narrowband speech transmitted via landline and mobile networks, which collectively constitute part of the *public switched telephone network* (PSTN). SRE'18 marks a significant difference from previous SREs with the inclusion of *voice over internet protocol* (VOIP) narrowband telephone speech and wideband audio-from-video (AfV). Both PSTN and VOIP telephone recordings are in Tunisian Arabic, while AfV recordings are in English. Essentially, SRE'18 evaluation set consists of two partitions: (i) narrowband telephone speech in Tunisian Arabic drawn from the *call-my-net 2* (CMN2) corpus, the design of which follows the CMN corpus [6], and (ii) wideband speech in English drawn from the *video annotation for speech technology* (VAST) corpus [7]. These partitions exhibit different set of challenges. For the narrowband CMN2 partition, the major challenge is the domain mismatch with respect to the *training* set which comprises mainly English utterances recorded over the PSTN (language and channel). For the wideband VAST partition, the major challenge is the multi-speaker test scenario, where the test segments consist not only the putative target speaker but speech from other impostors as well. Another challenge that we aimed to tackle is mixed-bandwidth training, where the same front-end could be used for both narrowband and wideband partitions.

Table 1: *Train and development dataset used for fixed training condition for CMN2 and VAST partitions [9].*

Partition	Corpora
CMN2-Train	SRE'04, 05, 06, 08, 10, 12 Switchboard-2 Phase I & II & III Switchboard Cellular Part 1 & 2 Fisher 1 & 2
VAST-Train	VoxCeleb1, VoxCeleb2
CMN2-Dev	SRE'18-Dev, SRE'16-Eval SRE'18-CMN2-Unlabeled
VAST-Dev	SRE'18-Dev, SITW-Eval

We introduced a number of new components in the NEC-TT 2018 speaker verification system [8] in order to deal with various challenges in SRE'18. Based on their order in the pipeline, these include multi-head attention model, mixed-bandwidth training, speaker augmentation, PLDA adaptation, diarization for multi-speaker test segments, and *top adaptive symmetric score normalization* (Top ASNORM). In this paper, we highlight those components that are new and have contributed to good performance on SRE'18 results. We also emphasize those techniques that have attracted much attention during the SRE'18 Workshop.

2. Train and Development sets

Parameter training and optimization of the component classifiers and the fusion device were carried out using the datasets as shown in Table 1 for the *fixed training condition* [9]. Most part of the train set was provided by NIST and LDC, and had been used in previous SREs. This encompasses Fisher, Switchboard, SRE' 04, 05, 06, 08, 10, 12. In the current case of SRE'18, this subset corresponds to out-of-domain data with respect to the CMN2 partition of SRE'18, i.e., 8 kHz conversational telephone speech (CTS) over PSTN and VOIP. Domain adaptation was performed using the unlabelled subset of SRE'18-Dev set. We also set aside the SRE'16-Eval as an additional development set to investigate various aspects of domain adaptation.

For VAST partition (i.e., wideband audio-from-video, or AfV), train data was drawn from VoxCeleb 1 and 2 corpora [10], while SITW-Eval [11] was used as the development set. In particular, the core-multi subtask of SITW-Eval was used as a VAST development set in addition to SRE'18-Dev. Notice that

http://dx.doi.org/10.21437/Interspeech.2019-1517

both VoxCeleb and SITW, though not provided by NIST and LDC, are allowed for Fixed Training Condition.

3. Advances in Front-end and Back-end Design

The vast majority of submissions to SRE'18 used x-vector embedding in one form or another. The high adoption rate of x-vector [12] signifies a major paradigm shift from i-vector representation [13] which has been used widely in previous SRE'12 and 16. In the following, we present our work on attention model and data augmentation at the front-end, and domain adaptation at the back-end.

3.1. Multi-head attention and data augmentation

Table 2 shows the configuration of the x-vector extractor used for NEC-TT SRE'18 submission. In particular, it consists of a 5-layer time-delay neural network (TDNN) [14], a pooling layer, followed by two fully-connected layers. The overall structure of the x-vector extractor is pretty much the same as that in the Kaldi recipe [1]. We describe below two new components added to the NEC-TT x-vector extractor, which we found beneficial to its performance: (i) two-head attention model, and (ii) audio and speaker augmentation.

Two-head attentive pooling. An x-vector extractor consists of three functional blocks – a frame-level feature extractor (`frame`) implemented with a TDNN, a statistical pooling layer (`pool`), followed by utterance classification (`utt`). The role of the pooling layer is to compute the average and standard deviation from the frame-level feature vectors produced by the TDNN. Instead of using an equal-weight averaging, we used a multi-head attention pooling mechanism (`att`). The use of attentive pooling for x-vector extraction was first reported in [15]. An attention model is a simple feed forward neural network attached side-by-side to the pooling layer. In [16], multi-head attention mechanism was shown effective for machine translation task.

Let K be the number of *heads*. We divide the frame-level feature vectors $\boldsymbol{h}(t)$ into K sub-vectors $\{\boldsymbol{h}_1(t), \ldots, \boldsymbol{h}_K(t)\}$. The attention weights $\alpha_k(t)$ for the k-th sub-stream $\boldsymbol{h}_k(t)$, for $t = 1, 2, \ldots, T$ are calculated with an attention model, one for each $k \in 1, \ldots, K$. The weighted mean $\boldsymbol{\mu}_k$ and standard deviation $\boldsymbol{\sigma}_k$ are calculated from each sub-stream, as follows:

$$\boldsymbol{\mu}_k = \sum_t^T \alpha_k(t) \boldsymbol{h}_k(t) \tag{1}$$

$$\boldsymbol{\sigma}_k^2 = \sum_t^T \alpha_k(t) \boldsymbol{h}_k(t) \odot \boldsymbol{h}_k(t) - \boldsymbol{\mu}_k \odot \boldsymbol{\mu}_k \tag{2}$$

Finally, the statistics derived from the sub-vectors are concatenated and passed to the subsequent fully connected layers. Table 2 shows the whole network structure with two-head attention used in our system. The frame level feature vectors at the output of the TDNN are divided into two steams of 750 dimensions each.The attention models are two small neural network with one hidden-layer with 64 hidden units. The entire network (together with the $K = 2$ attention models) are trained to minimize a multi-class cross-entropy loss, where the number N of target speakers is in the order of 10 thousands in our system.

The importance of data augmentation. The success of x-vector relies on the amount and diversity of the training data.

[1] https://github.com/kaldi-asr/kaldi/tree/master/egs/sre16/v2

Table 2: *Network structure of x-vector extractor. The notation T indicates the number of D-dimensional feature vectors in an utterance, t is the time index, and N is the number of speakers.*

Layer	Layer context	Input × Output (× Head)
`frame1`	$[t-2, t, t+2]$	$5D \times 512$
`frame2`	$\{t-2, t, t+2\}$	1536×512
`frame3`	$\{t-3, t, t+3\}$	1536×512
`frame4`	$\{t\}$	512×512
`frame5`	$\{t\}$	512×1500
`att1`	$\{t\}$	$750 \times 64\ (\times 2)$
`att2`	$\{t\}$	$64 \times 1\ (\times 2)$
`pool`	$[0, T)$	$750\ (\times T) \times 1500\ (\times 2)$
`utt6`	$\{0\}$	3000×512
`utt7`	$\{0\}$	512×512
`softmax`	$\{0\}$	$512 \times N$

An inexpensive way to multiply the amount of training data, and improve robustness, is by adding noise (e.g., babble, music), channel noise (codec) and imposing convolutive variation (e.g., room reverberation) to the original audio recordings. This is referred to as *audio* augmentation in speech processing. We applied three types of augmentation to cover acoustic variability in the test.

- **Type I** [12] consists of (i) adding noise, music, and mixed speech (babble) drawn from the MUSAN database [17], and (ii) adding reverberation by using simulated room impulse responses (RIR) [18].
- **Type II** [15] consists of (i) adding noise samples drawn from the PRISM corpus [19] at 8, 15, or 20dB SNR, (ii) adding reverberation using real RIR drawn from the REVERB challenge database [20], and (iii) encoding segments with an AMR codec at 6.7 or 4.75 kbps.
- **Type III** consists of speaker augmentation [21] with the aim to create additional target speakers by changing the audio speed with a factor from 0.9 to 1.1. Audio speed perturbation produces similar effects as those from vocal tract length perturbation in the spectral domain [22].

Comparing Types I and II, the major difference is *simulated* RIR versus *real* RIR. Type III is a new element we introduced to SRE'18.

3.2. Mixed-band training

Different from previous SREs, narrowband and wideband speech are available in SRE'18 train set (see Table 1). The narrowband train set consists of SREs 04–06, 08, 10, 12, Switchboard and Fisher. The wideband train set consists of VoxCeleb. We explore the combined use of narrowband and wideband train set with the belief that increased amount and diversity of training data is beneficial in training the x-vector extractor. Two obvious options are

(i) **Narrowband training**, where wideband VAST-Train set is down-sampled to 8 kHz and combined with the narrowband CMN2-Train set, and

(ii) **Mixed-band training**, where narrowband CMN2-Train set is up-sampled to 16 kHz and combined with wideband VAST-Train set.

The first option is straightforward and was used by most participants. We describe below a bandwidth extension (BWE) technique which enables mixed-band training. Consider the

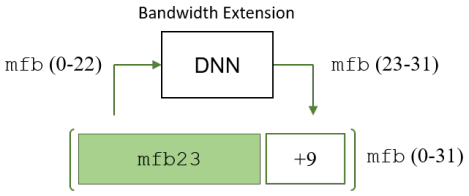

Figure 1: *Bandwith extension (BWE) with DNN for mixed-band training.*

use of log mel-filterbank (MFB) as acoustic features, a wideband spectrum requires 32 mel filters to cover the entire bandwidth. For narrowband speech, 9 of such filters at the higher frequency have zero input (and therefore zero output). As illustrated in Fig. 1, the goal of BWE is to infer the log MFB values `mfb` $(23-31)$ pertaining to the upper frequency band from the lower frequency `mfb` $(0-22)$. This is achieved with a feed-forward DNN, consisting of 5 hidden layers, with an input size of 23×5 (i.e., a context of $\pm$ 2), and output size of 9. The DNN was trained on wideband `VAST Train` set with a mean-square-error (MSE) loss.

3.3. PLDA and domain adaptation

Gaussian PLDA (G-PLDA) [23, 24] and heavy-tailed PLDA (HT-PLDA) [25, 26] were used as the scoring back-end in our system. Let ϕ be the speaker embeddings (i.e., x-vector, or i-vector) and $z \sim \mathcal{N}(z|0, I)$ be a latent speaker variable. In G-PLDA, we assume that ϕ is generated from a linear Gaussian model, as follows

$$p(\phi|\boldsymbol{z}) = \mathcal{N}(\phi|\boldsymbol{F}\boldsymbol{z}, \Phi_{\text{w}}) \tag{3}$$

where Φ_{w} is the within-speaker covariance matrix, and $\boldsymbol{F}$ is the speaker subspace which determines the between-speaker covariance $\Phi_{\text{b}} = \boldsymbol{F}\boldsymbol{F}^{\text{T}}$. Similarly, we have the following for heavy-tailed PLDA

$$p(\phi|\boldsymbol{z}, \lambda) = \mathcal{N}(\phi|\boldsymbol{F}\boldsymbol{z}, \lambda^{-1}\Phi_{\text{w}}) \tag{4}$$

where λ is a random scaling factor to the precision matrix (i.e., inverse of covariance matrix) that gives rise to the heavy-tailed nature of the residual covariance. See [26] for details.

For both variants of PLDA, unsupervised domain adaptation was applied using unlabelled dataset as listed in Table 1. We used three types of domain adaptation: embedding-level *correlation alignment* (CORAL) [27, 28], model-level CORAL+ [29], and Kaldi's unsupervised PLDA adaptation [30]. The central idea of domain adaptation is to estimate the between and within speaker covariance matrices, Φ_{b} and Φ_{w}, that are suitable for the in-domain evaluation set. We refer interested readers to [27, 28, 29] and references therein for more details.

4. Performance Evaluation and Results

4.1. Performance metric

The official performance metric defined for SRE'18 is the average cost computed for the CMN2 and VAST partitions:

$$C_{\text{primary}} = 0.25 \left(C_{\text{norm}}(\mathcal{T}_{\text{CMN2}}, P_{\text{tar}}^{(1)}) + C_{\text{norm}}(\mathcal{T}_{\text{CMN2}}, P_{\text{tar}}^{(2)}) \right) + 0.5\, C_{\text{norm}}(\mathcal{T}_{\text{VAST}}, P_{\text{tar}}^{(3)}) \tag{5}$$

Table 3: *Performance of the primary and single-best system evaluated on SRE'18 Eval set.*

CMN2	EER (%)	Min $C_{primary}$	Act $C_{primary}$
Primary	5.02	0.355	0.356
Single-best	6.05	0.429	0.430
VAST	EER (%)	Min $C_{primary}$	Act $C_{primary}$
Primary	12.70	0.417	0.441
Single-best	12.89	0.440	0.480

Table 4: *Configurations of x-vector extractor for CMN2, where `front1-5` are narrowband while `front6` is mixed-band.*

CMN2	Train set				Augmentation		
	SRE	SWB	FSH	Vox	(I)	(II)	(III)
`front1`	✓	✓			✓	✓	
`front2`	✓	✓	✓		✓	✓	
`front3`	✓	✓			✓	✓	✓
`front4`	✓	✓	✓		✓		✓
`front5`	✓	✓		✓	✓		✓
`front6`	✓	✓		✓	✓		✓

Table 5: *Configurations of x-vector extractor for VAST.*

VAST	Train sets				Augmentation		
	SRE	SWB	FSH	Vox	(I)	(II)	(III)
`front1`				✓	✓	✓	✓
`front2`	✓	✓		✓	✓		
`front3`	✓	✓		✓	✓		✓

where

$$C_{\text{norm}} = P_{\text{miss}} + \left(\frac{C_{\text{miss}}}{C_{\text{fa}}} \cdot \frac{1 - P_{\text{tar}}}{P_{\text{tar}}} \right) P_{\text{fa}}$$

The application parameters C_{miss} and C_{fa} are set to 1, while the probabilities of target are set to $P_{\text{tar}}^{(1)} = 0.01$, $P_{\text{tar}}^{(2)} = 0.005$, and $P_{\text{tar}}^{(3)} = 0.05$. The first term on the right-hand-side of (5) comprises the detection cost evaluated for the CMN2 trials, $\mathcal{T}_{\text{CMN2}}$, at two different thresholds determined by $P_{\text{tar}}^{(1)}$ and $P_{\text{tar}}^{(2)}$. The VAST trials, $\mathcal{T}_{\text{VAST}}$, accounts for the other half of the primary cost, C_{primary}. One could notice from the performance metric that there is no cross-domain CMN2/VAST trials, and more importantly both partitions could be processed with separate systems optimized for individual sub-tasks. This was the approach taken in most submissions to SRE'18. Table 3 shows the performance of NEC-TT's *primary* and *single best* submissions. Compared to CMN2, VAST is more difficult where the *equal error rates* (EER) are almost double. In the following, we analyze the performance on CMN2 and VAST partitions separately.

4.2. Narrowband CMN2

We trained 6 x-vector front-ends with different configurations as shown in Table 4. These front-ends were trained progressively by adding more data, with additional audio augmentation and attentive pooling. The performance of these x-vector front-ends used in conjunction with PLDA and heavy-tailed PLDA back-ends are shown in Table 6 for seven different configurations with and without domain adaptation at either feature or model levels. Among the 42 combinations, HT-PLDA with CORAL+ gives the best performance in terms of minimum C_{primary}. NEC-TT primary submission on CMN2 partition was

Table 6: *The comparison of six x-vector front-ends used in conjunction with seven different back-end configurations for CMN2 measured in terms of minimum $C_{primary}$. Front-end configurations are shown in Table 4.*

Backend	G-PLDA						HT-PLDA	Average
Feature adaptation	–	–	–	CORAL	CORAL	CORAL	–	
PLDA adaptation	–	KALDI	CORAL+	–	KALDI	CORAL+	CORAL+	
front1	0.422	0.453	0.419	0.418	0.442	0.424	0.414	0.427
front2	0.431	0.458	0.420	0.422	0.445	0.425	0.405	0.429
front3	0.436	0.460	0.429	0.434	0.450	0.435	0.407	0.436
front4	0.507	0.515	0.487	0.502	0.509	0.497	0.486	0.500
front5	0.470	0.482	0.453	0.465	0.476	0.467	0.459	0.467
front6	0.441	0.455	0.424	0.428	0.448	0.428	0.416	0.434
Average	0.451	0.471	0.439	0.445	0.462	0.446	0.431	

obtained by fusing the scores from the 42 combinations. The fusion results are shown in Table 3. The single-best was chosen based on the development set, which was the `front3 GPLDA` with `CORAL+`.

We first look at the result row wise in Table 6. An average measure is provided for each row at the last column of the table. Comparing `front1` and `front2`, adding more training data and speakers from the Fisher corpus does not seem to improve the performance. Comparing `front1` and `front3`, increasing the number of target speakers via speaker augmentation (Type III audio augmentation) does not seems to affect the performance as well. Comparing `front4` to `front1`, `2` and `3`, we observe considerable degradation by dropping Type II audio augmentation, which involves the use of real RIR and AMR codec. Comparing `front6` to `front5`, we gain 7% relative improvement from mixed-band training than just downsampling wideband speech to 8 kHz in `front5`.

Next, we look at the result column wise. Take the results in the second column as the baseline, where no domain adaptation is applied. Compared to the baseline, embedding-level CORAL adaptation and model-level CORAL+ adaptation consistently improve the performance. On the other hand, KALDI's PLDA adaptation degrades the performance when used alone or in conjunction with CORAL, though we observe consistent improvement on the development set (not presented here in view of page limit). It could also be noted that CORAL+ outperforms CORAL adaptation, and there is no performance gain of using CORAL followed by CORAL+ adaptation.

4.3. Wideband VAST

For the VAST sub-task, we constructed 3 x-vector front-ends with the configurations shown in Table 5. The back-end was a G-PLDA trained on VoxCeleb. Notice that no domain adaptation is required as the train and evaluation sets are English audio-from-video (AfV) speech. Different from that of CMN2, test segments in the VAST partition contain multiple speakers, which calls for the use of speaker diarization prior to x-vector extraction. To this end, we used the *agglomerative hierarchical clustering* (AHC) method proposed in [31]. Let C_e and C_t be the number of clusters produced by the diarization step for enrollment and test sides, respectively. For a given trial, we compute $(C_e + 1) \times (C_t + 1)$ scores with the G-PLDA. Long recording without diarization is taken as an additional cluster to cater for the case of single-speaker utterance. The maximum value among the $(C_e + 1) \times (C_t + 1)$ scores is taken as the final score.

We show in Table 7 the EER and minimum C_{primary} for the three different x-vector front-ends. The fusion results are shown in Table 3. The single best (`front1`) was chosen based on the development set, while `front3` turns out to be the best on the evaluation set. Also shown in the Table 7 are the results on *speech in the wild* (SITW) `core-multi` sub-task. Among the three front-ends, `front1` was trained solely on wideband speech, while `front2` and `front3` were trained from mixed-band speech as described in Section 3.2. In particular, the narrowband SRE and SWB utterances were up-sampled to 16 kHz and bandwidth extended (BWE). Comparing `front3` to `front1`, we notice that mixed-band training improves considerably the performance. The relative improvement are 8.5% and 4.7% on EER and min C_{primary}, respectively. Comparing `front3` to `front2`, we notice the benefit of speaker augmentation, were the relative improvement amounts to 10.9% and 5.7% on EER and C_{primary}, respectively. Similar trends could be observed on the SITW core-multi task.

Table 7: *Comparison of three x-vector front-ends for VAST. Front-end configurations are shown in Table 5.*

VAST	SRE'18 Eval		SITW core-multi	
	EER (%)	Min Cost	EER (%)	Min Cost
front1	12.89	0.440	3.32	0.178
front2	13.23	0.445	3.26	0.180
front3	11.79	0.419	2.90	0.171

5. Conclusions

NIST SRE'18 was positioned to tackle more open-ended problems commonly encountered in practical deployment – domain mismatch, lack of in-domain labeled data, unsupervised domain adaptation, and multi-speaker test scenario. In dealing with the domain mismatch problem, we found two recently proposed methods, i.e., the CORAL and CORAL+ unsupervised domain adaptation techniques to be effective when labelled in-domain data is unavailable. We also explored mixed-bandwidth training of x-vector front-end with considerable success on both CMN2 and VAST partitions. We introduced speaker augmentation as the mean to increase the amount of training data and the number of target speakers in training the x-vector front-end. Speaker augmentation improves slightly the performance on VAST but not on CMN2. The issue of multi-speaker test scenario could be dealt with, to a certain extent, with unsupervised clustering. Nevertheless, there are still much room for improvement. To date, x-vector has become the de facto speaker embedding, replacing the i-vector representation which has been used for a decade. Moving forward, we foresee further improvement could be obtained by building a deeper network.

6. References

[1] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: from features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010.

[2] J. H. L. Hansen and T. Hasan, "Speaker recognition by machines and humans: a tutorial review," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74 – 99, 2015.

[3] K. A. Lee, B. Ma, and H. Li, "Speaker verification makes its debut in smartphone," *IEEE Signal Processing Society Speech and language Technical Committee Newsletter*, 2013.

[4] J. F. Bonastre, J. Kahn, S. Rosatto, and M. Ajili, "Forensic speaker recognition: mirages and reality," *Individual Differences in Speech Production and Perception*, 2015.

[5] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, Feb 2012.

[6] K. Jones, S. Strassel, K. Walker, D. Graff, and J. Wright, "Call my net corpus: A multilingual corpus for evaluation of speaker recognition technology," in *Proc. Interspeech 2017*, 2017, pp. 2621–2624. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2017-1521

[7] J. Tracey and S. Strassel, "Vast: A corpus of video annotation for speech technologies," in *Proc. Eleventh International Conference on Language Resources and Evaluation (LREC)*, 2018, pp. 4318–4321.

[8] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, T. Koshinaka, J. Zhang, and K. Shinoda, "The NEC-TT speaker verification system for SRE18," *NIST SRE 2018 Workshop*, 2018.

[9] National Institute of Standards and Technology, "NIST 2018 Speaker Recognition Evaluation Plan," *NIST SRE*, 2018.

[10] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech 2017*, 2017, pp. 2616–2620.

[11] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, "The speakers in the wild (sitw) speaker recognition database," in *Interspeech 2016*, 2016, pp. 818–822.

[12] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP*, 2018, pp. 5329–5333.

[13] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[14] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Interspeech 2015*, 2015, pp. 3214–3218.

[15] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," in *Proc. Interspeech*, 2018, pp. 2252–2256.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[17] D. Snyder, G. Chen, and D. Povey, "MUSAN: a music, speech, and noise corpus," in *arXiv:1510.08484*, 2015.

[18] T. Ko, V. Peddinti, M. S. Daniel Povey, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. IEEE ICASSP*, 2017, pp. 5220–5224.

[19] L. Ferrer, H. Bratt, L. Burget, H. Cernocky, O. Glembek, M. Graciarena, A. Lawson, Y. Lei, P. Matejka, O. Plchot *et al.*, "Promoting robustness for speaker modeling in the community: the PRISM evaluation set," in *Proceedings of NIST 2011 workshop*, 2011.

[20] K. Kinoshita, M. Delcroix, S. Gannot, E. A. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj *et al.*, "A summary of the REVERB challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, p. 7, 2016.

[21] H. Yamamoto, K. A. Lee, K. Okabe, and T. Koshinaka, "Speaker augmentation and bandwidth extension for deep speaker embedding," in *Interspeech*, 2019, accepted.

[22] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1469–1477, Sep. 2015.

[23] S. Ioffe, "Probabilistic linear discriminant analysis," in *roceedings of the 9th European Conference on Computer Vision*, 2006.

[24] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. ICCV*, 2007, pp. 1–8.

[25] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Odyssey: Speaker and Language Recognition Workshop*, 2010.

[26] A. Silnova, N. Brümmer, Garcia-Romero, D. David Snyder, and L. Burget, "Fast variational bayes for heavy-tailed plda applied to i-vectors and x-vectors," in *Proc. Interspeech*, 2018.

[27] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proc. AAAI, vol. 6*, 2016, p. 8.

[28] J. Alam, G. Bhattacharya, and P. Kenny, "Speaker verification in mismatched conditions with frustratingly easy domain adaptation," in *Odyssey: Speaker and Language Recognition Workshop*, 2018.

[29] K. A. Lee, Q. Wang, and T. Koshinaka, "The CORAL+ algorithm for unsupervised domain adaptation of PLDA," in *IEEE ICASSP*, 2019, pp. 5821–5825.

[30] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE workshop on automatic speech recognition and understanding (ASRU)*, 2011.

[31] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Proceedings of the IEEE Spoken Language Technology Workshop*, 2014.