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

An End-to-End Text-independent Speaker Verification Framework with a Keyword Adversarial Network

Sungrack Yun, Janghoon Cho, Jungyun Eum, Wonil Chang, Kyuwoong Hwang

Qualcomm AI Research†, Qualcomm Korea YH

{sungrack, janghoon, c_jeum, wichang, kyuwoong}@qti.qualcomm.com

Abstract

This paper presents an end-to-end text-independent speaker verification framework by jointly considering the speaker embedding (SE) network and automatic speech recognition (ASR) network. The SE network learns to output an embedding vector which distinguishes the speaker characteristics of the input utterance, while the ASR network learns to recognize the phonetic context of the input. In training our speaker verification framework, we consider both the triplet loss minimization and adversarial gradient of the ASR network to obtain more discriminative and text-independent speaker embedding vectors. With the triplet loss, the distances between the embedding vectors of the same speaker are minimized while those of different speakers are maximized. Also, with the adversarial gradient of the ASR network, the text-dependency of the speaker embedding vector can be reduced. In the experiments, we evaluated our speaker verification framework using the LibriSpeech and CHiME 2013 dataset, and the evaluation results show that our speaker verification framework shows lower equal error rate and better text-independency compared to the other approaches.

Index Terms: text-independent speaker verification, end-to-end system, speaker embedding, adversarial training, triplet loss

1. Introduction

With the increasing number of researches, developments, and improvements on automatic speech recognition (ASR) [1–4], speaker verification (SV) [5–10], and spoken dialog system [11, 12], the voice interface has been widely adopted in various artificial intelligent (AI) applications such as mobile phones, smart home IoT devices, and automotive infotainment system. Especially, the speaker verification and recognition have been crucial components in several AI speakers [2, 4, 13, 14] for user authentication and personalized responses: given the user's voice command, the AI speaker can identify the user's voice and provide a user-specific services, e.g. music recommendation, equalizer adjustment, or schedule notification.

A number of researches on speaker verification and recognition have been proposed [5, 6, 8–10, 15]. In [9], the authors presented an end-to-end text-independent speaker verification for variable utterance length by applying a spatial pyramid pooling layer to the inception-resnet architecture. In [6], a simple pre-processing method to select noise-invariant frames from utterance was proposed for the text-independent SV system in unknown noisy environments. In [5,8], the tuple-based end-to-end (TE2E) loss and generalized end-to-end (GE2E) loss were introduced to improve speaker verification models. In [10], a deep neural network (DNN) was used to extract phonetically-aware i-vector and also bottleneck feature for short, text-constrained utterances. In [15], a data augmentation technique was investigated to improve the speaker recognition performance with the DNN-based embeddings trained to discriminate speakers.

This paper presents an end-to-end text-independent SV framework by jointly considering two components: the speaker embedding (SE) network and the ASR network. It can be considered that the ASR and SV are inversely-related. The ASR network distinguishes and classifies the phonetic context of the input utterances (text-dependent) from any speakers (speaker-independent), while the SE network extracts the speaker's identity (speaker-dependent) regardless of the input text-phrases (text-independent). Inspired from this property, we propose a text-independent SV framework where the SE network is combined jointly with the ASR network, as illustrated in Fig. 1. The SE network takes the raw speech waveform as input and outputs the speaker embedding vector using a deep end-to-end architecture consisting of a number of residual blocks [16–18], convolution layers, and an attention layer. The ASR network classifies the phonetic context of the embedding vector, and the adversarial gradient [19–22] is applied to the SE network such that the embedding vector is trained to be text-independent. Although we may also apply the adversarial gradient from the SE network to the ASR network for the speaker-independency in ASR, we only focus on the text-independent SE network in this research. In training SE network, we also combine the triplet loss [9] together with the adversarial gradient to obtain more discriminative speaker embedding vectors: the distances between the embedding vectors of the same speaker are minimized while those of different speakers are maximized.

The proposed SV framework was evaluated using the LibriSpeech [23] and CHiME 2013 dataset [24]. In the evaluation results, our SV framework shows lower equal error rate (EER) and better text-independent property compared to the other approaches.

2. Speaker Verification

Speaker verification is a decision process of accepting or rejecting an input utterance $\mathbf{x}$ based on the speaker characteristics, and it can be accomplished by comparing $\mathbf{x}$ with the reference speaker model $\mathbf{X}_{ref}$ as:

$$f(\mathbf{X}_{ref}, \mathbf{x}) \underset{reject}{\overset{accept}{\gtrless}} \tau \tag{1}$$

where $f(\cdot,\cdot)$ measures the similarity score between $\mathbf{X}_{ref}$ and $\mathbf{x}$. If the score is greater than a pre-defined threshold τ, $\mathbf{x}$ is accepted as a reference speaker's utterance; otherwise, it is rejected. The input observation $\mathbf{x}$ can be a raw speech waveform itself or an encoded vector using various feature extraction algorithms for speaker verification such as Mel-frequency cepstral coefficients (MFCCs) [25], i-vector [26–29], or speaker embedding vectors [5, 7, 8, 15]. In this paper, we model the raw speech

† Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

http://dx.doi.org/10.21437/Interspeech.2019-2208

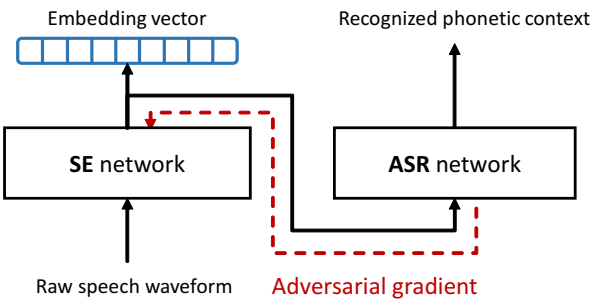

Figure 1: *Block diagram of the proposed SV framework: adversarial gradient from the ASR is used to obtain text-independent speaker embedding vector. The SE network can verify the user's voice independently of text-phrases.*

waveform directly and extract a D-dimensional speaker embedding vector for $\mathbf{x}$. The reference speaker model $\mathbf{X}_{ref}$ contains M enrollment embedding vectors $\mathbf{X}_{ref} = \{\mathbf{x}_{e_1}, ..., \mathbf{x}_{e_M}\}$, and we define the score function $f(\cdot,\cdot)$ based on the cosine similarity:

$$f(\mathbf{X}_{ref}, \mathbf{x}) = \frac{\mathbf{x} \cdot \mathbf{x}_e^{cent}}{||\mathbf{x}|| \, ||\mathbf{x}_e^{cent}||} \tag{2}$$

where $\mathbf{x}_e^{cent} = \sum_{i=1}^{M} \mathbf{x}_{e_i}/M$.

An example of speaker model enrollment and test vector verification is illustrated in Fig. 2: the color and texture of the embedding vectors represent the speaker and phonetic context, respectively. Here, we assume that the reference speaker model has the enrollment utterance of two different text phrases (diagonal and dotted). The test vector of the reference speaker, $\widetilde{\mathbf{x}}_1$, is from an unseen text phrase (dashed), and the test vector of an imposter, $\widetilde{\mathbf{x}}_2$, is from the seen text phrase (diagonal). Ideally, in the text-independent SV, $\widetilde{\mathbf{x}}_1$ should be accepted independently of the text phrase. In real-cases, however, $\widetilde{\mathbf{x}}_1$ may be falsely rejected due to its low score since $\mathbf{X}_{ref}$ does not contain the enrollment vector of the phrase (dashed). On the contrary, $\widetilde{\mathbf{x}}_2$ may be falsely accepted because the same phrase utterance (diagonal) exists in the enrollment set. As described in this example, the text-independent SV may have a performance degradation when the reference speaker model is enrolled with the vectors from few specific text-phrases. This may often happen in real-applications: a speaker model is enrolled with one or two voice commands, and the user speaks the other voice commands which need to be verified. In the next section, we will describe our text-independent SV framework where the embedding vector is extracted from a deep end-to-end neural net architecture with an adversarial ASR network.

3. Proposed SV Framework

As illustrated in Fig 1, the proposed SV framework consists of two components: the SE network and the ASR network. The SE network takes the raw speech waveform as the input and outputs an embedding vector, and the ASR network takes the embedding vector as the input and outputs the recognized phonetic context. In training the SE network, we use the adversarial gradient from the ASR to encourage the SE network to extract the embedding vector which is phonetically-independent and contains only speaker characteristics of the input.

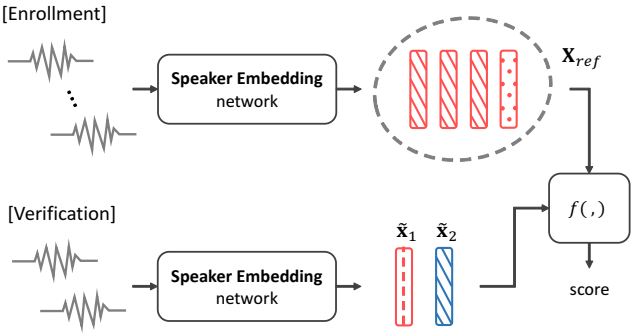

Figure 2: *The process of speaker model enrollment and test vector verification. Color (red, blue) and texture (diagonal, dotted, dashed) represent speaker and text phrase, respectively.*

3.1. Deep End-to-End SE Network

As illustrated in Fig. 3, the proposed end-to-end SE network extracts 128-dimensional embedding vector from raw speech waveform with the architecture of 10 conv-res units, 5 residual blocks [16–18], and one attention layer. Each conv-res unit consists of 1-dimensional convolution layer and residual block. From the first to seventh conv-res unit, the number of input-output channels in convolution is doubled (i.e. $1 \rightarrow 2 \rightarrow \cdots \rightarrow 128$) while it keeps the same number for remaining 3 conv-res units. The convolution width of the first conv-res unit is the length of the input waveform, and it is reduced by half at every conv-res unit (from the first to tenth). The residual block, illustrated in Fig. 3, consists of two convolution layers with batch normalization [30] and Relu, and a shortcut is connected from the input to output. The output of 5 residual blocks is combined with an attention layer to focus on more relevant region of the input, and finally we obtain the 128-dimensional embedding vector by averaging over the width (time-average).

3.2. Training Loss for SE Network

In training the SE network, we use the triplet loss and the adversarial gradient from the ASR network to obtain more discriminative and text-independent speaker embedding vectors.

3.2.1. Triplet loss

The objective of the triplet loss is to maximize the similarity between the embedding vectors of the same speakers while minimize that of different speakers:

$$f(\mathbf{x}_a, \mathbf{x}_p) - \delta > f(\mathbf{x}_a, \mathbf{x}_n) \tag{3}$$

where δ quantifies the minimum margin between two similarities. Given the anchor $\mathbf{x}_a$, the positive sample $\mathbf{x}_p$ is selected from the same speaker with the anchor while the negative $\mathbf{x}_n$ is selected from the different speaker. With the cosine similarity and normalized embedding vectors, the inequality (3) becomes

$$\cos\theta_{ap} > \cos\theta_{an} + \delta \tag{4}$$

where θ_{ap} and θ_{an} are respectively the angle between $\mathbf{x}_a$ and $\mathbf{x}_p$, and the angle between $\mathbf{x}_a$ and $\mathbf{x}_n$. As illustrated in Fig. 4, the minimization of the triplet loss pulls together the anchor and positive vector, while pushes apart the negative from the anchor vector. In the training, we can only choose the triplets which violate the condition in (3), and the loss can be expressed as

$$L_{triplet} = -\min(\cos\theta_{ap} - \cos\theta_{an}, \delta). \tag{5}$$

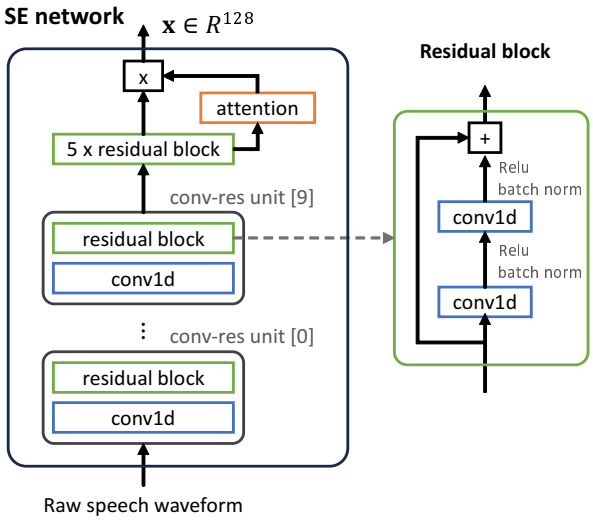

Figure 3: *The proposed end-to-end SE network: 10 conv-res units, 5 residual blocks, and one attention layer are used to extract the embedding vector* $\mathbf{x}$ *from raw speech waveform.*

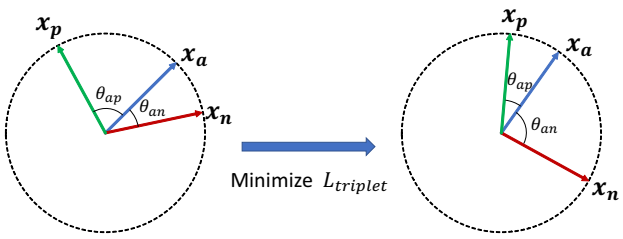

Figure 4: *Training with the triplet loss pulls the positive vector towards the anchor while pushes the negative vector away from the anchor.*

3.2.2. *Adversarial training loss from the ASR*

As illustrated in Fig. 1, the ASR network is combined with the SE network, and the adversarial gradient from the ASR is used to train the SE network for the text-independent speaker embedding vector. The objective of the ASR network is to accurately recognize the phonetic context of the embedding vector, and the network can be trained by minimizing the cross entropy between the true target label y and the recognized label $\hat{y}$ (output of the ASR network):

$$L_{ASR} = -\sum_{y} y \log(\hat{y}) \tag{6}$$

where the recognition unit y can be a character, phoneme or word. In this research, we use the word unit for y, and a DNN is used for the ASR network where the final soft-max layer gives the classification decision of N keywords. Since the ASR is used *adversarially* in training the SE network, we can obtain the embedding vector which does not discriminate the word difference, i.e. text-independent.

Combining the triplet loss in Eq. (5) with the adversarial training loss from the ASR, we can obtain the entire loss to train the SE network:

$$L_{SE} = L_{triplet} - \gamma L_{ASR} \tag{7}$$

where γ is a pre-determined value which controls the adversarial factor.

4. Experiments

4.1. Dataset and Training

The proposed SV framework is trained by two stages: training the baseline SE network and fine-tuning the SE network using the triplet loss combined with the ASR-adversarial loss. In training the baseline SE network, we used the LibriSpeech dataset [23] which contains 1,000 hours of 2,400 speakers' recordings based on the text from Project Gutenberg [31]. First, we randomly segmented the LibriSpeech utterances into the audio samples of a length between 1.5 and 2.0 sec. And, given these segmented audio samples, we trained the baseline SE network with a final softmax layer which is designed to classify 2,400 speakers by minimizing the cross entropy loss.

In fine-tuning the baseline SE network, we used the CHiME 2013 database [24] which was created for the 2nd speech separation and recognition challenge with two tracks. In this experiment, we chose the track1 database consisting of keyword utterances from 34 speakers. Each utterance in CHiME database consists of a sequence of six words: command, color, preposition, letter, number, and adverb. For the short-keyword SV experiment, we used only the first two words by segmenting the utterances using the word boundary labels provided by the database. With this segmentation, we obtained the utterances of 16 different keywords: 4 types of commands (*bin*, *place*, *set*, *lay*) followed by 4 types of colors (*white*, *red*, *green*, *blue*). We chose two, three, and four keywords among them to construct the ASR network classifying N = 2, 3, and 4 keywords. In this experiment setup, the number of keywords to classify is small, and thus we used one-layer DNN for the ASR network given the 128-dimensional input embedding vector. The database was split into the training-validation set of 24 speakers and the evaluation set of 10 speakers without any speaker overlap. In the training, we chose only one keyword data for a speaker, and the other keywords' data were used for the validation. For example, Fig. 5 shows the chosen keywords and speakers for the training and validation set when $N = 3$.

The networks were optimized with stochastic gradient descent with momentum of 0.9 and weight decay of 0.00001 for the baseline training and fine-tuning stages. Also, we applied

	Keyword 1 *binwhite*	**Keyword 2** *placered*	**Keyword 3** *setgreen*
Speaker Group 1 : speaker[0]-[7]	Training	Validation	Validation
Speaker Group 2 : speaker[8]-[15]	Validation	Training	Validation
Speaker Group 3 : speaker[16]-[23]	Validation	Validation	Training

: Used for training dataset

: Used for validation dataset

Figure 5: *An example of chosen keywords and speakers for the training and validation set when* N*=3.*

Table 1: *The EER (%) of four different SV models (GMM-BUM, DeepRes-Base, DeepRes-GE2E and DeepRes-TriKwdAdv) with different number of keywords ($N = 2, 3$, and 4).*

N	model	TK	NTK	Avg.
2	GMM-UBM	**1.76**	20.85	11.31
	DeepRes-Base	2.72	9.95	6.34
	DeepRes-GE2E	1.83	8.75	5.29
	DeepRes-TriKwdAdv	1.96	**4.76**	**3.36**
3	GMM-UBM	2.48	17.78	10.13
	DeepRes-Base	2.68	8.10	5.39
	DeepRes-GE2E	2.23	8.16	5.20
	DeepRes-TriKwdAdv	**2.16**	**5.15**	**3.65**
4	GMM-UBM	2.55	17.68	10.12
	DeepRes-Base	3.10	7.45	5.27
	DeepRes-GE2E	**1.90**	6.44	4.17
	DeepRes-TriKwdAdv	2.19	**5.32**	**3.76**

different values of γ to see the effect of adversarial factor.

4.2. Evaluation Result

With the fine-tuned SE network, we performed the speaker verification using the test data: for each speaker, 5 utterances were used to obtain the enrollment vectors $\mathbf{X}_{ref}$, and remaining utterances were used as verification. For the text-independent SV experiment, we chose all 5 enrollment utterances from the same keyword while the verification utterances contain both the same keyword (target keyword: TK) and different keywords (non-target keyword: NTK) from the enrollment. Similar experiment setup can be found in [8] where two keywords ('OK-Google' and 'Hey-Google') were differently used in the enrollment and verification data.

With this experiment setup, we evaluated four different SV models: GMM-UBM and the proposed SV frameworks without fine-tuning (DeepRes-Base), fine-tuned with GE2E-loss [8] (DeepRes-GE2E), and fine-tuned with triplet and keyword-adversarial loss (DeepRes-TriKwdAdv). In Table. 1, the performances of four SV models with different numbers of keywords (N = 2, 3, and 4) are summarized. For the performance metric, we used the EER of TK and NTK: TK is the verification performance when testing the same keyword with the enrollment keyword, and NTK is that when testing the different keyword from the enrollment keyword. Thus, better performance of NTK shows less text-dependency in speaker verification. As expected, the table shows that the EER of NTK is higher than that of TK. Especially, the EER difference between TK and NTK in GMM-UBM is quite high since the model is trained with only the target keyword, i.e. text-dependent model. The DeepRes-Base also shows big EER difference between TK and NTK. The GE2E loss which is proposed for the text-independent SV [8] improved the EER over the baseline model (DeepRes-Base), but only marginal improvements were observed. With the proposed model, DeepRes-TriKwdAdv, we obtained considerable improvements over the GMM-UBM and also GE2E in evaluating NTK. The Avg. in the table is the mean of TK EER and NTK EER. Overall, the DeepRes-TriKwdAdv obtained the best performance for all cases: N=2, 3, and 4.

In Table. 2, the performances of the proposed SV framework with different numbers of keywords (N = 2, 3, and 4) and adversarial factors (γ = 0.0, 0.2, and 0.4) are summarized. The factor $\gamma = 0.0$ means no adversarial training was applied: only triplet loss was used to obtain the model. When $\gamma = 0.0$, the keyword accuracies (Kwd Acc.) of the ASR network show high performance for all cases of $N = 2, 3$, and 4. With increasing of γ, the Kwd Acc. decreases and also the EER of the NTK decreases. These results show that the keyword-adversarial training reduces the keyword dependency of the speaker embedding vectors. For all cases (N = 2, 3, and 4), the best performance was obtained when $\gamma = 0.4$.

Table 2: *The performance of the proposed SV framework with different number of keywords (N = 2, 3, and 4) and adversarial factors (γ = 0.0, 0.2, and 0.4). We use the performance metric, EER (%) of TK and NTK. The Avg. is the mean of TK and NTK, and Kwd Acc is the accuracy of the ASR network.*

N	γ	TK	NTK	Avg.	Kwd Acc.
2	0.0	2.46	9.89	6.17	98.61
	0.2	2.33	6.22	4.27	47.22
	0.4	1.96	4.76	3.36	50.38
3	0.0	2.27	7.79	5.03	98.61
	0.2	2.59	6.45	4.52	36.11
	0.4	2.16	5.15	3.65	34.72
4	0.0	2.84	7.49	5.16	95.83
	0.2	2.32	6.35	4.34	26.39
	0.4	2.19	5.32	3.76	27.78

5. Conclusion and Future Work

In this paper, we presented an end-to-end text-independent speaker verification framework by considering the SE network and ASR network jointly. The SE network takes the raw waveform and outputs the embedding vector which distinguishes the speaker characteristics of the input utterance. In this research, we used a number of residual blocks, convolution layer, and attention for the SE network. To obtain more discriminative and text-independent speaker embedding vectors, we consider both the triplet loss and ASR network in training the SE network. The triplet loss maximizes the similarity between the embedding vectors of the same speaker and also minimizes that between those of different speakers. The ASR network is trained to recognize the phonetic context of the input, and using the adversarial gradient from the ASR network, the text-dependency of the embedding vectors can be reduced. In this research, we used one-layer DNN for ASR to classify N=2, 3, and 4 isolated keywords. We evaluated our SV framework using the LibriSpeech and CHiME database. With the LibriSpeech database, we trained the baseline SE network. And, then we fine-tuned the baseline model using the CHiME database by applying the proposed triplet and ASR-adversarial loss. For the short-keyword SV evaluation, we segmented the first two words of the utterances in CHiME database. In the experiments, we compared the proposed SV framework (DeepRes-TriKwdAdv) with the other algorithms (GMM-UBM, DeepRes-Base, and DeepRes-GE2E) which do not utilize the ASR network for the text-independent SV. In all experiments, DeepRes-TriKwdAdv outperformed the other SV models in EER. Especially, in the evaluation of NTK, the DeepRes-TriKwdAdv shows a considerable improvement over the DeepRes-Base. In this research, we set the ASR network as an isolated word classifier. For the further work, we will continue this work for more general cases: train the SE network with a general speech recognizer.

6. References

[1] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. Nahamoo, "Direct acoustics-to-word models for english conversational speech recognition," *CoRR*, vol. abs/1703.07754, 2017.

[2] B. Li, T. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Pundak, K. Chin *et al.*, "Acoustic modeling for Google home," *Proceedings of the INTERSPEECH*, pp. 399–403, 2017.

[3] R. Prabhavalkar, K. Rao, T. N. Sainath, B. Li, L. Johnson, and N. Jaitly, "A comparison of sequence-to-sequence models for speech recognition," in *Proceedings of the INTERSPEECH*, 2017, pp. 939–943.

[4] Y. Zhao, J. Li, S. Zhang, L. Chen, and Y. Gong, "Domain and speaker adaptation for cortana speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5984–5988.

[5] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5115–5119.

[6] S. Song, S. Zhang, B. Schuller, L. Shen, and M. Valstar, "Noise invariant frame selection: a simple method to address the background noise problem for text-independent speaker verification," *CoRR*, vol. abs/1805.01259, 2018.

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

[8] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[9] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with flexibility in utterance duration," in *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017, pp. 584–590.

[10] J. Zhong, W. Hu, F. Soong, and H. Meng, "DNN i-vector speaker verification with short, text-constrained test utterances," *Proceedings of the INTERSPEECH*, pp. 1507–1511, 2017.

[11] B. Liu and I. Lane, "Iterative policy learning in end-to-end trainable task-oriented neural dialog models," in *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 482–489.

[12] T. Ranzenberger, C. Hacker, F. Gallwitz, and N. Germany, "Integration of a Kaldi speech recognizer into a speech dialog system for automotive infotainment applications," in *Conference on Electronic Speech Signal Processing (ESSV), Ulm*, 2018.

[13] G. López, L. Quesada, and L. A. Guerrero, "Alexa vs. Siri vs. Cortana vs. Google assistant: a comparison of speech-based natural user interfaces," in *Proceedings of International Conference on Applied Human Factors and Ergonomics*. Springer, 2017, pp. 241–250.

[14] A. Purington, J. G. Taft, S. Sannon, N. N. Bazarova, and S. H. Taylor, "Alexa is my new BFF: social roles, user satisfaction, and personification of the amazon echo," in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 2017, pp. 2853–2859.

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

[16] D. Cai, Z. Cai, and M. Li, "Deep speaker embeddings with convolutional neural network on supervector for text-independent speaker recognition," in *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2018, pp. 1478–1482.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[18] ——, "Identity mappings in deep residual networks," in *Proceedings of European Conference on Computer Vision*, 2016, pp. 630–645.

[19] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016. [Online]. Available: http://jmlr.org/papers/v17/15-239.html

[20] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[21] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech enhancement generative adversarial network," *CoRR*, vol. abs/1703.09452, 2017.

[22] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *CoRR*, vol. abs/1511.06434, 2015.

[23] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[24] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The second CHiME speech separation and recognition challenge: Datasets, tasks and baselines," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 126–130.

[25] 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, 2012.

[26] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, pp. 2015–2028, 2013.

[27] N. Dehak, P. J. 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, 2011.

[28] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Proceedings of IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 413–417.

[29] A. Woubie, J. Luque, and J. Hernando, "Improving i-vector and PLDA based speaker clustering with long-term features," in *Proceedings of the INTERSPEECH*, 2016, pp. 372–376.

[30] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the International Conference on Machine Learning*, 2015, pp. 448–456.

[31] *http://www.gutenberg.org*.