PRETRAINING CONFORMER WITH ASR FOR SPEAKER VERIFICATION

Danwei Cai[1], *Weiqing Wang*[1], *Ming Li*[1,2], *Rui Xia*[3], *Chuanzeng Huang*[3]

[1]Department of Electrical and Computer Engineering, Duke University, Durham, USA
[2]Data Science Research Center, Duke Kunshan University, Kunshan, China
[3]Speech, Audio, and Music (SAMI) Group, Bytedance, China
`ming.li369@duke.edu, {rui.xia,huangchuanzeng}@bytedance.com`

ABSTRACT

This paper proposes to pretrain Conformer with automatic speech recognition (ASR) task for speaker verification. Conformer combines convolution neural network (CNN) and Transformer model for modeling local and global features, respectively. Recently, multi-scale feature aggregation Conformer (MFA-Conformer) has been proposed for automatic speaker verification. MFA-Conformer concatenates frame-level outputs from all Conformer blocks for further pooling. However, our experiments show that Conformer can be easily overfitted with limited speaker recognition training data. To avoid overfitting, we propose to transfer the knowledge learned from ASR to speaker verification. Specifically, an ASR pretrained Conformer is used to initialize the training of MFA-Conformer for speaker verification. Our experiments show that pretraining Conformer with ASR leads to significant performance gains across model sizes. The best model achieves 0.48%, 0.71% and 1.54% EER on Voxceleb1-O, Voxceleb1-E, and Voxceleb1-H, respectively.

Index Terms— speaker verification, Transformer, Conformer, pretraining, speaker recognition

1. INTRODUCTION

Speaker verification analyzes the voice pattern of the speech signal to verify the speakers identity. Over the past five years, the performance of speaker verification systems has significantly improved thanks to the application of deep neural networks (DNN) [1, 2]. Various network architecture [3, 4, 5, 6], training objectives [7, 8, 9], and training strategies [10, 11] are proposed for speaker verification systems. In terms of network architecture, convolution neural networks (CNN) and time delay neural networks (TDNN) have been the de facto choice for speaker verification tasks as their abilities to model local feature patterns and extract speaker characteristics. Variants of CNN and TDNN with residual connection [12], squeeze and excitation operation [13, 6], Res2Net block [14, 5, 6], and ResNeXt block [15, 5] further push the limit of speaker verification performance.

While CNNs and TDNNs are good at modeling local information, they are less effective for long-range global context extraction without deeper layers. In contrast, Transformers are more capable of capturing longer context with less fine-grained local patterns [16]. To better model both local and global feature patterns, Conformer combines the convolution module with Transformer and shows promising results on end-to-end speech recognition [17] and speech separation [18]. Recently, Zhang *et al.* proposed a multi-scale feature aggregation Conformer (MFA-Conformer) for speaker verification. MFA-Conformer concatenates frame-level outputs from all Conformer blocks to aggregate multi-scale representations [19]. However, as shown in [19] and our experiments, MFA-Conformer can be easily overfitted to the training set when the model size is increased. In this paper, we propose to use ASR pretrained Conformer for speaker verification to avoid overfitting for large models.

An ASR Conformer is trained to transcribe text from a speech signal and is thus equipped to model phonetic information. As shown in several studies, phonetic information helps to learn speaker information in speaker embedding network [20] as well as the statistical model of i-vector [21, 22]. Thus, an ASR pretrained Conformer may help model speaker characteristics in different phonetic units and extract robust speaker representations.

Several studies have explored using self-supervised pretrained Transformers for speaker verification. Fan *et al.* [23] and Vaessen *et al.* [24] directly fine-tune speaker verification model on the pretrained model with additional pooling layer. However, this method does not outperform the CNN- or TDNN-based model, which typically has smaller parameters than the pretrained Transformer. Another method substitutes the handcrafted feature with the pretrained frame-level feature to train speaker embedding networks [25, 26]. This method obtains satisfactory performance with large pretrained parameters and requires an additional speaker embedding network. In this paper, instead of self-supervised pretrained Transformers, ASR pretrained Conformer is used as the network backbone for the speaker embedding network. Fine-tuning is directly applied on the pretrained Conformer, and no additional

Corresponding author: Ming Li

speaker network is required. This transfer learning strategy of fine-tuning a pre-trained ASR Conformer transfer the knowledge learned from ASR to speaker verification.

2. METHODS

2.1. MFA-Conformer

This section describes the architecture of the MFA-Conformer speaker embedding network [19]. Conformer is adopted as the network backbone for speaker embedding network. It consists of a convolution subsampling layer and several Conformer blocks. Outputs of all Conformer layers are concatenated before an attentive statistics pooling layer is used to generate utterance-level representation. Fully connected layers are employed afterward to extract the speaker embedding and classify training speakers.

2.1.1. Conformer block

The conformer block combines CNN and Transformer to capture global and local information from the spectral feature. Figure 1 shows a Conformer building block [17]. It contains two feed forward networks (FFN) that sandwich a multi-head self-attention (MHSA) module followed by a convolution (Conv) module. The MHSA employs the relative sinusoidal positional encoding scheme from Transformer XL [27], which encodes the relative distance between input features. Unlike absolute positional encoding, relative positional encoding can generalize to sequences of unseen lengths, as it only encodes the relative pairwise distance between two frames theoretically. Therefore, the encoders trained with relative positional encoding are more robust to the variance of the utterance length [17]. The subsequent convolution module contains a point-wise convolution and a gated linear unit (GLU) followed by a single 1-D depth-wise convolution layer. Batch normalization is then employed to aid training deep models, followed by a swish activation and point-wise convolution with dropout. Residual connections are employed between blocks, except that the feed-forward layers have half-step residual connections. Layer normalization is applied on top of the Conformer block before output. Mathematically, given an input $\mathbf{h}_{i-1} \in \mathbb{R}^{d\times T}$, the output $\mathbf{h}_i \in \mathbb{R}^{d\times T}$ of the i-th Conformer block is:

$$\begin{aligned}
\tilde{\mathbf{h}}_i &= \mathbf{h}_{i-1} + \frac{1}{2}\text{FFN}(\mathbf{h}_{i-1}) \\
\mathbf{h}'_i &= \tilde{\mathbf{h}}_i + \text{MHSA}(\tilde{\mathbf{h}}_i) \\
\mathbf{h}''_i &= \mathbf{h}'_i + \text{Conv}(\mathbf{h}'_i) \\
\mathbf{h}_i &= \text{LayerNorm}(\mathbf{x}''_i + \frac{1}{2}\text{FFN}(\mathbf{x}''_i))
\end{aligned} \tag{1}$$

where d denotes the Conformer dimension, T denotes the frame length.

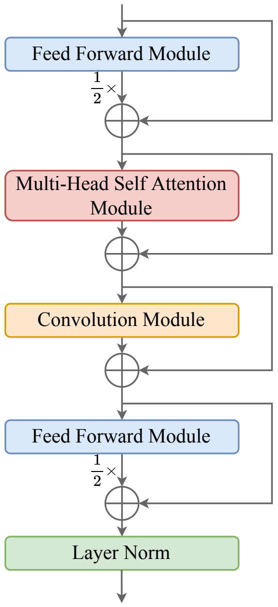

Fig. 1: A conformer building block.

2.1.2. Multi-scale feature aggregation

Multi-scale feature aggregation (MFA) concatenates frame-level outputs from all Conformer blocks:

$$\begin{aligned}
\mathbf{H}' &= \text{Concat}(\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_L) \\
\mathbf{H} &= \text{LayerNorm}(\mathbf{H}')
\end{aligned} \tag{2}$$

where L is the number of Conformer blocks, and $\mathbf{H}' \in \mathbb{R}^{D\times T}$ with $D = L \times d$. This aggregation of different-level representations contributes towards robust speaker embeddings and improves the performance of speaker verification [6].

Attentive statistics pooling is then applied to the concatenated outputs $\mathbf{H}$ to produce an utterance-level representation with fixed dimension.

2.2. Fine-tuning on ASR Conformer

Generally, deeper Transformers obtain better results with more training data [26, 28]. However, it is widely believed in the literature that large datasets are required for training deep Transformers from scratch [29]. Experiments in [19] show that increasing the number of layers of Conformer decreases the speaker verification performance, which indicates overfitting of the Conformer model. To avoid overfitting during training, we propose to use ASR pretrained Conformer for MFA-Conformer based speaker embedding network. This transfer learning strategy transfer the knowledge learned from ASR to speaker verification. ASR pretrained Conformers, which have the ability ZASLZto model phonetic information, may help model speaker characteristics in different phonetic units and extract robust speaker representations.

Table 1: Three ASR Conformer encoders

Model	#layers	#dim	#heads	hidden units
Small[1]	16	176	4	704
Medium[2]	18	256	4	1024
Large[3]	18	512	8	2048

Specifically, the parameter of the ASR pretrained Conformer encoder is used to initialize the MFA-Conformer speaker embedding network. We first fix the parameters of the Conformer encoder and only update the parameters of the pooling and speaker classification layers for several training epochs. Then, all the parameters of the MFA-Conformer are jointly fine-tuned.

3. EXPERIMENTAL SETUP

3.1. Dataset

The experiments are conducted on VoxCeleb [30, 31]. For model training, the development set of VoxCeleb 2 is used. Training data contains 1,092,009 audio files from 5,994 speakers.

Speed perturbation-based augmentation is applied by speeding up or down with the factor of 1.1 and 0.9, respectively. This produces two times extra copies of the original training data, resulting in an enlarged dataset with 17,982 speakers and 3,276,027 utterances.

For the enlarged training data, we use additive background noise or convolutional reverberation noise for data augmentation. MUSAN dataset [32] is used as the noise source. Addictive noises include ambient noise, music, and babble noise. To create babble noise, three to eight speech files are mixed. We randomly set signal-to-noise ratios (SNR) between 0 to 20 dB. The convolution operation is performed for the reverberation noise with 40,000 simulated room impulse responses (RIR) in [33]. We only use RIRs from small and medium rooms. We apply on-the-fly data augmentation with a probability of 0.6 during training.

3.2. ASR Conformer pretraining

We use the pretrained ASR model in NEMO toolkit [34]. The ASR Conformer has same encoder as in [17] but uses linear decoder and connectionist temporal classification (CTC) for decoding. Three ASR conformers with different hyperparameters are used in our experiments. All models have the same convolution subsampling rate of $\frac{1}{4}$. And the kernel size for the convolution module is 31. Table 1 shows the differences between the three models regarding the number of conformer layers, the encoder dimension, attention heads, linear hidden units, and convolution kernel size.

All Conformer-CTC models are trained with English corpora from 10 different datasets, which add up to around 34,000 hours of speech data.

3.3. Network configuration

Speech utterances are randomly cropped to 2 seconds for speaker embedding network training. Logarithmical Mel-spectrogram with 80 frequency bins is extracted as the acoustic feature. Mel-spectrograms are computed over Hamming windows of 20ms shifted by 10ms.

During training, Additive angular margin (AAM) loss [7] with a re-scaling factor of 32, and angular margin of 0.2 is used to learn discriminative representations. The dimension of speaker embedding is 256. AdamW is used as the optimizer with an initial learning rate of 0.001. Cosine annealing learning rate scheduler and 4000 steps warming up are applied. The batch size is set to 512, and weight decay is set to 1e-7.

We applied large margin fine-tuning [11] on the converged model. The speech is cropped to 6 seconds, and the angular margin of AAM loss is increased to 0.6. Data augmentation of speed perturbation is disabled, so the training data falls back to the original set.

3.4. Evaluation

For evaluation, the development and test sets of Voxceleb 1 are used. We report the speaker verification results on three trial lists of VoxCeleb 1-O, Voxceleb 1-E and Voxceleb 1-H as defined in [31]. System performance is reported as equal error rate (EER) and minimum detection cost (minDCF). The parameters of the detection cost function are set as: $C_{\text{Miss}} = 1$, $C_{\text{FA}} = 1$, $P_{\text{Target}} = 0.01$.

Adapted s-norm [35] is applied after cosine similarity scoring. 30,000 utterances are randomly selected from training data and used as the imposter cohort for score normalization. The adapted cohort size is set to 700.

4. EXPERIMENTAL RESULTS

Table 2 shows the speaker verification results. We first train three MFA-Conformer speaker embedding networks with different model sizes without ASR pretraining. The three models follow the network architecture of ASR Conformers as described in section 3.2. From the table, we can see that the speaker verification performance of MFA-Conformer does not improve with more trainable parameters. This indicates that Conformer can be easily overfitted to the training data without large-scale dataset.

[1] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_conformer_ctc_small

[2] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_conformer_ctc_medium

[3] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/stt_en_conformer_ctc_large

Table 2: Speaker verification results on VoxCeleb 1. *The results of WavLM use quality-aware score calibration, while other reported results in the table do not use this method. Comparisons between WavLM and other systems may not be fair.

Model	**Size**	**Pretrained**	**VoxCeleb 1-O**		**VoxCeleb 1-E**		**VoxCeleb 1-H**	
			EER[%]	minDCF	EER[%]	minDCF	EER[%]	minDCF
ECAPA-TDNN [11]	46.6M	×	0.68	0.0753	0.91	0.1006	1.72	0.1695
HuBERT Large [25]	316.61M+	√	0.72	-	0.70	-	1.32	-
Wav2Vec2.0 Large (XLSR) [25]	317.38M+	√	0.73	-	0.68	-	1.23	-
UniSpeech-SAT Large [25]	316.61M+	√	0.63	-	0.63	-	1.29	-
WavLM Large* [26]	316.62M+	√	0.38	-	0.48	-	0.99	-
NEMO small	15.88M	×	**0.88**	**0.1367**	**1.08**	**0.1342**	**2.20**	**0.2245**
NEMO medium	35.26M	×	0.94	0.1200	1.26	0.1487	2.41	0.2398
NEMO large	130.94M	×	0.96	0.1375	1.22	0.1391	2.35	0.2278
NEMO large first 4 layers	35.02M	×	0.86	0.1051	1.03	0.1188	1.97	0.1920
NEMO large first 6 layers	48.72M	×	0.80	0.1101	1.04	0.1202	2.04	0.2012
NEMO large first 8 layers	62.42M	×	0.81	0.1121	1.00	0.1183	1.93	0.1904
NEMO small	15.88M	√	0.74	0.1101	0.90	0.1054	1.90	0.1893
NEMO medium	35.26M	√	0.61	0.0946	0.78	0.0891	1.67	0.1649
NEMO large	130.94M	√	**0.48**	**0.0673**	**0.71**	**0.0785**	**1.54**	**0.1538**
NEMO large first 4 layers	35.02M	√	0.77	0.1065	1.04	0.1159	1.95	0.1862
NEMO large first 6 layers	48.72M	√	0.58	0.0618	0.84	0.0937	1.62	0.1571
NEMO large first 8 layers	62.42M	√	0.64	0.0982	0.86	0.0944	1.77	0.1732

ASR pretrained Conformer is then used to initialize the MFA-Conformer speaker embedding network training. With ASR pretrained Conformer, the MFA-Conformer models significantly outperform their counterparts without pretraining. For the small model, the ASR pretrained MFA-Conformer obtains an EER of 0.74% on VoxCeleb 1-O trails, which is 15.9% relative lower than the small MFA-Conformer without pretraining. When the model size increases eight times to 130.94 million, the ASR pretrained MFA-Conformer obtains an EER of 0.48% on VoxCeleb 1-O trails, which is 50% relative lower than the large MFA-Conformer without pretraining.

Compared to larger self-supervised pretrained models with more than 300 million parameters (HuBERT Large, Wav2Vec2.0 Large, UniSpeech-SAT Large), the ASR pretrained MFA-Conformers achieve comparable or even better verification performance on VoxCeleb 1-O. The large ASR pretrained MFA-Conformer with 130.94 million parameters obtains an EER of 0.48%. In comparison, the UniSpeech-SAT large model with more than 316.62 million parameters achieves an EER of 0.63% on VoxCeleb 1-O trails. We also observe that the MFA-Conformer models can hardly outperform the self-supervised pretrained models in VoxCeleb 1-E and VoxCeleb 1-H trials. The reason may be that self-supervised models are pretrained with more data (56k - 188k hours) while MFA-Conformers are pretrained with fewer data (around 34k hours). Nevertheless, the MFA-Conformer speaker embedding network is more flexible as it can be easily trained from an ASR pretrained Conformer by simply adding an MFA module and a pooling layer.

We also extract the lower layers of the large ASR Conformer to train MFA-Conformer. Three Conformers with first 4, 6, or 8 layers from the large Conformer model are investigated. We can see that the smaller versions of the large Conformer model outperform the full large model without ASR pretraining. With ASR pretraining, the large Conformer outperforms its smaller versions. This reassures our statement that ASR pretraining can help prevent overfitting when training MFA-Conformer.

5. CONCLUSION

This paper proposes an MFA-Conformer framework initialized by ASR pretraining for speaker verification. Conformers trained from scratch with limited data may be easily overfitted, but ASR pretraining can prevent the MFA-Conformer from overfitting and improve the performance. Experimental results on Voxceleb data also show that the EERs are significantly reduced on Voxceleb1-O, Voxceleb1-E and Voxceleb1-H with ASR pretraining.

6. REFERENCES

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN Embeddings for Speaker

Recognition," in *ICASSP*, 2018, pp. 5329–5333.

[2] W. Cai, J. Chen, J. Zhang, and M. Li, "On-the-Fly Data Loader and Utterance-Level Aggregation for Speaker and Language Recognition," *IEEE/ACM TASLP*, vol. 28, pp. 1038–1051, 2020.

[3] W. Cai, J. Chen, and M. Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," in *Speaker Odyssey*, 2018, pp. 74–81.

[4] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive Statistics Pooling for Deep Speaker Embedding," in *Interspeech*, 2018, pp. 2252–2256.

[5] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and Res2Net Structures for Speaker Verification," in *SLT*, 2021, pp. 301–307.

[6] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Interspeech*, 2020, pp. 3830–3834.

[7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *CVPR*, 2019, pp. 4685–4694.

[8] X. Xiang, S. Wang, H. Huang, Y. Qian, and K. Yu, "Margin Matters: Towards More Discriminative Deep Neural Network Embeddings for Speaker Recognition," in *APSIPA*, 2019, pp. 1652–1656.

[9] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In Defence of Metric Learning for Speaker Recognition," in *Interspeech*, 2020, pp. 2977–2981.

[10] D. Garcia-Romero, G. Sell, and A. Mccree, "MagNetO: X-vector Magnitude Estimation Network plus Offset for Improved Speaker Recognition," in *Odyssey*, 2020, pp. 1–8.

[11] J. Thienpondt, B. Desplanques, and K. Demuynck, "The Idlab Voxsrc-20 Submission: Large Margin Fine-Tuning and Quality-Aware Score Calibration in DNN Based Speaker Verification," in *ICASSP*, 2021, pp. 5814–5818.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *CVPR*, 2016, pp. 770–778.

[13] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *CVPR*, 2018.

[14] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A New Multi-Scale Backbone Architecture," *IEEE TPAMI*, vol. 43, no. 2, pp. 652–662, 2021.

[15] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated Residual Transformations for Deep Neural Networks," in *CVPR*, 2017.

[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 *NeurIPS*, 2017.

[17] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Interspeech*, 2020, pp. 5036–5040.

[18] S. Chen, Y. Wu, Z. Chen, J. Wu, J. Li, T. Yoshioka, C. Wang, S. Liu, and M. Zhou, "Continuous Speech Separation with Conformer," in *ICASSP*, 2021, pp. 5749–5753.

[19] Y. Zhang, Z. Lv, H. Wu, S. Zhang, P. Hu, Z. Wu, H.-y. Lee, and H. Meng, "MFA-Conformer: Multi-scale Feature Aggregation Conformer for Automatic Speaker Verification," in *Interspeech*, 2022, pp. 306–310.

[20] T. Zhou, Y. Zhao, J. Li, Y. Gong, and J. Wu, "CNN with Phonetic Attention for Text-Independent Speaker Verification," in *ASRU*, 2019, pp. 718–725.

[21] M. Li, L. Liu, W. Cai, and W. Liu, "Generalized I-vector Representation with Phonetic Tokenizations and Tandem Features for both Text Independent and Text Dependent Speaker Verification," *Journal of Signal Processing Systems*, vol. 82, no. 2, pp. 207–215, 2016.

[22] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A Novel Scheme for Speaker Recognition using a Phonetically-Aware Deep Neural Network," in *ICASSP*, 2014, pp. 1695–1699.

[23] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring Wav2vec 2.0 on Speaker Verification and Language Identification," in *Interspeech*, 2021, pp. 1509–1513.

[24] N. Vaessen and D. A. van Leeuwen, "Fine-Tuning Wav2vec2 for Speaker Recognition," in *ICASSP*, 2022, pp. 7967–7971.

[25] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, S. Liu, Y. Qian, and M. Zeng, "Large-Scale Self-Supervised Speech Representation Learning for Automatic Speaker Verification," in *ICASSP*, 2022, pp. 6147–6151.

[26] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[27] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-XL: Attentive Language Models beyond a Fixed-Length Context," in *ACL*, 2019, pp. 2978–2988.

[28] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," *IEEE/ACM TASLP*, vol. 29, pp. 3451–3460, 2021.

[29] P. Xu, D. Kumar, W. Yang, W. Zi, K. Tang, C. Huang, J. C. K. Cheung, S. J. Prince, and Y. Cao, "Optimizing Deeper Transformers on Small Datasets," in *ACL IJCNLP*, 2021, pp. 2089–2102.

[30] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A Large-Scale Speaker Identification Dataset," in *Interspeech*, 2017, pp. 2616–2620.

[31] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep Speaker Recognition," in *Interspeech*, 2018, pp. 1086–1090.

[32] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.08484*, 2015.

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

[34] O. Kuchaiev, J. Li, H. Nguyen, O. Hrinchuk, R. Leary, B. Ginsburg, S. Kriman, S. Beliaev, V. Lavrukhin, J. Cook, et al., "Nemo: a toolkit for building ai applications using neural modules," *arXiv preprint arXiv:1909.09577*, 2019.

[35] P. Matjka, O. Novotn, O. Plchot, L. Burget, M. D. Snchez, and J. ernock, "Analysis of Score Normalization in Multilingual Speaker Recognition," in *Interspeech*, 2017, pp. 1567–1571.