ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9746050

LOCAL INFORMATION MODELING WITH SELF-ATTENTION FOR SPEAKER VERIFICATION

Bing Han, Zhengyang Chen, Yanmin Qian[†]

MoE Key Lab of Artificial Intelligence, AI Institute
X-LANCE Lab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China

ABSTRACT

Transformer based on self attention mechanism has demonstrated its state-of-the-art performance in most natural language processing (NLP) tasks, but it's not very competitive when applied for speaker verification in previous works. Generally, speaker identity is mostly reflected by the relationship between adjacent tokens, whose extraction mainly depends on local modeling ability. However, the self-attention module, as the key component of transformer, can help the model make full use of global information but insufficient to capture the local information. To tackle this limitation, in this paper, we strengthen the local information modeling from two different aspects: restricting the attention context to be local and introducing convolution operation into transformer. Experiments conducted on Voxceleb illustrate that our proposed methods can notably improve system performance, verifying the significance of local information for speaker verification task.

Index Terms— Speaker Verification, Local Information, Gaussian-attention, Local-attention, Convolution-attention

1. INTRODUCTION

Speaker verification (SV) is a task that utilizes the uttered speech to verify the speakers' identities. Given two utterances, a typical SV system can extract speaker embeddings and automatically determine whether two utterances belong to the same speaker or not. In general, a typical SV system includes two parts. The first one is an embedding extractor [1, 2, 3, 4, 5] which is used to extract the fixed-length speaker representation from variable-length utterances. The other one is back-end model [6, 7] which aims to calculate the similarity between speaker embedding vectors.

With the widely application of deep learning methods in other fields, the effectiveness of DNN has been broadly demonstrated. Based on this, different network structures have been proposed for speaker embedding extraction, including the time-delay neural network (TDNN) [2], ResNet [3, 8] and more powerful architectures such as Dual Path Network (DPN) [9, 4] and ECAPA-TDNN [5].

Because of the powerful global information modeling and parallel computation ability of transformer [10], it has become the most popular backbone in natural language processing (NLP) [11] and automatic speech recognition (ASR) field [12, 13]. Recently, transformer has shown its strong competitiveness in computer vision (CV) [14] field compared to the dominant convolutional neural network (CNN). However, researchers find that it is non-trivial to leverage the transformer architecture in speaker verification task to achieve competitive results [15, 16] with the ResNet [3] and ECAPA-TDNN [5] based system. The self-attention module in the transformer enables it to see the global information of the input sequence while the speaker information is often reflected in local rhythmic changes.

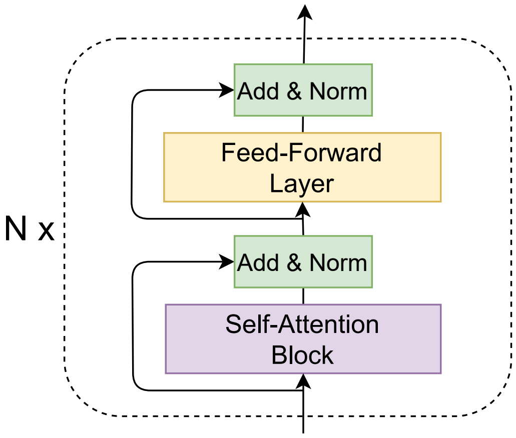


Fig. 1. Transformer Encoder

In order to emphasize the local information in transformer for speaker verification, in this paper, we introduce the local information to the original transformer from two different aspects. First, we restrict attention in tranformer to be local, including local self-attention and gaussian self-attention. Second, we combine the transfomer with the convolution operation which naturally models the local information. The experiments are conducted on Voxceleb [17], and the results illustrate that the proposed three methods significantly improve the system performance which demonstrates the importance of local information for speaker verification.

The rest of the paper is organized as below: In Section 2, we give a simple description of transformer encoder's architecture. Then, we present our methods of local information modeling with self-attention for speaker verification. Next, experimental results on Voxceleb [17] are presented and analyzed in Section 4 and 5. And finally, the conclusion is given in Section 6.

2. RELATED WORK

Transformer, which is proposed in [10], has been widely used in various fields and achieved the state-of-the-art performance [11, 12, 13, 14]. For speaker verification task, [15, 16] also adopted transformer as embedding extractor to encode the speaker characteristics into the

[†] Yanmin Qian is the corresponding author

discriminative embeddings. The transformer encoder is composed of a stack of N identical blocks with two sub-layers and the corresponding architecture is shown in Figure 1. The first sub-layer is a multi-head self-attention mechanism which is the key component of the transformer encoder. It helps the encoder look at other frames in the input sentence as it encodes a specific token. The second is a simple position-wise feed-forward network. It is composed of two full-connected-layer and is independently applied to each position. Besides, a residual connection [8] is employed around each of the two sub-layers, and it is followed by a layer normalization [18].

3. METHODS

The most important component in transformer is the self-attention module. The attention function can be described as mapping a set of *query* and *key-value* pairs to an output, where the *query, keys, values* and outputs are all vectors. The output is computed as a weighted sum of the *values*, where the weight assigned to each *value* is computed by a compatibility function of the *query* with the corresponding *key*. In this section, we firstly introduce the original global attention mechanism in transformer blocks, and then two kinds of modification focusing on local information will be described.

3.1. Self-Attention

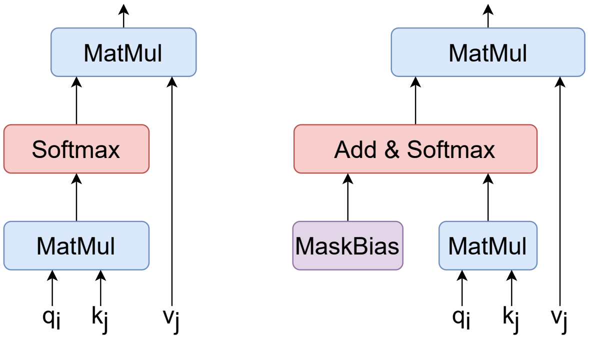


Fig. 2. Illustration of attentions: left is the original global attention of transformer, right is local restricted attention with bias.

Considering an input audio sequence $X = [x_1, x_2, \dots, x_T]$ of length T to the self-attention block, with $x_t \in \mathbb{R}^{d_m}$, and a set of trainable parameters $\{W_Q, W_K\} \in \mathbb{R}^{d_m \times d_k}$, $W_V \in \mathbb{R}^{d_m \times d_v}$. Then, the model transforms the input X into namely queries $Q \in \mathbb{R}^{T \times d_k}$, keys $K \in \mathbb{R}^{T \times d_k}$, and values $V \in \mathbb{R}^{T \times d_v}$, which are defined as follows:

$$\begin{aligned} q_i &= x_i W_Q \\ k_i &= x_i W_K \\ v_i &= x_i W_V \end{aligned} \tag{1}$$

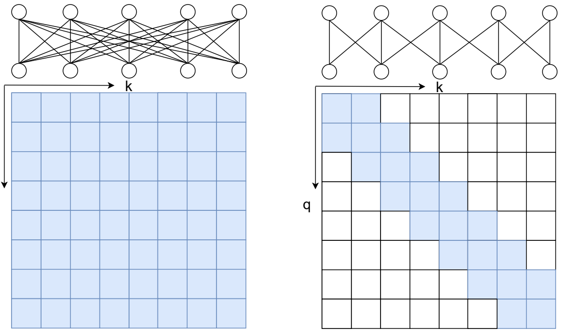
Then, the vanilla dot-product self-attention works as the left of Figure 2: It soft-aligns each input token $x_j \in X$ to the output o_i, according to the compatibility function computed by the softmax of dot products and then sums the attended values together. As a result, we can get the output o_i of time instance i:

$$\begin{aligned} o_i &= \sum_{j \in T} Softmax_j(q_i k_j) v_j \\ &= \sum_{j \in T} \frac{exp(q_i k_j)}{\sum_m exp(q_i k_m)} v_j \end{aligned} \tag{2}$$

Self-attention mechanism brings the ability to model global information to the model, which is very effective to solve the thorny long-term dependence problem in sequence problem but loses the ability to capture local features. In this section, we mainly consider helping the transformer explicitly model the local information from two different aspects. For the first aspect, we can constrain the context that each query can attend from the whole sequence to the adjacent area. For the other aspect, we can directly encode the local information to the query, key, and value. Next, we will introduce our proposed local attention mechanism in these two aspects.

3.2. Constraint Attention Context

3.2.1. Local Self-Attention



(a) Global

(b) Local

Fig. 3. The illustrations of global and local attention. The colored squares means corresponding attention scores are calculated, and a blank square means the attention score is discarded.

For the original definition of self-attention, it treats the similar frames at different positions almost equally and performs globally, which is shown in Figure 3 (a). It is inconsistent with our cognition that adjacent frames contribute more to speaker embedding extraction. While CNNs / RNNs model this 'chunking' phenomenon internally, the vanilla self-attention mechanism in Transformer could not capture the local structure of texts.

Since speech comes with a strong property of locality, it is natural to restrict each query to attend to its neighbor nodes. A widely adopted class of such pattern is local self-attention, in which the attention matrix is a band matrix as illustrated in the right part of Figure 3. Given a fixed window size $2w$, each frame only focuses on w frames on each side. To implement the local self-attention in transformer, we can generate a bias matrix and add it to the score matrix in order to mask the frames, which is shown as the right part of Figure 2.

$$\begin{aligned} o_i &= \sum_{j=i-w}^{i+w} Softmax_j(q_i k_j) v_j \\ &= \sum_{j \in T} Softmax_j(q_i k_j + b_{ij}) v_j \end{aligned} \tag{3}$$

where b_{ij} is a bias and defined as follows:

$$b_{ij} = \begin{cases} 0 & \text{if } |i-j| \le w \\ -\infty & \text{if } |i-j| > w \end{cases} \tag{4}$$

3.2.2. Gaussian Self-Attention

Local self-attention can directly constraint the attention context, but the fixed window size is not flexible. For better modeling the local information, we also propose another self-attention method based on Gausssian distribution to reduce the score weight continuously according to the distance between tokens. Compared with local self-attention using the hard weight (0 or 1) to restrict local information, Gaussian self-attention can be regarded as a soft version of local self-attention. We hypothesize that the contribution to the central frame from tokens at different distance obey a normal distribution, and then use a variant of Gaussian prior to correct the score weight of tokens which neighbor with the current central frame.

For simplicity, we assume that this weight satisfies the standard norm distribution whose mean and variance are 0 and $1/2\pi$. Then, its probability density function can be simplified to $\phi(d_{ij}) = exp(-\pi d_{ij}^2)$ where d_{ij} is the distance between frame i and j. To correct the weight of frames at various distances, we insert Gaussian prior $\phi(d)$ to Eq. 2:

$$\begin{aligned} o_i &= \sum_{j \in T} \frac{\phi(d_{ij}) exp(q_i k_j)}{\sum_m \phi(d_{im}) exp(q_i k_m)} v_j \\ &= \sum_{j \in T} \frac{exp(q_i k_j - \pi d_{ij})}{\sum_m exp(q_i k_m - \pi d_{im})} v_j \\ &= \sum_{j \in T} Softmax_j(-\pi d_{ij}^2 + q_i k_j) v_j \end{aligned} \tag{5}$$

Then, Eq. 5 converts the multiplication operation into the addition operation of the Gaussian bias term, which has the same form with the right of Figure 2. Because our previous assumptions about the Gaussian distribution are too strong. To loose the restriction, we introduce a learnable parameter w to adjust the shape of Gaussian distribution in the following:

$$o_i = \sum_{j \in T} Softmax_j(-w d_{ij}^2 + q_i k_j) v_j \tag{6}$$

Besides, inspired by [19, 20, 21], a punishment term b is applied to reduce the weight of the central word attending itself:

$$o_i = \sum_{j \in T} Softmax_j(-|w d_{ij}^2 + b| + q_i k_j) v_j \tag{7}$$

where $|\cdot|$ represents the absolute value with scalar parameters $w > 0$ and $b \le 0$.

3.3. Convolution Self-Attention

Convolutions have also been successfully applied for speaker verification task [2, 3], which capture local context progressively via a local receptive field layer by layer. In this paper, we also make an exploration about how to effectively combine convolutions with self-attention to enhance the ability of the model to capture the local information.

Conformer [22] is a state-of-the-art ASR encoder architecture, which inserts a convolution layer into a Transformer block to increase the local information modeling capability of the traditional Transformer model. In the first, we tried to adopt conformer as the embedding extractor to extract speaker embedding, but obtained unsatisfactory performance. Based on this, we propose two kinds of convolution-augmented transformer for speech verification, which are described in the following:

Conv-SAB: As mentioned in Equ. 1, Query, Key, and Value are obtained by transforming input **X** with learnt matrices W_q, W_k and W_v in self-attention block (SAB). To introduce convolution into SAB, we replace the matrices with three distinct convolution 1d layers, which is named Conv-SAB. With the help of convolution layers, local information can be introduced when calculating the attention.

Conv-FFN: Another idea is to introduce convolution between attention. Inspired by [23], we use a 2-layer convolution 1D network with ReLU activation to replace the original fully connected layers in Feed-Forward Network (FFN), which is described in Figure 1. Then, the conv-FFN is defined as follows:

$$\begin{aligned} ConvFFN(x) &= Conv(ReLU(Conv(x))) \\ ReLU(x) &= max(0, x) \end{aligned} \tag{8}$$

where $Conv$ is convolution 1d layer, $ReLU$ is the activation function and x is the input of Conv-FFN.

4. EXPERIMENT SETUP

4.1. Dataset

In our experiment, we trained all the systems on the development set of Voxceleb2 [17], which contains 1,092,009 utterances among 5,994 speakers. For the evaluation, the development set and test set of Voxceleb1 are used. We report the experimental results on 3 trial sets as defined in [17]: the original test set of Voxceleb 1 contains 37,720 trials from 40 speakers, the Voxceleb 1-E test set (using the entire dataset) contains 581,480 trials from 1251 speakers, and the Voxceleb 1-H test set (within the same nationality and gender) contains 552,536 trials from 1190 speakers.

4.2. Training Detail

To enrich the training data, we perform online data augmentation [24] with MUSAN dataset [25]. The noise type includes ambient noise, music, television, and babble noise for the background additive noise. Augmented data is generated by mixing noise with original speech. For the reverberation, the convolution operation is performed with 40,000 simulated room impulse responses (RIR) [26]. During the training, we decide whether to do augmentation for each sample with the probability 0.6.

We used 40 dimension Fbank with 25ms length Hamming windows and 10ms window shift as the input feature, while no voice activity detection (VAD) is involved. All the features are mean normalized with a sliding window of up to 3 seconds. The whole training process will last 165 epochs. Noam [10], with 25,000 warm-up steps, is applied as the optimizer to train the models on softmax loss function. After the model optimization, we use Probability Linear Discriminant Analysis (PLDA) [6] as back-end to score trials.

5. RESULTS

5.1. Constraint Attention Context

First, we will demonstrate the effectiveness to incorporate the local information by constraining the attention context. And the results are shown in Table 1. According to the table, we can see that the best L-SA (size=5) achieves an average relative $> 15.0\%$ improvement

Table 1. Results comparison of different systems on Voxceleb dataset. Equal error rate (EER) and the minimum detection cost function at $P_{target} = 0.01$ (MinDCF$_{0.01}$) are used as the performance evaluation metrics. For the baseline system with the original self-attention, we reproduce the model with setup describe in [16]. Attention dimension and head number are 512 and 8 respectively. All systems use 6 layers. SA (Self-Attention), L-SA (Local Self-Attention), G-SA (Gaussian Self-Attention), C-SA (Convolution Self-Attention)

Attention	**Configure**	Voxceleb-O		Voxceleb-E		Voxceleb-H	
		EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF
SA	[16]*	7.700	—	6.320	—	6.940	—
	—	2.915	0.3486	2.872	0.3289	4.754	0.4459
L-SA	size=2	2.574	0.2926	2.615	0.2966	4.490	0.4372
	size=5	2.287	**0.2573**	**2.447**	**0.2778**	**4.318**	**0.4135**
	size=8	**2.282**	0.2704	2.520	0.2840	4.394	0.4240
G-SA	—	**2.133**	**0.2519**	**2.281**	**0.2494**	**4.003**	**0.3851**
C-SA	Conformer [22]	3.946	0.3778	3.846	0.4044	6.526	0.5355
	Conv-SAB	2.122	0.2843	2.234	0.2670	3.949	0.3928
	Conv-FFN	**2.090**	**0.2534**	**2.131**	**0.2478**	**3.745**	**0.3741**
L-SA & C-SA	size=5	2.223	0.2777	2.336	0.2720	4.113	0.4075
G-SA & C-SA	Conv-FFN	**1.963**	**0.2654**	**2.071**	**0.2373**	**3.659**	**0.3687**

* results are cited from [16]

on both EER and MinDCF over the baseline system for each test set. It is observed that, performing with local information can obtain significant system improvements, and the proposed L-SA is better than the original self-attention for speaker verification. In addition, we also conduct an exploration about the influence of different attention sizes on performance. Based on this, the original self-attention with global mode can be regarded as a very large attention size. For the different attention sizes, the system can achieve the best performance position when the size is 5, and too small (size=2) or too large (size=8) are not appropriate for the local self-attention.

Thus, attention size has a significant impact on the models, but it is difficult to select the appropriate value in practical application. As a result, we propose Gaussian self-attention (G-SA) with learnable attention size, which can be regarded as a soft version of L-SA. According to the results presented in Table 1, it is obviously observed that the G-SA system outperforms all L-SA systems, which means that G-SA is more flexible than L-SA with fixed attention size.

5.2. Convolution Self-Attention

In this section, we also conducted an investigation on the effect of introducing convolution layers in different positions of transformer. And the results are also shown in Table 1. In order to illustrate the superiority of the methods proposed in this work, we also provide the results of the conformer [22], which is designed to combine transformer and convolution, and has been widely used in ASR. However, in terms of results, conformer performs poorly, even worse than the baseline. This also shows that the performance can only be improved by putting the convolution layer in the right place.

The proposed two modes with convolution self-attention (C-SA), i.e. Conv-SAB and Conv-FFN, can both obtain obvious improvement compared to the usual self-attention. Especially, the Conv-FFN method outperforms all the other self-attention systems shown in Table 1. Then, we can conclude that introducing convolution can help the network utilize local information, so as to improve the overall performance of the model.

5.3. Combination

It is worth noting that the L-SA and G-SA change the context that the query can see, whereas the C-SA change the way to calculate the elements for attention by involving the convolution operation. It will be intuitive and simple to combine L-SA or G-SA with C-SA. For simplicity, we only leverage the best system Conv-FFN for system combination. The corresponding results are shown at the bottom of table 1. From the results, we find that the L-SA is not compatible with C-SA and the system performance degrades. Encouragingly, the G-SA complements the C-SA well and achieved further improvement. This combined system leads to the best results among all the systems and obtains relative ∼25.0% reduction on both EER and MinDCF.

6. CONCLUSION

To better focus on local information in transformer, in this work, we propose three improved methods for vanilla self-attention, including L-SA, G-SA, and C-SA. The former two achieve the goal by restricting the size of attention and the latter one is to obtain the performance gain by combining convolution. In the experiments, the results show that these methods all can significantly improve the performance, which demonstrates the importance of the local information for transformer-based speaker verification. Among them, G-SA with dynamic attention size also shows better performance and flexibility, compared with L-SA. In addition, to further improve the system, we also integrate the proposed G-SA with Conv-FFN, and this system achieves the best performance and obtains relative ∼25.0% improvement on both EER and MinDCF over the traditional self-attention.

7. ACKNOWLEDGEMENTS

This work was supported by the China NSFC projects (No. 62122050 and No. 62071288), and Shanghai Municipal Science and Technology Major Project (No. 2021SHZDZX0102). Experiments have been carried out on the PI super-computer at Shanghai Jiao Tong University.

8. REFERENCES

[1] Yuan Liu, Yanmin Qian, Nanxin Chen, Tianfan Fu, Ya Zhang, and Kai Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[2] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *Proc. IEEE ICASSP*. IEEE, 2018, pp. 5329–5333.

[3] Hossein Zeinali, Shuai Wang, Anna Silnova, Pavel Matějka, and Oldřich Plchot, "But system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.

[4] Xu Xiang, "The xx205 system for the voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2011.00200*, 2020.

[5] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[6] Sergey Ioffe, "Probabilistic linear discriminant analysis," in *European Conference on Computer Vision*. Springer, 2006, pp. 531–542.

[7] Shreyas Ramoji, Prashant Krishnan, and Sriram Ganapathy, "NPLDA: A Deep Neural PLDA Model for Speaker Verification," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 202–209.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[9] Bing Han, Zhengyang Chen, Zhikai Zhou, and Yanmin Qian, "The sjtu system for short-duration speaker verification challenge 2021," *Proc. Interspeech 2021*, pp. 2332–2336, 2021.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[12] Xun Gong, Yizhou Lu, Zhikai Zhou, and Yanmin Qian, "Layer-wise fast adaptation for end-to-end multi-accent speech recognition," in *Proc. ISCA Interspeech*, 2021, pp. 1274–1278.

[13] Wei Wang, Zhikai Zhou, Yizhou Lu, Hongji Wang, Chenpeng Du, and Yanmin Qian, "Towards data selection on tts data for children's speech recognition," in *Proc. IEEE ICASSP*. IEEE, 2021, pp. 6888–6892.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[15] Sandesh V Katta, S Umesh, et al., "S-vectors: Speaker embeddings based on transformer's encoder for text-independent speaker verification," *arXiv preprint arXiv:2008.04659*, 2020.

[16] Pooyan Safari, Miquel India, and Javier Hernando, "Self-attention encoding and pooling for speaker recognition," in *Proc. ISCA Interspeech*, 2020, pp. 941–945.

[17] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. ISCA Interspeech*, 2018, pp. 1086–1090.

[18] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[19] Maosheng Guo, Yu Zhang, and Ting Liu, "Gaussian transformer: a lightweight approach for natural language inference," in *Proc. AAAI*, 2019, vol. 33, pp. 6489–6496.

[20] Jaeyoung Kim, Mostafa El-Khamy, and Jungwon Lee, "T-gsa: Transformer with gaussian-weighted self-attention for speech enhancement," in *Proc. IEEE ICASSP*. IEEE, 2020, pp. 6649–6653.

[21] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang, "Disan: Directional self-attention network for rnn/cnn-free language understanding," in *Proc. AAAI*, 2018, vol. 32.

[22] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al., "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[23] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, "Fastspeech: Fast, robust and controllable text to speech," *arXiv preprint arXiv:1905.09263*, 2019.

[24] Weicheng Cai, Jinkun Chen, Jun Zhang, and Ming Li, "On-the-fly data loader and utterance-level aggregation for speaker and language recognition," *IEEE/ACM Trans. ASLP.*, vol. 28, pp. 1038–1051, 2020.

[25] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[26] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. IEEE ICASSP*. IEEE, 2017, pp. 5220–5224.