INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Transformer-based end-to-end speech recognition with residual Gaussian-based self-attention

Chengdong Liang, Menglong Xu, Xiao-Lei Zhang

CIAIC, School of Marine Science and Technology, Northwestern Polytechnical University, China

{liangchengdong, mlxu}@mail.nwpu.edu.cn, xiaolei.zhang@nwpu.edu.cn

Abstract

Self-attention (SA), which encodes vector sequences according to their pairwise similarity, is widely used in speech recognition due to its strong context modeling ability. However, when applied to long sequence data, its accuracy is reduced. This is caused by the fact that its weighted average operator may lead to the dispersion of the attention distribution, which results in the relationship between adjacent signals ignored. To address this issue, in this paper, we introduce relative-position-awareness self-attention (RPSA). It not only maintains the global-range dependency modeling ability of self-attention, but also improves the localness modeling ability. Because the local window length of the original RPSA is fixed and sensitive to different test data, here we propose Gaussian-based self-attention (GSA) whose window length is learnable and adaptive to the test data automatically. We further generalize GSA to a new residual Gaussian self-attention (resGSA) for the performance improvement. We apply RPSA, GSA, and resGSA to Transformer-based speech recognition respectively. Experimental results on the AISHELL-1 Mandarin speech recognition corpus demonstrate the effectiveness of the proposed methods. For example, the resGSA-Transformer achieves a character error rate (CER) of 5.86% on the test set, which is relative 7.8% lower than that of the SA-Transformer. Although the performance of the proposed resGSA-Transformer is only slightly better than that of the RPSA-Transformer, it does not have to tune the window length manually.

Index Terms: speech recognition, end-to-end, gaussian-based self-attention, transformer

1. Introduction

Recently, automatic speech recognition (ASR) based on dot-product *self-attention* (SA) [1] has been widely studied. SA has a simple mathematical structure. It not only is able to do global dependency modeling for an input sequence, but also supports parallel computing naturally. Due to such advantages, it has been applied successfully to hybrid-model-based ASR [2,3] and Transformer-based ASR [4–7]. However, its weighted average calculation may lead to the dispersion of attention distribution, which results in insufficient calculation on the dependency of neighboring signals, denoted as the insufficient *localness modeling* problem. This problem becomes apparent in long utterances because of the excessive flexibility of the SA in context modeling. In addition, the computational complexity of SA is about the square of the length of its input.

Recently, relative positional embedding [8, 9] uses relative embedding to solve the insufficient localness modeling problem for the embedding layers of the Transformer-based ASR. However, the relative embedding itself does not limit the attention to the neighborhood of the frame. This still allows SA to attend to the distant frames that are not important for ASR. To remedy this weakness, Xu *et al.* [10] proposed to add a local dense synthesizer attention together with self-attention, where the local dense synthesizer attention focuses on the localness modeling. However, the window length of the method is a hyperparameter. Masking [11] limits the range of SA by a soft Gaussian window. Contrast to the relative positional embedding, masking directly limits the range of the attention structurally. Compared to [10], its advantage is that its window length is automatically determined in the training stage. However, the window length is fixed in the test stage again, which may not be the best choice since that different frames or characters may have different dependencies on neighbor frames. Moreover, choosing a suitable window length for each self-attention layer is time consuming, and case by case for different test data.

To solve the insufficient localness modeling problem in a more flexible way, in this paper, we propose an alternative Transformer-based ASR based on relative-position-awareness self-attention (RPSA), Gaussian-based self-attention (GSA), and residual Gaussian-based self-attention (resGSA). Specifically, RPSA, which was originally proposed for machine translation [12], adds a window to limit the range of SA which enforces SA to concentrate on learning the local connection between frames. However, its window length is fixed, which may not be the best choice since that different frames may have different dependency on their neighboring frames. Moreover, choosing a suitable window length is time consuming. To solve this problem, we further apply GSA [13] to the Transformer-based ASR. The window of GSA is dynamic and adaptive to frames in the test stage, i.e., different test frames may have different window lengths and weights. Motivated by [14], we further propose resGSA to improve the performance, where each resGSA layer takes the attention score from the previous layer as an additive residual score of GSA.

Experiment results on the AISHELL-1 Mandarin dataset show that the proposed resGSA significantly improves the performance without increasing the computational complexity. Specifically, the resGSA-Transformer achieves a relative character error rate (CER) reduction of 7.8% over the SA-Transformer with roughly the same number of parameters and computational complexity as the latter. Among the proposed methods, although the resGSA-Transformer achieves a slightly lower CER than the RPSA-Transformer, the resGSA-Transformer is more flexible than the latter. It does not need to take extra time to find an appropriate window length for each self-attention.

The most related work of resGSA for ASR is [11], in which a soft Gaussian mask window is used to address the *localness modeling* problem. However, the window length, which is fixed during the testing time, is independent of the input sequences. Different from [11], we use a feed-forward neural network to learn the mean and variance of a Gaussian function, which makes resGSA more flexible than the method in [11].

http://dx.doi.org/10.21437/Interspeech.2021-427

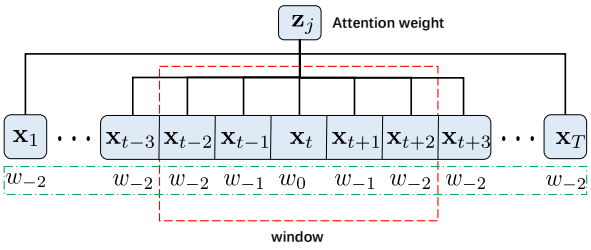

Figure 1: *Example of relative-position-awareness with* $m = 2$.

2. Background

In this section, we first introduce the scaled dot-product self-attention, which causes the insufficient localness modeling problem. Then, we introduce the masking self-attention, which is a solution closely related to our proposed methods.

2.1. Scaled dot-product self-attention

The attention layer in Transformer adopts dot-product attention which has the following form:

$$\mathbf{H} = \mathrm{Attn}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_{\mathrm{k}}}}\right)\mathbf{V} \quad (1)$$

where $\mathrm{Attn}(\cdot)$ is the attention function, and $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ are formulated as:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}^{\mathrm{Q}}, \mathbf{K} = \mathbf{X}\mathbf{W}^{\mathrm{K}}, \mathbf{V} = \mathbf{X}\mathbf{W}^{\mathrm{V}} \quad (2)$$

where $\mathbf{W}^{\mathrm{Q}}$, $\mathbf{W}^{\mathrm{K}}$, $\mathbf{W}^{\mathrm{V}} \in \mathbb{R}^{d\times d_k}$ are learnable projection matrices with $d_k = d/h$ as the individual dimension of each attention head, and $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots, \mathbf{x}_T]$ is an input sequence with T as the length of the sequence and $\mathbf{x}_t$ as a d-dimensional acoustic feature. It is known that d is also the size of the SA layer.

The SA layer in Transformer usually uses multi-head attention to perform parallel attention. It calculates the scaled dot-product attention h times, and then projects the concatenation of all h outputs for the final attention values. The multi-head self-attention is formulated as:

$$\mathrm{MHA}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \mathrm{Concat}\left[\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_h\right]\mathbf{W}^{\mathrm{O}} \quad (3)$$

where $\mathbf{W}^{\mathrm{O}} \in \mathbb{R}^{d\times d}$ is the weight matrix of the linear projection layer, and $\mathbf{H}_i$ is the output of the i-th attention head.

2.2. Masking self-attention

Sperber *et al.* used a soft mask to mask the scaled dot-product self-attention [11].

$$\mathrm{Attn}^{\mathrm{masking}}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_{\mathrm{k}}}} + \mathbf{M}\right)\mathbf{V} \quad (4)$$

where $\mathbf{M}$ is the soft mask defined as $M_{i,j} = \frac{-(i-j)^2}{2\sigma^2}$ with σ as a trainable parameter controlling the window size.

3. Proposed algorithms

In this section, we present RPSA, GSA and resGSA respectively.

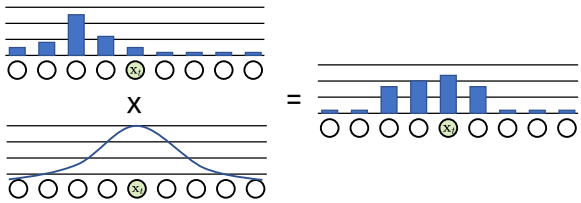

Figure 2: *Effect of the Gaussian bias on local features.*

3.1. Relative-position-awareness self-attention

Self-attention has strong global-dependency modeling ability which could build a connection to any two samples. This flexibility is an advantage in some long dependency tasks such as machine translation. However, for speech recognition, local information is more important than global characteristics for representing phonetic features, especially in long sequences. To enhance the localness modeling ability, we propose to replace SA by RPSA. RPSA adds neighboring edge connections between two speech frames $\mathbf{x}_t$ and $\mathbf{x}_j$ that are close to each other, which modifies (1) to the following localness-aware model:

$$\mathrm{Attn}^{\mathrm{RPSA}}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \mathrm{Softmax}\left(\frac{\mathbf{Q}(\mathbf{K}+\mathbf{A})^{\mathrm{T}}}{\sqrt{d_{\mathrm{k}}}}\right)\mathbf{V} \quad (5)$$

where $\mathbf{A}$ is the edge representation for matrix $\mathbf{K}$. Let a_{tj} denote the element of matrix $\mathbf{A}$, which strengthens the relative contribution of the acoustic feature $\mathbf{x}_t$ to the j-th attention weight at time t. It is calculated as follows:

$$a_{tj} = \begin{cases} w_{-m}, & j - t \le -m \\ w_{j-t}, & -m < j - t < m \\ w_m, & j - t \ge m \end{cases} \quad (6)$$

where m is the maximum length of the relative distance. Figure 1 shows an example of the relative-position-awareness with $m = 2$. From the figure, we see that, when $j \le t - 2$ or $j \ge t + 2$, the representation a_{tj} is fixed as w_{-2} and w_2 respectively.

3.2. Gaussian-based self-attention

RPSA learns a fixed representation weight for the localness modeling, and the length of its window is fixed as well. Therefore, it is time consuming to choose a suitable window length for each RPSA layer in practice. In order to achieve dynamic local enhancement, GSA uses a Gaussian distribution as an additive bias:

$$\mathrm{Attn}^{\mathrm{GSA}}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}} + \mathbf{G}\right)\mathbf{V} \quad (7)$$

where $\mathbf{G} \in \mathbb{R}^{d\times d}$ is the Gaussian bias matrix.

In this algorithm, the window size is predicted by each input sequence. Compared with RPSA, Gaussian distribution naturally focuses more attention on the closer position, as shown Figure 2.

3.2.1. Learning algorithm of the Gaussian matrix

Each bias element G_{tj} means the relation between current query $\mathbf{x}_t$ and position j:

$$G_{tj} = -\frac{(j - P_t)^2}{2\sigma_t^2} \quad (8)$$

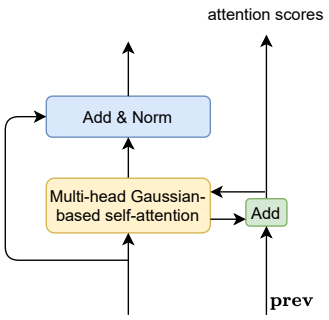

Figure 3: *The resGSA layer.*

where P_t is the central position of $\mathbf{x}_t$, σ_t is the standard deviation and $G_{tj} \in (-\infty, 0]$. The mean and deviation of G_{tj}, which decide the curve of the distribution, are related to $\mathbf{x}_t$. Adding such bias before softmax approximates to multiplying $(0, 1]$ after softmax layer. The key problem is how to choose right P_t and σ_t. An intuitive choice for P_t is to set $P_t = t$, since that the t-th attention weight is highly related to the input $\mathbf{x}_t$. We predict the central position P_t from $\mathbf{x}_t$:

$$P_t = T \cdot \text{sigmoid}\left(\mathbf{v}_p^\top \tanh\left(\mathbf{W}_p \mathbf{x}_t\right)\right) \tag{9}$$

where $\mathbf{v}_p$ and $\mathbf{W}_p$ denote learnable parameters of the feed-forward neural network (FNN), and T is the sequence length. Because the output of the sigmoid activation is constrained to $(0, 1)$, then, $P_t \in (0, T)$, and the final predicted position is $\lceil P_i \rceil$. Similar to (9), D_t can be predicted by:

$$D_t = T \cdot \text{sigmoid}\left(\mathbf{v}_d^\top \tanh\left(\mathbf{W}_d \mathbf{x}_t\right)\right) \tag{10}$$

We set $\sigma_t = \frac{D_t}{2}$ which determines the steepness of the curve. The larger the σ_t, the smoother the distribution. When both P_t and σ_t are fixed, the Gaussian deviation is a special case of the relative-position-awareness method, in which the weights of the window follow a Gaussian distribution.

3.3. Residual Gaussian-based self-attention

As shown in Figure 3, each resGSA layer takes the raw attention scores of all attention heads from the previous layer as additive residual scores of the current attention function. The sum of the two scores is then used to compute attention weights via softmax:

$$\begin{aligned} &\text{Attn}^{\text{resGSA}}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{prev}) = \\ &\text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}} + \mathbf{G} + \mathbf{prev}\right)\mathbf{V} \end{aligned} \tag{11}$$

where $\mathbf{prev}$ is the attention scores from previous layer. Finally, new attention scores $\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}} + \mathbf{G} + \mathbf{prev}$ are sent to the next layer. ResGSA converges well as GSA.

4. Experiments

4.1. Model architecture

As shown in Figure 4, the proposed resGSA-Transformer is an improved Speech-Transformer [15], which contains an encoder and a decoder. The encoder is composed of a stack of $N = 12$ encoder sub-blocks, each of which contains a resGSA layer and

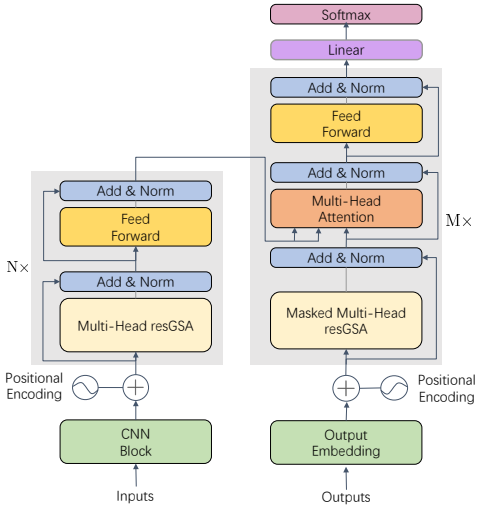

Figure 4: *Model architecture of the resGSA-Transformer.*

a position-wise feed-forward layer. For the convolution block, we stack two 3×3 convolution layers with stride 2 in both time dimension and frequency dimension to downsample the input features. The decoder is composed of a stack of $M = 6$ identical decoder sub-blocks and an embedding layer. In addition to the position-wise feed-forward layer, the decoder sub-block contains a resGSA layer and a multi-head attention layer. The former is used to receive the embedded label sequence and the latter is used to receive the output of the encoder. We add mask to the resGSA in the decoder sub-block to ensure that the predictions for the current position only depend on the previous positions. The output dimensions of resGSA and the feed-forward layers are both 256. The output of each layer in the sub-blocks has a residual connection to the input of the layer, followed by a layer normalization operator. The number of the attention heads in each attention layer is 4.

The structures of the proposed RPSA-Transformer and GSA-Transformer are similar to that of the resGSA-Transformer. They replace the resGSA layers in the encoder with RPSA and GSA respectively. Their decoder sub-blocks use the SA layer. The number of heads of RPSA and GSA is 4. The other layers of the encoder are the same as the resGSA-Transformer.

To demonstrate the effectiveness of the proposed methods in an apple-to-apple comparison, we also constructed a standard dot-product self-attention based Transformer (SA-Transformer) and a masking self-attention based Transformer (masking-Transformer). Their model structures are similar to the GSA-Transformer except replacing the GSA layers in the encoder with scaled dot-product self-attention and masking self-attention respectively.

4.2. Experimental setup

We evaluated the proposed models on a publicly-available Mandarin speech corpus AISHELL-1 [16], which contains over 170 hours of Mandarin speech data recorded from 400 speakers. We used the offical partitioning of the corpus, with 120, 098 utterances from 340 speakers for training, 14, 326 utterances from 40 speakers for validation and 7, 176 utterances from 20 speakers for testing. For each speaker, around 360 utterances are released. For all experiments, we used 80-dimensional Mel fil-

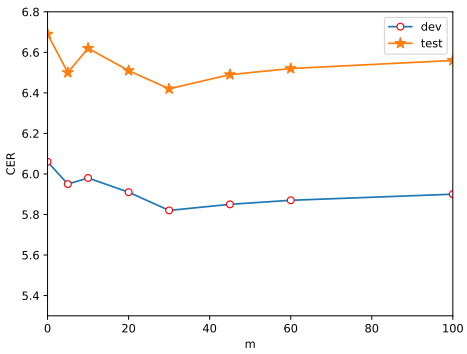

Figure 5: *CER (no LM) with different m.*

ter banks coefficients (Fbank) features as input, and the frame length and shift was set to 25 ms and 10 ms respectively. We use a vocabulary set of 4230 Mandarin characters and 4 non-language symbols ("unk", "eos", "pad" and "space"), which denote unknown characters, the start or end of a sentence, padding character and blank character respectively.

For the model training, we used Adam with Noam learning rate schedule (25000 warm steps) [1] as the optimizer. We used SpecAugment [17] for data augmentation. The attention dropout probalility is 0.1. For the language model, we used recurrent neural network (RNN) language model, which consisted of 2 RNN layers with 1024 units. We integrated language model into beam search by shallow fusion [18]. And the weight of the language model was set to 0.2 for all experiments. After 50 epochs training, the parameters of the last 10 epochs were averaged as the final model. In the decoding stage, we used a beam search with a width of 5 for all models. We use Pytorch [19] for modeling and Kaldi [20] for data preparation.

We compared with TDNN-Chain [21], listen-attend-spell (LAS) model [22], Self-attention Transducer (SA-Transducer) [23], Speech-Transformer [24] and hybrid-attention Transformer (HA-Transformer) [10]. TDNN-Chain uses the time-delay neural networks (TDNN) as the acoustic model. LAS is an attention-based model, which contains a pyramidal BLSTM encoder and an attention-based decoder. Self-attention Transducer contains a SA-based encoder and a SA-based prediction network. The structure of Speech-Transformer is the same as the SA-Transformer. The HA-Transformer replaces the SA layers in the encoder of the SA-Transformer with hybird-attention, which combines the local dense synthesizer attention and SA.

4.3. Results

We first investigated the effect of the window length m of the RPSA-Transformer on AISHELL-1. Figure 5 shows the CER curves of the model with respect to m. From the figure, we see that the CER first decreases, and then becomes stable with the increase of m. Based on the above finding, we set m to 30 in all of the following comparisons.

Then, we conducted an apple-to-apple comparison between the algorithms mentioned in Sections 2 and 3. From Table 1, we see that the proposed models outperform the baseline models. Specifically, the RPSA-Transformer achieves a lower CER than the baseline models, which demonstrates the effectiveness of the localness-aware modeling for ASR. The GSA-Transformer outperforms the RPSA-Transformer, which shows that the dynamic window leads to a better localness-aware model than the fixed window. Finally, the resGSA-Transformer achieves the best performance among all comparison methods. It achieves a relative CER reduction of 7.8% over the SA-Transformer and 6.8% over the masking-Transformer. Although the resGSA-Transformer is slightly better than the RPSA-Transformer, it is more flexible than the latter and does not need to take extra time to find the suitable window length.

Table 1: *CER comparison between the representative ASR systems.*

Model	Dev	Test
TDNN-Chain (kaldi) [21]	-	7.45
LAS [22]	-	10.56
Self-attention Transducer [23]	8.30	9.30
Speech-Transformer [24]	6.57	7.37
HA-Transformer [10]	5.66	6.18
SA-Transformer (baseline)	5.81	6.36
Masking-Transformer (baseline)	5.67	6.29
RPSA-Transformer (m=30) (proposed)	5.53	6.12
GSA-Transformer (proposed)	5.41	5.94
resGSA-Transformer (proposed)	5.38	5.86

To further verify the effectiveness of the proposed models, we compare them with several representative ASR systems which are the TDNN-chain, LAS, self-attention Transducer, speech-Transformer, and HA-Transformer, respectively. From the comparison results in Table 1, we see that the proposed models are superior to the five comparison systems. For example, the performance of the resGSA-Transformer, which achieves a CER of 5.86%, outperforms the strongest reference method, i.e. the HA-Transformer.

5. Conclusions

In this paper, we have proposed three attention schemes for the encoder of the Transformer-based speech recognition, which are the RPSA, GSA, and resGSA respectively. Specifically, RPSA was proposed to replace the common SA for remedying the inefficient localness-aware modeling problem of the SA-Transformer. To overcome the weakness of RPSA on the window length selection problem, we further proposed GSA and an improved version resGSA, whose window lengths are learnable and highly related to the input sequences. GSA and resGSA can achieve dynamic localness modeling, i.e., different test frames may be assigned with different window lengths and weights. Experimental results on AISHELL-1 show that the GSA- and resGSA-Transformer achieve better performance than RPSA-Transformer, and do not have to tune the window length; the proposed models are significantly better than the representative ASR systems.

6. Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2018AAA0102200, in part by National Science Foundation of China under Grant No. 61761146001, 61831019, 61671381, in part by the Project of the Science, Technology, and Innovation Commission of Shenzhen Municipality under grant No. JCYJ20170815161820095, and in part by the Open Research Project of the State Key Laboratory of Media Convergence and Communication, Communication University of China, China under Grant No. SKLMCC2020KF009.

7. References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017. 1, 4.2

[2] Y. Wang, A. Mohamed, D. Le, C. Liu, A. Xiao, J. Mahadeokar, H. Huang, A. Tjandra, X. Zhang, F. Zhang *et al.*, "Transformer-based acoustic modeling for hybrid speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6874–6878. 1

[3] D. Povey, H. Hadian, P. Ghahremani, K. Li, and S. Khudanpur, "A time-restricted self-attention layer for asr," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5874–5878. 1

[4] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang *et al.*, "A comparative study on transformer vs rnn in speech applications," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 449–456. 1

[5] A. Mohamed, D. Okhonko, and L. Zettlemoyer, "Transformers with convolutional context for asr," *arXiv preprint arXiv:1904.11660*, 2019. 1

[6] A. Zeyer, P. Bahar, K. Irie, R. Schlüter, and H. Ney, "A comparison of transformer and lstm encoder decoder models for asr," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 8–15. 1

[7] X. Chang, W. Zhang, Y. Qian, J. Le Roux, and S. Watanabe, "End-to-end multi-speaker speech recognition with transformer," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6134–6138. 1

[8] P. Zhou, R. Fan, W. Chen, and J. Jia, "Improving generalization of transformer for speech recognition with parallel schedule sampling and relative positional embedding," *arXiv preprint arXiv:1911.00203*, 2019. 1

[9] N.-Q. Pham, T.-L. Ha, T.-N. Nguyen, T.-S. Nguyen, E. Salesky, S. Stueker, J. Niehues, and A. Waibel, "Relative positional encoding for speech recognition and direct translation," *arXiv preprint arXiv:2005.09940*, 2020. 1

[10] M. Xu, S. Li, and X.-L. Zhang, "Transformer-based end-to-end speech recognition with local dense synthesizer attention," *arXiv preprint arXiv:2010.12155*, 2020. 1, 4.2, 1

[11] M. Sperber, J. Niehues, G. Neubig, S. Stüker, and A. Waibel, "Self-attentional acoustic models," *arXiv preprint arXiv:1803.09519*, 2018. 1, 2.2

[12] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *arXiv preprint arXiv:1803.02155*, 2018. 1

[13] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015. 1

[14] R. He, A. Ravula, B. Kanagal, and J. Ainslie, "Realformer: Transformer likes residual attention," *arXiv e-prints*, pp. arXiv–2012, 2020. 1

[15] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888. 4.1

[16] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5. 4.2

[17] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019. 4.2

[18] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5828. 4.2

[19] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017. 4.2

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011. 4.2

[21] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi." in *Interspeech*, 2016, pp. 2751–2755. 4.2, 1

[22] C. Shan, C. Weng, G. Wang, D. Su, M. Luo, D. Yu, and L. Xie, "Component fusion: Learning replaceable language model component for end-to-end speech recognition system," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5361–5635. 4.2, 1

[23] Z. Tian, J. Yi, J. Tao, Y. Bai, and Z. Wen, "Self-attention transducers for end-to-end speech recognition," *arXiv preprint arXiv:1909.13037*, 2019. 4.2, 1

[24] Z. Tian, J. Yi, J. Tao, Y. Bai, S. Zhang, and Z. Wen, "Spike-triggered non-autoregressive transformer for end-to-end speech recognition," *arXiv preprint arXiv:2005.07903*, 2020. 4.2, 1