INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Transformer with Bidirectional Decoder for Speech Recognition

Xi Chen[1], *Songyang Zhang*[2], *Dandan Song*[3], *Peng Ouyang*[3], *Shouyi Yin*[1,4]

[1]Institute of Microelectronics, Beijing Innovation Center for Future Chip, Tsinghua University
[2]ShanghaiTech University
[3]Tsingmicro Intelligent Technology Co., Limited
[4]Beijing National Research Center For Information Science And Technology

yinsy@tsinghua.edu.cn

Abstract

Attention-based models have made tremendous progress on end-to-end automatic speech recognition(ASR) recently. However, the conventional transformer-based approaches usually generate the sequence results token by token from left to right, leaving the right-to-left contexts unexploited. In this work, we introduce a bidirectional speech transformer to utilize the different directional contexts simultaneously. Specifically, the outputs of our proposed transformer include a left-to-right target, and a right-to-left target. In inference stage, we use the introduced bidirectional beam search method, which can not only generate left-to-right candidates but also generate right-to-left candidates, and determine the best hypothesis by the score.

To demonstrate our proposed speech transformer with a bidirectional decoder(STBD), we conduct extensive experiments on the AISHELL-1 dataset. The results of experiments show that STBD achieves a 3.6% relative CER reduction(CERR) over the unidirectional speech transformer baseline. Besides, the strongest model in this paper called STBD-Big can achieve 6.64% CER on the test set, without language model rescoring and any extra data augmentation strategies.[1]

Index Terms: Speech Transformer, End-to-end Speech Recognition, Bidirectional Decoder

1. Introduction

End-to-end models for automatic speech recognition (ASR) have attracted much attention recently because of their concise pipeline, which integrate *acoustic model*, *pronunciation* and *language model* into a single model. Comparing with conventional hybrid system, the various components can be optimized jointly in such an end-to-end framework. The most popular end-to-end speech recognition approaches typically include *connectionist temporal classification (CTC)* [1, 2, 3, 4], *recurrent neural network transducer (RNN-T)* [5, 6, 7, 8], and *attention-based encoder-decoder architecture* [9, 10, 11, 12, 13].

The self-attention based methods (also called *transformer*) [14], have demonstrated promising results in various natural language processing (NLP) tasks recently. The transformer model exploits the short/long range context by connecting arbitrary pairs of position in the input sequence directly. Further more, the model can be trained in a parallel way, which is much more efficient than conventional recurrent neural networks.

There have been several attempts to build an end-to-end system for speech recognition tasks based on the transformer model. Specifically, [12] first introduces the transformer model for the WSJ task. [15] explores different model units and finds the character-based model works best on the Mandarin ASR task. [16, 17] demonstrates the transformer for streaming ASR. More recent efforts aim to enhance the transformer model on ASR by integrating connectionist temporal classification (CTC) for training and decoding [18] or using unsupervised pre-training strategy [19].

However, these speech transformer models are typically trained in a left-to-right style, which means the decoding model predicts token depending on its left generated outputs, leaving the right-to-left contexts unexploited. Moreover, as the attention-based encoder-decoer network is an autoregressively generative model, they usually suffer from the issue of exposure bias in the sequence-to-sequence tasks.

Inspired by recent works [20, 21], which introduce an interactive decoding model to exploit the bidirectional context for neural machine translation(NMT) task, we propose to utilize the right-to-left context and bidirectional targets on speech recognition tasks to tackle the aforementioned issues. To this end, we propose a speech transformer model with bidirectional decoder structure. The STBD consists of a shared encoder and a directional decoder which includes two different unidirectional decoders. The bidirectional model predicts the next left-to-right token and right-to-left token simultaneously, separately depending on the history contexts predicted by left-to-right and right-to-left decoding. Therefore, the outputs of our proposed transformer include a left-to-right target, and a right-to-left target. In inference stage, the bidirectional decoder generate both left-to-right and right-to-left targets with bidirectional beam search method. Finally, we keep the hypothesis with the highest score of different directional candidates. The extensive experiments and ablation study are conducted on the AISHELL-1 dataset [22].

The main contributions of this work can be summarized as follows:

- We propose a novel speech transformer with bidirectional decoder to exploit the bidirectional context information for ASR task.
- We investigate the effectiveness of the bidirectional target and bidirectional decoding with comprehensive experiments.
- Our method achieves a 3.6% relative CER reduction. The best model in this paper described as STBD-big achieves 6.64% CER with a large margin improvement on AISHELL-1 dataset.

2. Method

In this work, we introduce a novel speech transformer with bidirectional decoder for the ASR task. Below we start with a brief

[1]This work was supported in part by the National Key R&D Project (2018YFB2202600), the NSFC (61774094 and U19B2041), the China Major S&T Project (2018ZX01031101002) and the Beijing S&T Project (Z191100007519016). The authors thank Hao Ni and Long Zhou for their advices and helpful feedback.

http://dx.doi.org/10.21437/Interspeech.2020-2677

introduction of the conventional speech transformer used in previous works in Sec.2.1. We then present our network structure and beam search mechanism in Sec.2.2. Finally, we describe the training and inference process of our model in Sec.2.3

2.1. Speech Transformer

The conventional speech transformer model [12] has an encoder-decoder structure, the encoder maps an input sequence of acoustic feature $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ to a sequence of hidden states $\mathbf{H} = \{\mathbf{h}_1, \ldots, \mathbf{h}_n\}$, Given $\mathbf{H}$, the decoder generates the outputs sequence $\{y_1, \ldots, y_m\}$ step by step, where n is the frame number of input sequence, m is the length of the output sequence. Both encoder and decoder are stacked with attention and position-wise feed-forward networks.

2.1.1. Scaled Dot-product Attention

The basic unit of the transformer model is self-attention mechanism, which is designed to map queries and a set of key-value pairs to outputs, where the queries $\mathbf{Q} \in \mathbb{R}^{t_q \times d_q}$, keys $\mathbf{K} \in \mathbb{R}^{t_k \times d_k}$, values $\mathbf{V} \in \mathbb{R}^{t_v \times d_v}$ and outputs are all vectors. The t is the number of elements and d is the feature dimensions of corresponding elements. We usually have $d_q = d_k, t_k = t_v$.

Firstly, queries compute dot-product with all keys, and then divide by scaled factor $\sqrt{d_k}$, which is used to prevent pushing the softmax function into extremely small regions. After softmax, we obtain attention scores which weight the values to generate final outputs. The process of scaled dot-product attention can be described as Eq 1.

$$\text{Attention}\ (\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{1}$$

2.1.2. Multi-head Attention

In speech transformer, multi-head attention is applied to leverage different attending representations jointly. Specifically, it is beneficial to linearly project the queries, keys and values h times with different, learned linear projections to d_k, d_k and d_v dimensions. Then the outputs of multi-head attention is generated by concatenating h different head, which is illustrated in Eq 2.

$$\begin{aligned}&\text{MultiHead}\ (\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{Head}_1, \cdots, \text{Head}_\text{h})\mathbf{W}^O \\ &\text{Head}_\text{i} = \text{Attention}\left(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V\right)\end{aligned} \tag{2}$$

where $\mathbf{W}_i^Q \in \mathbb{R}^{d_m \times d_q}$, $\mathbf{W}_i^K \in \mathbb{R}^{d_m \times d_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{d_m \times d_v}$ are the parameters of linear projections, $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_\text{m}}$, d_m is the feature dimension of the final outputs. Normally, $d_q = d_k = d_v = d_\text{m}/h$.

2.1.3. Position-wise Feed-Forward Networks

In addition to attention sub-layers, each layer of encoder and decoder usually contains a fully connected feed-forward network, which consists of two linear layers with a ReLU activation between them. The outputs of feed-forward networks can be computed as Eq 3.

$$\text{F}(\mathbf{x}) = \max\left(0, \mathbf{x}\mathbf{W}_1 + \mathbf{b}_1\right)\mathbf{W}_2 + \mathbf{b}_2 \tag{3}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_m \times d_f}$, $\mathbf{W}_2 \in \mathbb{R}^{d_f \times d_m}$, $\mathbf{b}_1 \in \mathbb{R}^{d_f}, \mathbf{b}_2 \in \mathbb{R}^{d_m}$, d_f is the feature dimensionality of inner layer.

2.1.4. Standard Beam Search

With a trained speech transformer, beam search is used to find best hypothesis sentences for inference. At each step, we only remain the best N(called *beam size*) hypothesis to generate the outputs of the next step. The process of standard beam search is illustrated in Fig. 1a.

2.2. Speech Transformer with Bidirectional Decoder

In order to exploit the right-to-left contexts, and improve the agreement between different directional hypothesis, we introduce the bidirectional decoder to speech transformer and denote our proposed model as *STBD*.

2.2.1. Structure of STBD

Different with the conventional speech transformer, the STBD uses bidirectional targets to learn the encoder and decoder network. The overall framework of speech transformer with bidirectional decoder(STBD) is illustrated in Fig. 2.

As the figure shows, the STBD includes an encoder and a bidirectional decoder, which consists of two unidirectional decoders with shared weight. The encoder is similar to standard encoder of speech transformer, which stacks N times of sub-layers multi-head attention, feed-forward networks, and adds residual connection between the inputs and outputs of sub-layers. In STBD, there are two different unidirectional decoders, which decode from opposite direction and generate two different directional targets.

2.2.2. Bidirectional decoder

In addition to left-to-right self-attention layers, the decoder contains right-to-left self-attention layers. The inputs of bidirectional dot-product attention consists of queries($[\overrightarrow{\mathbf{Q}}; \overleftarrow{\mathbf{Q}}]$), keys($[\overrightarrow{\mathbf{K}}; \overleftarrow{\mathbf{K}}]$), values($[\overrightarrow{\mathbf{V}}; \overleftarrow{\mathbf{V}}]$), which are concatenated by L2R and R2L states. The bidirectional hidden states $\overrightarrow{\mathbf{H}}$ and $\overleftarrow{\mathbf{H}}$ can be computed as:

$$\overrightarrow{\mathbf{H}} = \text{Attention}\ (\overrightarrow{\mathbf{Q}}, \overrightarrow{\mathbf{K}}, \overrightarrow{\mathbf{V}}) \tag{4}$$

$$\overleftarrow{\mathbf{H}} = \text{Attention}\ (\overleftarrow{\mathbf{Q}}, \overleftarrow{\mathbf{K}}, \overleftarrow{\mathbf{V}}) \tag{5}$$

The bidirectional attention also can be expanded into the multi-head form as:

$$\begin{aligned}&\text{MultiHead}\ ([\overrightarrow{\mathbf{Q}}; \overleftarrow{\mathbf{Q}}], [\overrightarrow{\mathbf{K}}; \overleftarrow{\mathbf{K}}], [\overrightarrow{\mathbf{V}}; \overleftarrow{\mathbf{V}}]) \\ &= \text{Concat}\ \left(\left[\overrightarrow{\mathbf{H}}_1; \overleftarrow{\mathbf{H}}_1\right], \ldots, \left[\overrightarrow{\mathbf{H}}_h; \overleftarrow{\mathbf{H}}_h\right]\right)\mathbf{W}^O\end{aligned} \tag{6}$$

The bidirectional decoder is expanded from the standard decoder of speech transformer. Specifically, it can be seen in Eq 6, left-to-right decoder and the right-to-left decoder share the same network parameter. One unidirectional decoder decodes from left to right, generates the forward contexts, and the other one decodes from right to left, generates the backward contexts. Instead of using $\langle$SOS$\rangle$ as the start of sentences, we introduce two begin symbol $\langle$L2R$\rangle$ and $\langle$R2L$\rangle$, leading to generate different directional sentences separately.

2.2.3. Bidirectional Beam Search

Equipped with the bidirectional decoder, we also use the bidirectional beam search for sequence inference, which is illustrated in Fig. 1b. When the beam size is 4, we decode half of the beam from left to right, and decode the other half from right to left. At each time-step, the alive hypothesis sentences includes two candidates guided by $\langle$L2R$\rangle$, and two candidates guided by $\langle$R2L$\rangle$. The decoding process terminates when the $\langle$EOS$\rangle$ symbol is predicted. Finally, the best hypothesis is determined by the scores of finished sentences. If the hypothesis sentence with

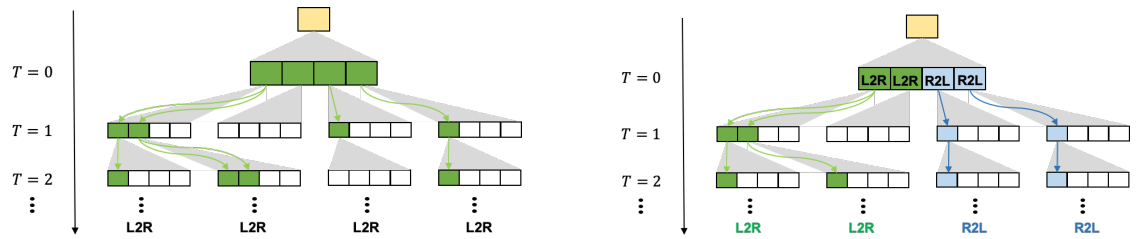

Figure 1: *a), Standard beam search with beam size 4, the green squares are alive beam. b) Bidirectional beam search with beam size 4.*

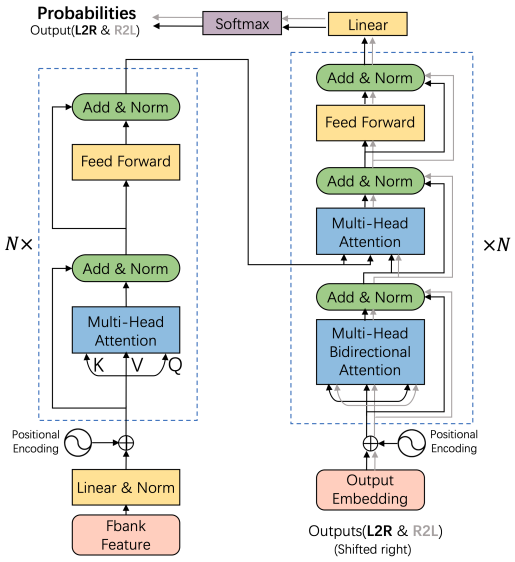

Figure 2: *The structure of bidirectional speech transformer*

the highest score is guided by ⟨R2L⟩, we will reverse the final prediction.

2.3. Training and Inference

In the training process, we jointly optimize the two different directional decoders with two directional targets, and the optimization objective is described as:

$$\mathcal{L}_{\text{total}} = \alpha * \mathcal{L}_{\text{L2R}} + (1 - \alpha) * \mathcal{L}_{\text{R2L}} \tag{7}$$

Table 1: *CER Comparison with Previous Works on AISHELL-1*

Model	Enc-Dec-Dm-Head	Dev CER	Test CER	Para
LFMMI[22]	-	6.44%	7.62%	-
LAS[23]	-	-	10.56%	-
ST[24]	6-6-512-16	-	8.23%	-
SAST[24]	6-6-512-16	7.59%	7.82%	-
SA-T+PA[25]	-	8.3%	9.30%	-
ST-L2R	8-4-512-8	6.52%	7.45%	56.9M
ST-R2L	8-4-512-8	6.62%	7.45%	56.9M
STBD	8-4-512-8	**6.23%**	**7.18%**	56.9M
STBD-Big	8-4-1024-16	**5.80%**	**6.64%**	222.9M

The $\mathcal{L}_{\text{L2R}}$ and $\mathcal{L}_{\text{R2L}}$ are cross-entropy loss computed for two directions. The ratio factor $\alpha \in (0, 1)$ characterizes the preference of the model. In this paper, we suppose that both directions are equally important, and set it to $\alpha = 0.5$.

3. Experiment

We evaluate our methods on the speech recognition by conducting a set of experiments on AISHEEL-1 dataset. In this section, we introduce the experimental setup and report detailed experimental results.

3.1. Experimental Setups

We demonstrate our method on the Chinese Mandarin dataset AISHELL-1 [22], which contains about 178 hours of Mandarin speech audio. The input data is a sequence of 80-dim fbank acoustic features, which are extracted with the frame size is 25ms and the frame shift is 10ms. The global CMVN is used to normalize the features, and 3-times downsample is applied to reduce the input frames of encoder.

AISHELL-1 has 4235 output units in total, which consist of 4230 Chinese characters and 5 extra tokens(an unknown token (<UNK>), a padding token (<PAD>), and two sentence start tokens(<R2L>, <L2R>) and one end tokens (<EOS>)).

According to [12, 26], the transformer for end-to-end speech recognition prefers more layers of encoder than decoder. Thus we build our speech transformer baseline and STBD with 8 layers encoder and 4 layers decoder. We use $d_m = 512, d_f = 2048$ and the number of heads is 8. The Adam optimizer is used with warmup strategy, and we set $\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 10^{-9}$. The learning rate is computed as Eq 8:

$$\text{lr} = k \cdot \min(\text{step}^{-0.5}, \text{step} \cdot \text{warmup\_steps}^{-1.5}) \tag{8}$$

where we set $k = 1.0, \text{warmup\_steps} = 16000$. The feed-forward dropout is set to 0.2 to prevent overfitting. We train the ST-L2R, ST-R2L and STBD for 70 epochs with 4 GPUs. To improve the training efficiency, the batch size is decided dynamically by the length of audios during the training process. Finally, we choose the best 5 epochs with lowest CER on dev set, and average them to get the final model. In inference stage, we use a narrow beam size of 2 and the length penalty is set to 0.6.

3.2. Quantitative Results

The overall comparison results are listed in Tab 1. Compared with speech transformer baseline in [24], we implement a stronger baseline with the similar structure and denote it as **ST-L2R**.

In Tab 1, our strong baseline ST-L2R performs better than LFMMI (chain model) [22] in kaldi tools with 2.2% relative CER gain. By exploiting the bidirectional targets and bidirectional contexts, our proposed **STBD** model can improve the conventional L2R speech transformer with a margin(test CER from **7.45**% to **7.18**%), and achieves **5.8**% relative CER gain compared with LFMMI. We also present the right-to-left speech

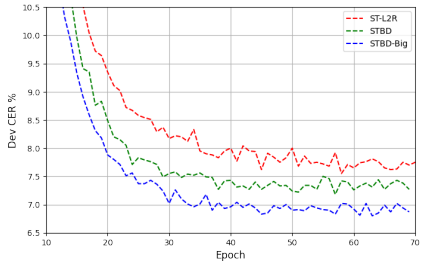

Figure 3: *Validation curve of baselines and our models.*

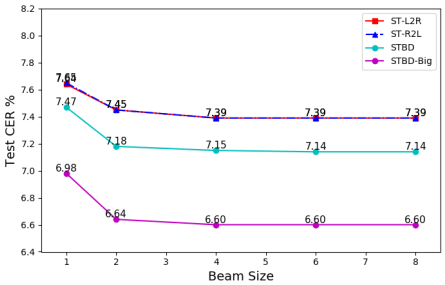

Figure 4: *CER with different beam size*

transformer model(denoted as **ST-R2L**) as the reference. From the perspective of CER, there is not much difference between ST-L2R and ST-R2L. What't more, the two unidirectional decoders in STBD share their weight, thus in Tab 1, we could observe that there are no significant difference in parameters between the STBD and the ST-R2L. We further investigate the STBD with bigger encoder-decoder structure, and refer it as **STBD-Big**, which can achieve **5.8**% for Dev CER and **6.64**% for Test CER. Compared with other previous works, STBD-Big achieves a large margin improvement.

We compare the training process of our STBD models with the strong baseline by plotting validation performance curve during training in Fig.3. It is evident that the STBD is able to improve the convergence and the final performance significantly, which indicates that the bidirectional decoder efficiently.

3.3. Ablation Analysis

3.3.1. Influence of Beam Size

We first investigate various beam size of STBD and baseline models, and plot the performance curve in Fig 4. When the beam size is increased from 1 to 2, the improvements are significant for all models. The performance gain will be saturate when we increase the beam size further. Compared with baseline models, our STBD performs better in all different beam sizes.

3.3.2. Effects of Bidirectional Beam Search

We also conduct comprehensive experiments to verify the effect of bidirectional beam search method. We report the experiments results of with/without bidirectional search in Tab 2.

The default STBD model is trained with bidirectional targets, and we conduct the bidirectional beam search during the inference process as described in Fig 1b. We also conduct the standard beam search on the trained STBD model to generate the prediction and denote these two methods as STBD-BS$_{\text{L2R}}$ and STBD-BS$_{\text{R2L}}$.

In Tab 2, compared with ST-L2R, the STBD-BS$_{\text{L2R}}$ and

Table 2: *CER w/o bidirectional beam search*

Model	Dev CER	Test CER
ST-L2R	6.52%	7.45%
STBD	**6.23**%	**7.18**%
STBD-BS$_{\text{L2R}}$	6.39%	7.33%
STBD-BS$_{\text{R2L}}$	6.48%	7.39%

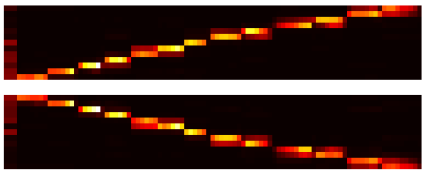

Figure 5: *The attention alignment of bidirectional decoder. The horizontal axis represents the audio timeline, and the vertical axis represents the decoded hypothesis. The brighter color means the higher attention score. The **top** figure is the left-to-right decoder, and the **bottom** figure is the right-to-left decoder in STBD.*

STBD-BS$_{\text{R2L}}$ can also have a little improvement even we use the standard unidirectional beam search, which indicates optimization with the bidirectional targets is helpful for the speech recognition. We check the decoding results and find that the error accumulation exists in the unidirectional baseline, that is, the current character recognition error may affect the following characters recognition. With the error accumulation from different directions as the regularization, this problem can be alleviated in STBD, so that the STBD can gain a little improvement even with standard beam search.

Additionally, we find that some characters are easier to infer from the forward context and some characters are easier to infer from the reverse context. This should explain that the bidirectional beam search can improve the performance of the model. In the experiment, we find that 49.4% of sentences are decoded by backward decoder.

3.3.3. The Visualization of Attention Alignment

We also visualize the vanilla-attention scores between encoder and decoder in Fig 5 for better understanding our proposed bidirectional decoder. From the Fig 5, we find that the attention score of the left-to-right decoder is a diagonal line from the lower left corner to the upper right corner, and the attention score of the right-to-left decoder is a diagonal line from the lower right corner to the upper left corner.

4. Conclusions

In this work, we have proposed a novel speech transformer with directional decoder(STBD), which can exploit both of the right-to-left contexts and left-to-right contexts. The STBD model simultaneously generates different directional targets, and utilize the bidirectional beam search synchronously, in order to update the left-to-right and right-to-left candidates. Finally, we choose the hypothesis sentences with the highest score as final prediction, which may be leaded by ⟨L2R⟩ or ⟨R2L⟩. Furthermore, we demonstrate the efficacy of the STBD model by extensive experiments on the AISHELL-1 dataset, which clearly show out our model has achieved competitive performance even without language model rescoring and additional data enhancement strategies.

5. References

[1] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[2] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[3] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International conference on machine learning*, 2016, pp. 173–182.

[4] K. Audhkhasi, B. Kingsbury, B. Ramabhadran, G. Saon, and M. Picheny, "Building competitive direct acoustics-to-word models for english conversational speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4759–4763.

[5] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[6] T. N. Sainath, Y. He, B. Li, A. Narayanan, R. Pang, A. Bruguier, S.-y. Chang, W. Li, R. Alvarez, Z. Chen *et al.*, "A streaming on-device end-to-end model surpassing server-side conventional model quality and latency," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6059–6063.

[7] B. Li, S.-y. Chang, T. N. Sainath, R. Pang, Y. He, T. Strohman, and Y. Wu, "Towards fast and accurate streaming end-to-end asr," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6069–6073.

[8] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6381–6385.

[9] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[10] V. T. Pham, H. Xu, Y. Khassanov, Z. Zeng, E. S. Chng, C. Ni, B. Ma, and H. Li, "Independent language modeling architecture for end-to-end asr," *arXiv preprint arXiv:1912.00863*, 2019.

[11] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 4945–4949.

[12] 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.

[13] S. Li, D. Raj, X. Lu, P. Shen, T. Kawahara, and H. Kawai, "Improving transformer-based speech recognition systems with compressed structure and speech attributes augmentation," *Proc. Interspeech 2019*, pp. 4400–4404, 2019.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[15] S. Zhou, L. Dong, S. Xu, and B. Xu, "Syllable-based sequence-to-sequence speech recognition with the transformer in mandarin chinese," *arXiv preprint arXiv:1804.10752*, 2018.

[16] H. Miao, G. Cheng, C. Gao, P. Zhang, and Y. Yan, "Transformer-based online ctc/attention end-to-end speech recognition architecture," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6084–6088.

[17] N. Moritz, T. Hori, and J. L. Roux, "Streaming automatic speech recognition with the transformer model," *arXiv preprint arXiv:2001.02674*, 2020.

[18] T. Nakatani, "Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration," 2019.

[19] D. Jiang, X. Lei, W. Li, N. Luo, Y. Hu, W. Zou, and X. Li, "Improving transformer-based speech recognition using unsupervised pre-training," *arXiv preprint arXiv:1910.09932*, 2019.

[20] X. Zhang, J. Su, Y. Qin, Y. Liu, R. Ji, and H. Wang, "Asynchronous bidirectional decoding for neural machine translation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[21] L. Zhou, J. Zhang, and C. Zong, "Synchronous bidirectional neural machine translation," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 91–105, 2019.

[22] 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.

[23] 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.

[24] Z. Fan, J. Li, S. Zhou, and B. Xu, "Speaker-aware speech-transformer," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 222–229.

[25] 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.

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