An Empirical Study on Transformer-Based End-to-End Speech Recognition with Novel Decoder Masking

Shi-Yan Weng[1] Hsuan-Sheng Chiu[2] and Berlin Chen[1]
[1]National Taiwan Normal University, Taipei, Taiwan
E-mail: 60947007S@ntnu.edu.tw, berlin@ntnu.edu.tw
[2]Chunghwa Telecom Laboratories, Taipei, Taiwan
E-mail: samhschiu@cht.com.tw

***Abstract*— The attention-based encoder-decoder modeling paradigm has achieved impressive success on a wide variety of speech and language processing tasks. This paradigm takes advantage of the innate ability of neural networks to learn a direct and streamlined mapping from an input sequence to an output sequence for ASR, without any prior knowledge like audio-text alignments or pronunciation lexicons. An ASR model built on this paradigm, however, is inevitably faced with the issue of inadequate generalization especially when the model is not trained with huge amounts of speech data. In view of this, we in this paper propose a decoder masking based training approach for end-to-end (E2E) ASR models, taking inspiration from the celebrated speech input augmentation (viz. SpecAugment) and masked language modeling (viz. BERT). During the training phase, we randomly replace some portions of the decoder's historical text input with the symbol [mask] to encourage the decoder to robustly output a correct token even when parts of its decoding history are masked. The proposed approach is instantiated with the top-of-the-line transformer-based E2E ASR model. Extensive experiments conducted on two benchmark datasets (viz. Librispeech960h and TedLium2) seem to demonstrate the efficacy of our approach in relation to some existing E2E ASR systems.**

I. INTRODUCTION

End-to-end (E2E) ASR models, particularly with the attention-based encoder-decoder framework [1], have achieved competitive performance in comparison to traditional hybrid DNN-HMM methods [2]. In the E2E models, the encoder directly maps the input acoustic signals to a higher-order semantic representation, which serves as an acoustic model, while the decoder maps the representations into the final output transcriptions acting as a language model. This E2E paradigm, rather than breaking down the ASR process into cascaded submodules, such as acoustic modeling, lexical modeling, and language modeling as those adopted in the conventional hybrid DHH-HMM architecture, can significantly reduce the processing pipeline. Although this E2E paradigm drastically simplifies the ASR pipeline, one well-acknowledged downside is that little is known about how to develop a principled framework to tune specific model components to serve some certain purposes. One observed problem is that the attention-based autoregressive E2E model tends to repeat tokens or words, which implies that the language modeling power of the model may be weak, possibly due to the insufficient amount or quality of the training data, or the mismatch between the training and evaluation conditions. Further, due also to the end-to-end paradigm adopted by the attention-based model, it remains unclear how to systematically improve the predictive power of its decoder given the output history. Recently, non-autoregressive E2E manner [23] has shown its effectiveness and performed comparable results to the autoregressive models.

Although various techniques have been designed and developed to strengthen the language model predictive power of an attention-based model, either by N-best hypothesis rescoring or through the so-called shallow or deep fusion mechanisms [3]. Those improvements are usually mild, and it would inevitably introduce additional computational costs since an external language model may be introduced. Taking inspiration from the celebrated speech input augmentation (viz. SpecAugment [4]) and masked language modeling (viz. BERT [5]), we in this paper put forward a decoder masking approach to improve the robustness of language modeling in the decoder module of the attention-based E2E ASR model, which meanwhile promotes the generalization capacity of the ASR model as well. Unlike SpecAugment [4], which masks out on the spectral features of input speech signals during model training, our approach instead randomly replaces certain tokens (e.g., word or a word-piece) of the partially decoded result of the decoder with the symbol [mask] at each time stamp during model training. The motivation is to encourage the decoder not only to predict the next token but also to fill in the missing token (or alleviate the negative impact caused by caused by imperfect ASR) based on the contextual information. As such and consequently, the ASR model trained with this strategy is anticipated to have a stronger language modeling power and is more robust.

In principle, our approach can be applied to the attention-based E2E framework with arbitrary types of neural networks. However, we focus exclusively on the transformer architecture [6] since it digests a sequence of partially decoded tokens as the decoder's input at each time stamp which provide us more explicit contextual information for improving language modeling in comparison to the RNN/LSTM architectures. Of late, it has been shown that the transformer model can achieve

competitive or even superior recognition performance in relation to the RNN/LSTM-based E2E model for ASR [8]. In contrast to RNN/LSTM, the transformer-based model can capture long-term correlations between the tokens of the partially decoded result with a relatively lower computational complexity, without the need of using many steps of back-propagation through time as done in RNN/LSTM. We evaluate our enhanced transformer model with decoder masking on the Librispeech [9] and TedLium2 [10] benchmark datasets. We show that decoder masking can achieve significant word error rate (WER) reduction on top of SpecAugment, setting highly competitive results on both the test sets of the TedLium2 corpus and on Librispeech corpus for current E2E ASR models. We also compare among disparate masking strategies and their fusion in concert with the transformer-based E2E ASR model, which also confirms the complementarity of our approaches to the existing ones.

II. Related Work

A. SpecAugment

Very recently, SpecAugment [4], a label-preserving data augmentation mechanism, has drawn much attention from the ASR community, since it can achieve a good level of success in increasing the diversity of training data so as to avoid overfitting and improve robustness of ASR models. More specifically, SpecAugment treats the spectrogram (or spectral features) of a speech signal as an image, and in turn warps it along the time axis, mask blocks of consecutive frequency along the time axis bins and mask the whole frequency bins in short spans of time. These operations collectively lead to considerable word error rate reductions on several benchmark tasks, without the need to make any modifications to the ASR models.

B. Semantic Masking

In contrast to SpecAugment, the semantic masking method [11] is more lexical structure-aware in the sense that the time spans of the spectral features of a speech signal to be randomly masked correspond exactly to the lexical tokens conveyed by the speech signal. This method encourages an E2E ASR model to reconstruct masked portions of an input speech signal (that respectively correspond to different lexical tokens) based on their contextual information, which implicitly improves the predictive power of language modeling for the ASR model. However, this method would require additional computational overheads for aligning the spectral features of the training speech signals and their corresponding orthographic transcripts during the training phase, which would be susceptible to alignment errors.

C. BERT

BERT [5] is a mechanism to obtain pretrained contextualized language models, which essentially is a bidirectional encoder that comprises multiple layers of transformer-based neural networks [6]. A BERT-based contextualized language model can be pre-trained on huge amounts of general-domain text and then be fine-tuned on a relatively small amount of task-specific

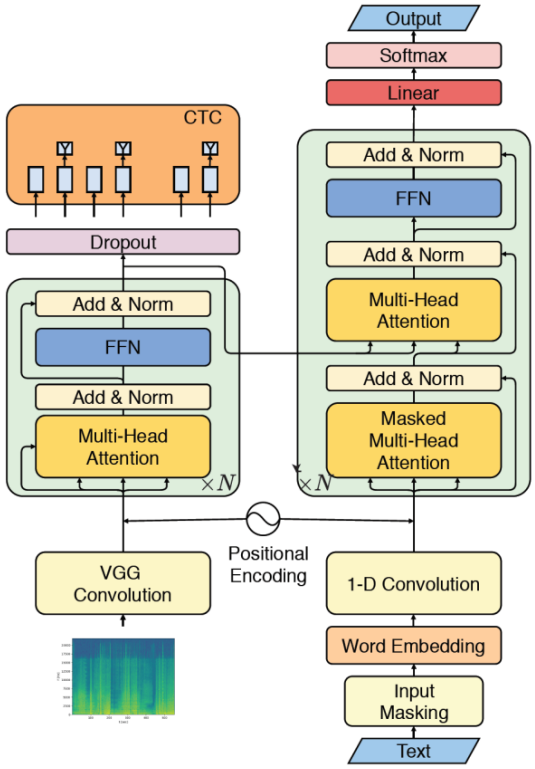

Fig. 1 Overview of our proposed framework.

text, demonstrating excellent performance on many downstream natural language processing (NLP) tasks. BERT originally has two pre-training objectives: masked language modeling (MLM) and next sentence prediction (NSP). MLM randomly replaces some of the input tokens with [mask] symbols and then bases the prediction the original tokens on their both left and right contexts. NSP predicts whether two input sentences appear consecutively in a corpus to model between-sentence cohesion. In this paper, we extend the notion of MLM for training the encoder of an E2E ASR model for better robustness. To this end, we randomly mask the partially decoded results of the decoder during the training phase, with the purpose of making it more robustly predict the output of each time stamp in succession.

III. Decoder Masking

A. Masking strategy

Our decoder masking method (as shown in Figure 2) is a lightweight data-augmentation mechanism, since it requires minimal preprocessing for ASR model training. In the training phase, we randomly selected and masked a certain percentage

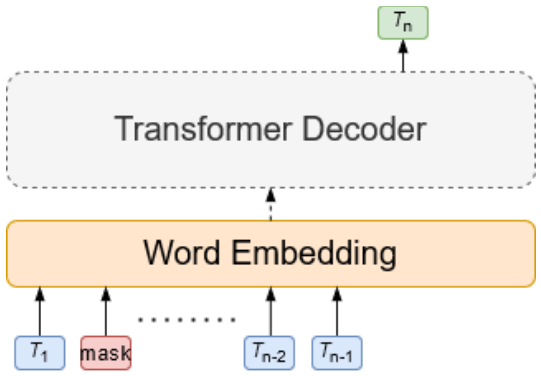

Fig. 2 A schematic depiction of our purposed decoder masking method. To obtain the output prediction T_n, the transformer decoder consumes it corresponding partially-masked historical output $[T_1, [\text{mask}], \ldots, T_{n-2}, T_{n-1}]$.

of the tokens involved in the partially decoded results of the decoder (before they were fed into the decoder) at each time stamp for predicting the next output token. Following [5], in our work, we randomly sampled 15% of the tokens and replaced it with the special [mask] symbol. We also placed a restriction on the masking function so that it will only be activated when the decoding history is accumulated to have more than 15 tokens to avoid unstable training. It should be noted that our decoder masking method is easy to be combined with the other previously proposed masking methods since our method focuses on token-level masking of the decoder while the other masking methods work on the spectral feature-level of a speech signal to be fed into the encoder.

B. *Why Decoder Masking Works*

Our method is similar in spirit to the masked language modeling (MLM) for pre-training of BERT, since both of them are proposed to mask a certain portion of tokens for enhancing the robustness of language modeling. However, the intuitions behind these two methods are different. BERT aims to predict the masked tokens based on both their left and right contexts, with the goal of pre-training its model parameters a priori for possible downstream tasks. The significant difference between BERT and our method is that the ASR decoder can only reach the right-side information, so our method is more like a data-augmentation approach to enhance the robustness of the decoding process and meanwhile alleviate the overfitting problem. Also, during the training phase of an E2E ASR system, the encoder is often viewed as acoustic model while the decoder as language model. In view of the aforementioned aspects, the novelty of our method is the direct integration of the MLM notion into the decoder of E2E ASR in an explicit manner, which is orthogonal to the earlier attempt of semantic masking.

IV. E2E Transformer-Based ASR model

We model the ASR problem as a mapping from an input speech feature vector sequence X to an output syllable sequence Y with Transformer [6]. Transformer can be simplified as a stacking of the following residual-normalization structure [7] to smooth information flow and avoid gradient explosion and vanishing:

$$z^{l+1} = \mathrm{LN}(z^l + f(z^l)) \tag{1}$$

where l denotes index of the layers and $\mathrm{LN}(\cdot)$ is the layer normalization. Function $f(\cdot)$ represents the basic building block of Transformer, such as attention network or feed-forward network which would be formalized in the following subsection. The encoder in Transformer is a stack of L identical layers, with each layer involving a self-attention sub-layer (SAN) and a feed-forward sub-layer (FFN). The decoder uses a similar structure except for an extra cross-attention sub-layer (CAN) inserted in-between the above two sub-layers, which is schematically depicted in Figure 1.

A. *Transformer Block*

In this work, Transformer architecture was implemented following the setup suggested in [8]. The Transformer module consumes input speech feature vectors to obtain their high-level abstraction with a self-attention mechanism. Suppose that Q, K and V are inputs of a transformer block, its outputs are calculated by the following equation:

$$\mathrm{SelfAttention}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{softmax}\left(\frac{\mathrm{QK}}{\sqrt{(d^k)}}\right)\mathrm{V} \tag{2}$$

where d^k is the dimensionality of input feature vectors. To account for multiple attentions, the so-called multi-head attention scheme was adopted, which is expressed by

$$\mathrm{MultiHeadAtt}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \left[\mathrm{H}_1, \ldots, \mathrm{H}_{\mathrm{d_{head}}}\right]\mathrm{W}^{\mathrm{head}} \tag{3}$$

Where $\mathbf{H}_i$ stands for $\mathrm{SelfAttention}(\mathrm{Q}_i, \mathrm{K}_i, \mathrm{V}_i)$ and $\mathrm{d_{head}}$ is the number of attention heads.

B. *ASR Training Process*

Following the previous work [8], we adopted a multi-task learning strategy to train the E2E ASR model. Systematically speaking, both the E2E decoder module and the CTC module predict the distribution of Y at each time stamp given the input X, denoted as $P_{att}(Y|X)$ and $P_{ctc}(Y|X)$. We made use of the weighted average of the following two negative log likelihoods to train our model

$$\mathcal{L} = -\alpha \log \mathrm{P_{att}}(Y|X) - (1-\alpha)\log \mathrm{P_{ctc}}(Y|X) \tag{4}$$

where the interpolation parameter α controls the degree of reliance on $\log P_{att}(Y|X)$ rather than $\log P_{ctc}(Y|X)$. In the test phase, we combined the scores of attention model P_{att}, CTC score P_{ctc} and an RNN-based (viz. LSTM) language model P_{LM} to guide the decoding process, which is formulated by

$$\begin{aligned}\mathrm{P}(\mathrm{y_i}|\mathrm{X}, \mathrm{y}_{<\mathrm{i}}) = {} & \mathrm{P_{ctc}}(\mathrm{y_i}|\mathrm{X}, \mathrm{y}_{<\mathrm{i}}) + \beta_1 \mathrm{P_{att}}(\mathrm{y_i}|\mathrm{X}, \mathrm{y}_{<\mathrm{i}}) \\ & + \beta_2 \mathrm{P_{LM}}(\mathrm{y_i}|\mathrm{X}, \mathrm{y}_{<\mathrm{i}})\end{aligned} \tag{5}$$

V. Experiments

In this section, we conduct a series of experiments on two widely-used ASR benchmark datasets, viz. LibriSpeech [9] and

	Dev		Test	
	clean	other	clean	other
E2E Model				
RWTH (E2E) [14]	2.90	8.40	2.80	9.30
LAS [4]	-	-	3.20	9.80
LAS+SpecAugment [4]	-	-	2.50	5.80
ESPNET Transformer [8]	2.20	5.60	2.60	5.70
Wav2letter Transformers [15]	2.56	6.65	3.05	7.01
+ LM Fusion [15]	2.11	5.25	2.30	5.64
Baseline Transformer	3.51	9.10	3.69	8.95
+ LM Fusion	2.40	6.02	2.66	6.15
Base Model with SpecAugment	3.33	9.05	3.57	9.00
+ LM Fusion	2.20	5.73	2.39	5.94
Base Model with Semantic Masking	2.93	7.75	3.04	7.43
+ LM Fusion	2.09	5.31	2.32	5.55
Base Model with Decoder Masking	2.81	6.97	3.02	7.26
+ LM Fusion	2.10	5.17	2.28	5.39
Large Model with Semantic Masking	2.64	6.90	2.74	6.65
+ LM Fusion	2.02	4.91	2.19	5.19
Large Model with Decoder Masking	2.52	6.43	2.58	6.61
+ LM Fusion	1.98	5.02	2.24	5.15
Hybrid Model				
RWTH (HMM) [14]	1.90	4.50	2.30	5.00
Wang et al. [16]	-	-	2.60	5.59
+ Rescore	-	-	2.26	4.85
Multi-stream self-attention [17]	1.80	5.80	2.20	5.70

TABLE I.
WER results achieved by various ASR methods and their variants on the Librispeech dataset.

SpecAugment	Semantic masking	Decoder masking	Test
-	-	-	10.46
✓	-	-	8.93
-	✓		8.58
-	-	✓	8.44
✓	✓	-	8.26
✓	-	✓	8.19
-	✓	✓	8.11
✓	✓	✓	8.05

TABLE III.
Ablation test (in terms of WER) on TedLium2.

	Test
Kaldi(Chain+TDNN+Large LM)	8.9
ESPnet RNN	11.0
ESPnet Transformer	10.4
+SpecAugment	8.9
+LM Fusion	8.1
+Semantic Masking	8.5
+LM Fusion	7.7
+Decoder Masking	8.4
+LM Fusion	7.5

TABLE II.
Experiment results (in terms of WER) on TedLium2.

TedLium2 [10]. We compare our ASR method with a few top-of-the-line hybrid DNN-HMM and E2E systems. We implemented our method based on ESPnet codebase [8], and the experimental setups for these two datasets were the same as [8], except for the decoding setting, for which we adopted a beam size of 20. In addition, the values of β_1 and β_2 were set to 0.5 and 0.7, respectively.

A. Librispeech 960h

We represented an input speech signal as a sequence of 80-dimensional log-Melfilter bank feature vectors, each of which was appended with 3-dimensional pitch features [18]. SentencePiece [19] was adopted as the tokenizer, and the size of resulting token inventory was 5,000. We trained a baseline E2E ASR model with a 12-layer transformer-based encoder and a 6-layer transformer-based decoder, of which the attention vector size was 512 with 8 heads, amounting to roughly 75M parameters. To explore the influence of using a larger model, we enlarge the model to be equipped with a 24-layer transformer-based encoder and 12-layer transformer-based decoder, having roughly 138M parameters. The setting of the hyperparameters for SpecAugment followed [8] for a fair comparison. We employed the Adam algorithm to update the model, and the warmup step was 25,000. We trained our ASR model for 50 epochs on 4 Titan RTX GPUs, which approximately cost 4 days to coverage. Two additional copies of the original speech training data were created by perturbing the speaking rate of each training utterance to 0.9 times and 1.1 times of its original one, respectively. In this way, the training data had increased three-fold. Following [8], we averaged the parameters of the ASR models obtained at the last 5 checkpoints to form the final model. In addition, the RNN language model used in our experiments was instantiated with LSTM, which was trained using ESPnet.

In the first set of experiments, we evaluate our method from three viewpoints. First, we compare our method with some state-of-the-art methods, whose results, in terms of word error rate (WER), are shown in the first five rows of Table 1. As can be seen from the middle part of Table 1, when working in concert with either the base or the large transformer-based ASR models, our proposed decoding masking method can lead to quite competitive results in comparison to other existing strong E2E ASR models. The performance gap between our method and the other E2E models becomes almost negligible when it is additionally equipped with a language model fusion component. Second, we confirm the utility of our decoding masking method by comparison to the two existing masking methods (viz. SpecAugment and semantic masking). Our method outperforms both semantic masking and SpecAugment when with the base model setting, which indeed shows the solid gains brought by decoder masking for enhancing the robustness of the ASR model. Third, we show the comparison between our method and some well-acknowledged hybrid DNN-HMM systems, whose results are shown in the bottom part of Table 1. As can be seen from Table 1, our method performs comparably with them on the test-clean set, but is still worse than the best hybrid model on the test-other dataset. Furthermore, the improvement achieved by our method on test-other dataset is more pronounced.

B. TedLium2

We further conduct another set of experiments on TedLium2 [10] dataset, which was compiled from TED Talks. In this set of experiments, we followed the same large model setting as those we mentioned in the previous subsection the inventory of distinct output tokens for ASR is set to 1,000.

The experimental results are depicted in Table 2, exhibiting a similar trend as those obtained from the Librispeech dataset.

Our method shows excellent performance on TedLium2 as compared to the two existing masking methods and the strong hybrid DNN-HMM systems. The experiment also confirms the practical feasibility of decoder masking when the speech dataset available for training the ASR models is relatively small.

We also conduct an ablation test on TedLium2 to assess the effectiveness of combining different masking methods. Note here that the corresponding experiments were conducted based on the same settings of Table 2. When working in isolation, our decoder masking method outperforms the other two masking methods. The sixth and seventh rows verify that our method can be effectively paired with either SpecAugment or semantic masking, resulting in further WER reductions. The last row of Table 3 shows that the combination of all three masking methods can achieve the best results. This also reveals the complementary merits of these three masking methods.

VI. Conclusion and Future Work

In this paper, we have proposed a novel data-augmentation method, viz. decoder masking, for end-to-end ASR, which can improve the predictive power of language modeling for the ASR model. Moreover, we have explored the combinations of different masking strategies for use in training the ASR model, and evaluated their effectiveness. The corresponding results have shown that our method can achieve state-of-the-art performance on TedLium2 in relation to several strong E2E ASR systems. As to future work, we plan to explore more variants of training data augmentation to further enhance E2E ASR performance and also explore the effectiveness of combining our proposed method to more powerful neural models [20] and non-autoregressive end-to-end ASR frameworks [21, 22].

References

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, 2015.

[2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82–97, 2012.

[3] S. Toshniwal, A. Kannan, C. Chiu, Y. Wu, T. N. Sainath, and K. Livescu,"A comparison of techniques for language model integration in encoder decoderspeech recognition," in 2018 IEEE Spoken Language Technology Workshop (SLT), 2018, pp. 369–375.

[4] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in Proc. Interspeech 2019, 2019, pp. 2613–2617.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota, June 2019, pp. 4171–4186, Association for Computational Linguistics.

[6] 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 30, pp. 5998–6008. Curran Associates, Inc., 2017.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun "Deep Residual Learning for Image Recognition," in CoRR, abs/1512.03385, 2015

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

[9] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 5206–5210.

[10] A. Rousseau, P. Del´eglise, and Y. Est'eve, "TED-LIUM: an automatic speech recognition dedicated corpus," in Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12),

Istanbul, Turkey, May 2012, pp. 125–129.

[11] C. Wang, Y. Wu, Y. Du, J. Li, S. Liu, L. Lu, S. Ren, G. Ye, S. Zhao, and M. Zhou, "Semantic mask for transformer based end-to-end speech recognition," in arXiv, 2020.

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

[13] L. J. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," CoRR, vol. abs/1607.06450, 2016.

[14] C. L¨uscher, E. Beck, K. Irie, M. Kitza, W. Michel, A. Zeyer, R. Schl¨uter, and H. Ney, "RWTH ASR Systems for LibriSpeech: Hybrid vs Attention," in Proc. Interspeech 2019, 2019, pp. 231–235.

[15] G. Synnaeve, Q. Xu, J. Kahn, E. Grave, T. Likhomanenko, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, "End-to-end ASR: from supervised to semi-supervised learning with modern architectures," CoRR, vol. abs/1911.08460, 2019.

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

[17] K. J. Han, R. Prieto, and T. Ma, "State-of-the-art speech recognition using multi-stream self-attention with dilated 1d convolutions," in 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2019, pp. 54–61.

[18] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 2494–2498.

[19] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Brussels, Belgium, Nov. 2018, pp. 66–71, Association for Computational Linguistics.

[20] A. Gulati, J. Qin, 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 Proceedings of the Interspeech 2020, pp.5036-5040.

[21] Higuchi, Y., Watanabe, S., Chen, N., Ogawa, T., Kobayashi, T. "Mask CTC: Non-Autoregressive End-to-End ASR with CTC and Mask Predict," in Proceedings of the Interspeech 2020, pp.3655-3659.

[22] E. Chi, J. Salazar and K. Kirchhoff. "Align-Refine: Non-Autoregressive Speech Recognition via Iterative Realignment," in Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1920-1927.

[23] J. Gu, J. Bradbury, C. Xiong, V. Li, and R. Socher. "Non-autoregressive neural machine translation." arXiv preprint arXiv:1711.02281., 2017.