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

Pushing the Limits of Non-Autoregressive Speech Recognition

Edwin G. Ng[1*], *Chung-Cheng Chiu*[1], *Yu Zhang*[1†], *William Chan*[2†]

[1]Google Research, Brain Team, USA
[2]Google Research, Brain Team, Canada
eg.ng@alum.utoronto.ca, {chungchengc,ngyuzh,williamchan}@google.com

Abstract

We combine recent advancements in end-to-end speech recognition to non-autoregressive automatic speech recognition. We push the limits of non-autoregressive state-of-the-art results for multiple datasets: LibriSpeech, Fisher+Switchboard and Wall Street Journal. Key to our recipe, we leverage CTC on giant Conformer neural network architectures with SpecAugment and wav2vec2 pre-training. We achieve 1.8%/3.6% WER on LibriSpeech test/test-other sets, 5.1%/9.8% WER on Switchboard, and 3.4% on the Wall Street Journal, all without a language model.

Index Terms: speech recognition, non-autoregressive models

1. Introduction

End-to-end speech recognition models have been tremendously successful, and have achieved state-of-the-art results on various speech recognition tasks [1]. However, much of recent advancements in WER have focused on autoregressive models such as LAS [2] and RNN-T [3, 4]. There has been much recent work on non-autoregressive models as well [5, 6, 7, 8], however these models generally underperform the best autoregressive models. In this work, we are interested in pushing the limits of non-autoregressive speech recognition.

Deep neural networks have been observed to benefit from scaling model capacity [9, 10, 11]. Computer vision classification performance has benefited extensively from larger, deeper models [9, 10, 11]. Similarly, language models have benefited from very large Transformer networks [12, 13, 14]. Recent work in speech recognition have also observed this phenomena for end-to-end autoregressive speech recognition models [3, 4].

Pre-training has significantly improved performance of neural networks for a multitude of tasks, for example natural language understanding [15, 13], object recognition [16], and speech recognition [17]. The idea of pre-training is to train a model with one task to learn useful parameters that can be transferred to other downstream tasks. For example, BERT [13] has shown great success by using the Masked Language Model and Next Sentence Prediction pre-training tasks to achieve strong results on a variety of downstream NLP tasks such as the GLUE tasks [15]. Likewise, wav2vec pre-training [17] uses a similar pre-training task of masking encoded features and optimizing over a contrastive loss to obtain strong representations for the downstream task of automatic speech recognition.

A major challenge of training large models is due to their high capacities and ability to easily overfit the training data. Zhang et al. [4] show that scaling up to larger autoregressive speech recognition models without the use of wav2vec pre-training causes degradation in accuracy and that scaling up model size with wav2vec pre-training enables further performance gains on the LibriSpeech dataset [18].

In this work, we combine recent techniques by training giant Conformer [3] models using wav2vec pre-training and a CTC decoder [19] to push the limits of accuracy for non-autoregressive ASR models. We achieve state-of-the-art non-autoregressive performance with WERs of 1.8%/3.6% on LibriSpeech [18] test-clean/test-other sets, 5.1%/9.8% on Switchboard/CallHome sets [20, 21], and 3.4% on the Wall Street Journal [22] test92 set without utilizing an external language model.

The key contributions of this paper are summarized as follows:

1. We investigate the scaling of model sizes for non-autoregressive speech recognition.
2. We investigate the importance of pre-training in very large models for non-autoregressive speech recognition.
3. We show that by combining large parameter models with pre-training, we can push the limits of non-autoregressive speech recognition. We achieve SoTA on LibriSpeech, Fisher+Switchboard and Wall Street Journal compared to other non-autoregressive work.

2. Related Work

There has been much work done on scaling up models with convolution-based architectures [23, 24, 25, 3]. Jasper [24] uses a series of blocks containing 1D convolutions, batch norm, ReLU, dropout and residual connections to enable deep variants of the network to be trained effectively (up to 54 convolution layers shown in the work) and QuartzNet [24] further improves on the parameter efficiency by replacing the 1D convolutions with depthwise separable convolutions. ContextNet [25] improves the encoding of global context information by adding squeeze-and-excitation modules [26] and also shows clear improvements to WER on LibriSpeech by scaling the model size from 11M to 113M parameters. Our work leverages the Conformer architecture [3], which combines multi-headed self-attention [12] with convolutions to model local and global dependencies of the audio sequence in a parameter efficient way. The Conformer architecture has been observed to improve WER as we scale model capacity from 10M to 100M parameters [3].

However, continuing to scale up model size does not enable continued improvements on performance. In fact, Zhang et. al [4] show that naively scaling up Conformer to model sizes of 600M and 1B actually degrades WER performance on LibriSpeech. However, by adding wav2vec pretraining, Zhang et. al [4] showed continued performance improvements for WERs on LibriSpeech as model sizes are scaled up and achieves state-of-the-art results for autoregressive speech recognition. In our work, we replace the Conformer RNN-T decoder with a CTC decoder and use the same combination of techniques as [4]

*Work done as a part of the Google AI Residency.
†Equal contribution.

Copyright © 2021 ISCA

http://dx.doi.org/10.21437/Interspeech.2021-337

Table 1: *Configurations of our models. We utilize different configurations for our base models (without pre-training) vs. our pre-trained models (with wav2vec pre-training).*

Method	# Params (M)	Enc. Layers	Enc. Dim	Att. Head	Kernel Size	Batch Size
Base						
ConformerCTC 9M	8.9	16	144	4	32	2048
ConformerCTC 28M	27.6	16	256	4	32	2048
ConformerCTC 116M	115.7	17	512	8	32	2048
ConformerCTC 486M	485.6	18	1024	8	32	512
ConformerCTC 631M	631.1	24	1024	8	32	512
Pre-training						
ConformerCTC 120M	119.8	12	1024	8	5	256
ConformerCTC 582M	581.7	24	2048	8	5	256
ConformerCTC 1017M	1016.5	42	2048	8	5	512

without utilizing an external language model to achieve state-of-the-art results for non-autogressive speech recognition on LibriSpeech, Switchboard and Wall Street Journal.

3. Method

3.1. Model Architecture: Conformer

For the neural network architecture, we use a Conformer [3] encoder connected to a CTC decoder. The Conformer encoder is a convolution-augmented transformer which contains a stack of Conformer blocks, where each block has a series of feed forward layers, multi-headed self attention [12], convolutions and layer norm. Like [4], we remove the relative positional embedding [27] from the self attention layer to speed up training for the models with wav2vec pre-training. Conformers of various sizes ranging from 9M to 1B parameters are trained from scratch without pre-training (base models) or with pre-training. The CTC decoder is a linear layer followed by a softmax function to project the encoder output to posterior probabilities. Details on specific architectures are shown in Table 1.

3.2. Wav2vec Pre-Training

We pre-train the Conformer encoder using the variation of wav2vec in [4] with utterances from the "unlab-60k" subset of Libri-Light [28]. Encoded features from the convolution subsampling layer are masked and fed into the Conformer encoder to produce context vectors. Target context vectors are obtained from running a linear layer on the encoded features and is compared with the context vectors from the encoder to optimize the contrastive loss.

4. Experiments

We use wav2vec (w2v) pre-trained Conformers with a CTC decoder to obtain SoTA non-autoregressive performance on Librispeech, Switchboard and Wall Street Journal test sets. We compare with other non-autoregressive models without a language model, since using a language model no longer makes the model end-to-end, and almost all practical language models are autoregressive. We however also include results with a language model from prior work for both non-autoregressive and autoregressive models for the readers convenience.

4.1. Experiment Settings

Data: We follow the same procedure as [4] to pre-train our Conformer models with unlabeled audio from the Libri-Light [28] dataset. We use 80-dimensional log-mel filter bank coefficients of the utterances as single-channel input features for the networks. The pre-trained models are fine-tuned separately on two read speech datasets (i.e. 960h LibriSpeech and 80h Wall Street Journal) and one conversational speech dataset (2000h Fisher+Switchboard [20, 21]). ConformerCTC 9M and ConformerCTC 582M fine-tuned on Wall Street Journal use characters to tokenize the transcripts. All other models use a 1024-token subword vocabulary [29] constructed from the LibriSpeech 960h training set. More details comparing character versus subword tokenization are discussed in Section 5.2.

SpecAugment: We use SpecAugment [1] with two frequency masks using frequency mask parameter $F = 27$ and ten adaptive time masks [30] with a maximum time mask ratio of $p_s = 0.05$ to augment the audio input for all models.

Pre-training Parameters: For masking, we follow the same parameters as [17] by sampling $p = 0.065$ of all time steps to be starting indices and masking the subsequent $M = 10$ time steps. We train with a global batch size of 2048 on 64/256/512 Google TPU V3 chips for 4-5 days for ConformerCTC 120M, ConformerCTC 582M and ConformerCTC 1017M models respectively. We use Adam optimization with a transformer learning rate schedule (section 5.3 of [12]) with 25k warm-up steps and peak learning rate 2e-3/2e-4 for ConformerCTC 120M and ConformerCTC 582M respectively. We apply gradient scaling to cap the norm of the gradient to 20. For ConformerCTC 1017M, we use Adafactor [31] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.98$ and a transformer learning rate schedule with peak learning rate 1e-3 and 25k warm-up steps.

Fine-tuning Parameters: We fine-tune the pre-trained checkpoints with global batch size 256/256/512 on 16/16/64 Google TPU V3 chips for 1-3 days for the ConformerCTC 120M, ConformerCTC 582M and ConformerCTC 1017M models respectively. The encoder and decoder are optimized with different learning rate schedules because the encoder was pre-trained separately. For ConformerCTC 120M and ConformerCTC 582M, we use Adam optimization with a transformer learning rate schedule with a peak learning rate of 3e-4 with 5k warm-up steps for the encoder and a peak learning rate of 8e-4 with 1.5k warm-up steps for the decoder. For ConformerCTC 1017M fine-tuned on LibriSpeech and Fisher+Switchboard, we use Adafactor with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.98$ with a transformer learning rate schedule with a peak learning rate of 3e-4 with 5k warm-up steps for the encoder and a peak learning rate of 5e-4 with 1.5k warm-up steps for the decoder. For ConformerCTC 1017M fine-tuned on Wall Street Journal, we use the same parameters except we lower the encoder peak learning rate to 3e-5 and the decoder peak learning rate to 8e-5.

Parameters for Base Models without Pre-training: To better understand the effects of pre-training, we train from scratch five models of various sizes ranging from 9M to 631M parameters without pre-training. More details comparing pre-trained versus models trained from scratch can be found in Section 5.1. For all models trained from scratch, we train with global batch size 2048 on 64 Google TPU V3 chips for 1-4 days. We use Adam optimization with a transformer learning rate schedule with peak learning rate of 4e-3/3e-3/2e-3/1e-3/1e-3 with 10k warm-up steps for ConformerCTC 9M, ConformerCTC 28M, ConformerCTC 116M, ConformerCTC 486M and ConformerCTC 631M respectively.

4.2. Experimental Results

We present our results compared to previous non-autoregressive work and some non-autoregressive models which use an external LM in Table 2, Table 3 and Table 4. We find that character tokenization performs substantially better than subword tokenization for Wall Street Journal (more details in Section 5.2). We also find that lowering the learning rate for ConformerCTC 1017M on Wall Street Journal, as described in Section 4.1, is critical towards the optimization.

For LibriSpeech and Fisher+Switchboard (evaluated on Hub5'00), we find that our largest model ConformerCTC 1017M performs the best. For Wall Street Journal, the smallest of the three datasets, we find that our second largest model ConformerCTC 582M performs the best. No language model is used in our models to allow the models to stay non-autoregressive. We hypothesize that because Wall Street Journal is a small dataset (80hrs compared to 960h LibriSpeech and 2000h Fisher+Switchboard), that larger models can easily overfit the training data, so a smaller learning rate is required for convergence which causes performance to degrade.

Table 2: *WERs (%) for LibriSpeech. Our ConformerCTC models do not utilize an external language model, yet we outperform many prior work with strong language models.*

Method	LM	test-clean	test-other
Autoregressive			
LAS+SpecAugment [1]	RNN	2.5	5.8
ContextNet [25]	-	2.1	4.6
ContextNet [25]	LSTM	1.9	4.1
Conformer [3]	-	2.1	4.3
Conformer [3]	LSTM	1.9	3.9
Conformer+w2v [4]	-	1.6	3.3
Conformer+w2v+NST [4]	Trans.	**1.4**	**2.6**
Semi-Autoregressive			
Imputer [6]	-	4.0	11.1
Non-Autoregressive			
DeepSpeech2 [32]	5-gram	5.3	13.3
wav2letter++ [33]	ConvLM	3.4	11.2
Jasper [23]	-	3.9	12.0
Jasper [23]	T-XL	2.8	7.8
QuartzNet [24]	-	3.9	11.3
QuartzNet [24]	T-XL	2.7	7.3
CTC+w2v [17]	-	2.2	4.5
CTC+w2v [17]	Trans.	**1.8**	**3.3**
Base (Ours)			
ConformerCTC 116M	-	2.7	5.9
Pre-training (Ours)			
ConformerCTC 1017M	-	**1.8**	**3.6**

Table 3: *WERs (%) for Switchboard.*

Method	LM	SWB	CH
Autoregressive			
Attention Seq2Seq [34]	-	8.3	15.5
RNN-T [35]	4-gram	8.1	17.5
Joint BPE [36]	RNN	**4.9**	**9.5**
Non-Autoregressive			
CTC + Gram-CTC [23]	N-gram	7.3	14.7
Jasper [23]	T-XL	**7.8**	**16.2**
Base (Ours)			
ConformerCTC 116M	-	6.1	12.9
Pre-training (Ours)			
ConformerCTC 1017M	-	**5.1**	**9.8**

Table 4: *WERs (%) for WSJ. Character tokenization significantly improves results compared to subword tokenization. See Section 5.2.*

Method	LM	test92
Autoregressive		
seq2seq + deep conv [37]	-	10.5
Joint BPE [36]	RNN	3.4
Joint BPE + Stoc. Layer [36]	RNN	**3.0**
Semi-Autoregressive		
Imputer [6]	-	12.7
Mask-CTC [38]	-	9.1
Non-Autoregressive		
wav2letter++ [33]	4-gram	5.6
wav2letter++ [33]	ConvLM	**4.1**
Jasper [23]	-	13.3
Jasper [23]	T-XL	6.9
QuartzNet [24]	4-gram	5.8
QuartzNet [24]	T-XL	4.5
Base (Ours)		
ConformerCTC 9M (Char)	-	11.8
Pre-training (Ours)		
ConformerCTC 582M (Char)	-	**3.4**

5. Discussion

5.1. Scaling Model Size and the Importance of Pre-training

In this section, we will study the importance of model size and pre-training. We vary the model size and ablate pre-training. Table 5, Table 6 and Table 7, summarizes the results for LibriSpeech, Switchboard and WSJ.

On LibriSpeech we see that scaling up model size generally improves WER. However, we find that when we want to scale beyond 116M parameter models, we need pre-training to improve the model performance. Without pre-training, we found our models to overfit and generalize poorly. Pre-training allows us to scale to larger models and benefit from its capacity.

On Switchboard we see the exact same trend as LibriSpeech. Scaling up to our 116M parameter model improves WER, but beyond that pre-training is necessary for continual improvements.

On Wall Street Journal we find that larger models worsen the WER. Pre-training significantly improves the WER and allows us to take advantage of the capacity of larger models, however scaling beyond the 582M parameter model also worsens the WER. We hypothesize that this different trend in results is due to the comparatively smaller training size of WSJ compared to the other two datasets. While Librispeech contains 960h of training audio and Fisher+Switchboard contains 2000h of train-

ing audio, WSJ only contains 80h of training audio. With the smaller training set, a high capacity model can easily overfit the data and damage generalizability in the test set. From these results, we see that pre-training allows one to exploit the capacity of larger models and that to continue improving performance of these larger models, the training data must also be sizable to prevent the issue of overfitting the training data.

Table 5: *We compare different model sizes with and without pre-training (Base) on LibriSpeech using subword tokenization. No external language model is used. We observe that pre-training is needed to scale to very large models.*

Method	dev	dev-other	test	test-other
Base				
ConformerCTC 9M	4.4	11.4	4.7	11.2
ConformerCTC 28M	3.0	7.7	3.1	7.6
ConformerCTC 116M	**2.4**	**5.7**	**2.7**	**5.9**
ConformerCTC 486M	2.7	6.2	2.8	6.3
ConformerCTC 631M	2.8	6.9	3.0	6.9
Pre-training				
ConformerCTC 120M	2.5	6.5	2.7	6.4
ConformerCTC 582M	1.9	4.1	1.9	4.0
ConformerCTC 1017M	**1.7**	**3.6**	**1.8**	**3.6**

Table 6: *We compare different model sizes with and without pre-training on Switchboard using subword tokenization. No external language model is used. We observe that pre-training is needed to scale to very large models.*

Method	SWB	CH
Base		
ConformerCTC 9M	8.5	16.2
ConformerCTC 28M	6.3	14.4
ConformerCTC 116M	**6.1**	**12.9**
ConformerCTC 486M	7.9	16.4
ConformerCTC 631M	6.5	15.0
Pre-training		
ConformerCTC 120M	6.5	13.9
ConformerCTC 582M	5.5	12.2
ConformerCTC 1017M	**5.1**	**9.8**

5.2. Tokenization

We compare the character tokenization versus the subword tokenization for Wall Street Journal. In Table 8, we see the character tokenization improves substantially over the subword tokenization for Wall Street Journal. The Wall Street Journal corpus is substantially smaller than Switchboard and LibriSpeech, we hypothesize the subword tokenization is easier to overfit compared to the character tokenization. We also experimented with character tokenization for Switchboard and LibriSpeech, however in those cases we found the subword tokenization to outperform the character tokenization.

6. Conclusion

We combine recent advancements in end-to-end speech recognition, and push the limits of non-autoregressive speech recognition. Key to our recipe, we leverage CTC on giant Conformer neural networks, SpecAugment and wav2vec2 pretraining. Our CTC models establish new state-of-the-art word error rates for non-autoregressive speech recognition on LibriSpeech, Switchboard and Wall Street Journal. Our work outperforms prior non-autoregressive work when compared without a external language model, and even outperforms many autoregressive prior work with strong language models, despite our work not using an language model. We achieve 1.8%/3.6% WER on LibriSpeech test-clean/test-other; on Switchboard, we achieve 5.1%/9.8% WER; and finally on Wall Street Journal we achieve 3.4% WER.

Table 7: *We compare different model sizes with and without pre-training (Base) on WSJ using subword tokenization. No external language model is used. Pre-training improves results, however degradation is observed in our largest model.*

Method	dev93	test92
Base		
ConformerCTC 9M	**22.9**	**19.4**
ConformerCTC 28M	34.9	30.5
ConformerCTC 116M	40.9	36.2
ConformerCTC 486M	37.4	31.9
ConformerCTC 631M	39.4	35.2
Pre-training		
ConformerCTC 120M	8.5	5.7
ConformerCTC 582M	**6.5**	**4.4**
ConformerCTC 1017M	7.1	5.8

Table 8: *We compare two different tokenizations, subword vs. character, for WSJ. We find that character tokenization significantly outperforms subword tokenization on WSJ.*

Method	Vocab	dev93	test92
Base			
ConformerCTC 9M	Subword	22.9	19.4
ConformerCTC 9M	Char	14.0	11.8
ConformerCTC 28M	Subword	34.9	30.5
ConformerCTC 28M	Char	15.9	12.2
ConformerCTC 116M	Subword	40.9	36.2
ConformerCTC 116M	Char	21.0	16.7
Pre-training			
ConformerCTC 120M	Subword	8.5	5.7
ConformerCTC 120M	Char	6.4	4.6
ConformerCTC 582M	Subword	6.5	4.4
ConformerCTC 582M	Char	**5.1**	**3.4**
ConformerCTC 1017M	Subword	7.1	5.8
ConformerCTC 1017M	Char	4.8	4.0

7. Acknowledgements

We give thanks to Samy Bengio, Bo Li, and Anmol Gulati for reviewing our paper.

8. References

[1] 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," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[2] L. Dong, S. Xu, and B. Xu, "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition," in *ICASSP*, 2018.

[3] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[4] Y. Zhang, J. Qin, D. S. Park, W. Han, C.-C. Chiu, R. Pang, Q. V. Le, and Y. Wu, "Pushing the limits of semi-supervised learning for automatic speech recognition," *arXiv preprint arXiv:2010.10504*, 2020.

[5] N. Chen, S. Watanabe, J. Villalba, and N. Dehak, "Non-Autoregressive Transformer Automatic Speech Recognition," in *arXiv*, 2019.

[6] W. Chan, C. Saharia, G. Hinton, M. Norouzi, and N. Jaitly, "Imputer: Sequence modelling via imputation and dynamic programming," *arXiv preprint arXiv:2002.08926*, 2020.

[7] Y. Higuchi, S. Watanabe, N. Chen, T. Ogawa, and T. Kobayashi, "Mask CTC: Non-Autoregressive End-to-End ASR with CTC and Mask Predict," in *INTERSPEECH*, 2020.

[8] Y. Higuchi, H. Inaguma, S. Watanabe, T. Ogawa, and T. Kobayashi, "Improved Mask-CTC for Non-Autoregressive End-to-End ASR," in *arXiv*, 2020.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[12] 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.

[13] J. Devlin, M.-W. 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)*, 2019, pp. 4171–4186.

[14] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[15] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018, pp. 353–355.

[16] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big self-supervised models are strong semi-supervised learners," *arXiv preprint arXiv:2006.10029*, 2020.

[17] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[18] 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)*. IEEE, 2015, pp. 5206–5210.

[19] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International conference on machine learning*. PMLR, 2014, pp. 1764–1772.

[20] C. Cieri, D. Miller, and K. Walker, "The fisher corpus: a resource for the next generations of speech-to-text." in *LREC*, vol. 4, 2004, pp. 69–71.

[21] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 1. IEEE Computer Society, 1992, pp. 517–520.

[22] D. B. Paul and J. Baker, "The design for the wall street journal-based csr corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

[23] J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J. M. Cohen, H. Nguyen, and R. T. Gadde, "Jasper: An end-to-end convolutional neural acoustic model," *Proc. Interspeech 2019*, pp. 71–75, 2019.

[24] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, "Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6124–6128.

[25] W. Han, Z. Zhang, Y. Zhang, J. Yu, C.-C. Chiu, J. Qin, A. Gulati, R. Pang, and Y. Wu, "Contextnet: Improving convolutional neural networks for automatic speech recognition with global context," *arXiv preprint arXiv:2005.03191*, 2020.

[26] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[27] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 2978–2988.

[28] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7669–7673.

[29] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5149–5152.

[30] D. S. Park, Y. Zhang, C.-C. Chiu, Y. Chen, B. Li, W. Chan, Q. V. Le, and Y. Wu, "Specaugment on large scale datasets," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6879–6883.

[31] N. Shazeer and M. Stern, "Adafactor: Adaptive learning rates with sublinear memory cost," in *International Conference on Machine Learning*, 2018, pp. 4596–4604.

[32] 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.

[33] N. Zeghidour, Q. Xu, V. Liptchinsky, N. Usunier, G. Synnaeve, and R. Collobert, "Fully convolutional speech recognition," *arXiv preprint arXiv:1812.06864*, 2018.

[34] C. Weng, J. Cui, G. Wang, J. Wang, C. Yu, D. Su, and D. Yu, "Improving attention based sequence-to-sequence models for end-to-end english conversational speech recognition." in *Interspeech*, 2018, pp. 761–765.

[35] E. Battenberg, J. Chen, R. Child, A. Coates, Y. G. Y. Li, H. Liu, S. Satheesh, A. Sriram, and Z. Zhu, "Exploring neural transducers for end-to-end speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 206–213.

[36] W. Wang, Y. Zhou, C. Xiong, and R. Socher, "An investigation of phone-based subword units for end-to-end speech recognition," *arXiv preprint arXiv:2004.04290*, 2020.

[37] Y. Zhang, W. Chan, and N. Jaitly, "Very deep convolutional networks for end-to-end speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4845–4849.

[38] Y. Higuchi, H. Inaguma, S. Watanabe, T. Ogawa, and T. Kobayashi, "Improved mask-ctc for non-autoregressive end-to-end asr," *arXiv preprint arXiv:2010.13270*, 2020.