U2++ MOE: SCALING 4.7X PARAMETERS WITH MINIMAL IMPACT ON RTF

Xingchen Song[1,2†], *Di Wu*[1,2], *Binbin Zhang*[1,2],
Dinghao Zhou[2], *Zhendong Peng*[2], *Bo Dang*[2], *Fuping Pan*[1], *Chao Yang*[1,2]

[1]GuaSemi Inc., Beijing, China [2]WeNet Open Source Community
xingchen.song@gua.com[†]

ABSTRACT

Scale has opened new frontiers in natural language processing, but at a high cost. In response, by learning to only activate a subset of parameters in training and inference, Mixture-of-Experts (MoE) [1, 2] have been proposed as an energy efficient path to even larger and more capable language models and this shift towards a new generation of foundation models is gaining momentum, particularly within the field of Automatic Speech Recognition (ASR). Recent works [3, 4, 5, 6] that incorporating MoE into ASR models have complex designs such as routing frames via supplementary embedding network, improving multilingual ability for the experts, and utilizing dedicated auxiliary losses for either expert load balancing or specific language handling. We found that delicate designs are not necessary, while an embarrassingly simple substitution of MoE layers for all Feed-Forward Network (FFN) layers is competent for the ASR task. To be more specific, we benchmark our proposed model on a large scale inner-source dataset (160k hours), the results show that we can scale our baseline Conformer (Dense-225M) to its MoE counterparts (MoE-1B) and achieve Dense-1B level Word Error Rate (WER) while maintaining a Dense-225M level Real Time Factor (RTF). Furthermore, by applying Unified 2-pass framework with bidirectional attention decoders (U2++) [7], we achieve the streaming and non-streaming decoding modes in a single MoE based model, which we call U2++ MoE. We hope that our study can facilitate the research on scaling speech foundation models without sacrificing deployment efficiency.

***Index Terms*—** speech recognition, mixture-of-expert, streaming

1. INTRODUCTION

Scaling up neural network models has recently received great attention, given the significant quality improvements on a variety of tasks including natural language processing [8, 9] and speech processing [10, 11].

While training massive models on large amounts of data can almost guarantee improved quality, there are two factors affecting their practicality and applicability: (1) *training efficiency* and (2) *inference efficiency*. Large dense models are often prohibitively compute-intensive to train, with some models requiring TFlops-days of compute [9, 12]. A recent line of work has proposed sparsely-gated Mixture-of-Experts (MoE) layers [1, 2] as an efficient alternative to dense models in order to address both training and inference efficiency limitations.

There have been several related Mixture-of-Expert approaches for ASR modeling [3, 4, 5, 6]. In those models each frame of the input sequence activates a different subset of the experts, hence the computation cost per frame becomes only proportional to the size of the activated sub-network. To avoid collapse to just a few experts while ignoring all others, all of those works use load balancing mechanisms such as dedicated auxiliary losses [2, 13]. Nonetheless, the resulting complex optimization objectives often lead to a large amount of hyper parameter tuning, such as the weight of each auxiliary loss. Moreover, load balancing is designed to address the issue of expert sparsity in the NLP field when routing different tokens. However, this issue may not hold in the speech domain, as there is a high degree of similarity between neighboring speech frames [14]. Forcing speech frames to be evenly distributed among all experts does not align with intuition, as it conflicts with the natural continuity observed in the relationships between adjacent speech frames.

Despite several notable successes of speech MoE, widespread adoption has been hindered by training complexity and the lack of streaming capabilities. We address these with the introduction of U2++ MoE. We simplify the integration of MoE and preclude the necessity for any auxiliary losses. Our proposed method mitigate the complexities, and we show large sparse models may be trained, for the first time, with unified streaming & non-streaming fashion.

2. RELATED WORKS

Several mixture-of-expert strategies have been developed for enhancing ASR modeling, but our work differs from them in the following ways.

1) In contrast to all prior studies [3, 4, 5, 6], our MoE model do not include any auxiliary losses for expert

[†] Corresponding Author.

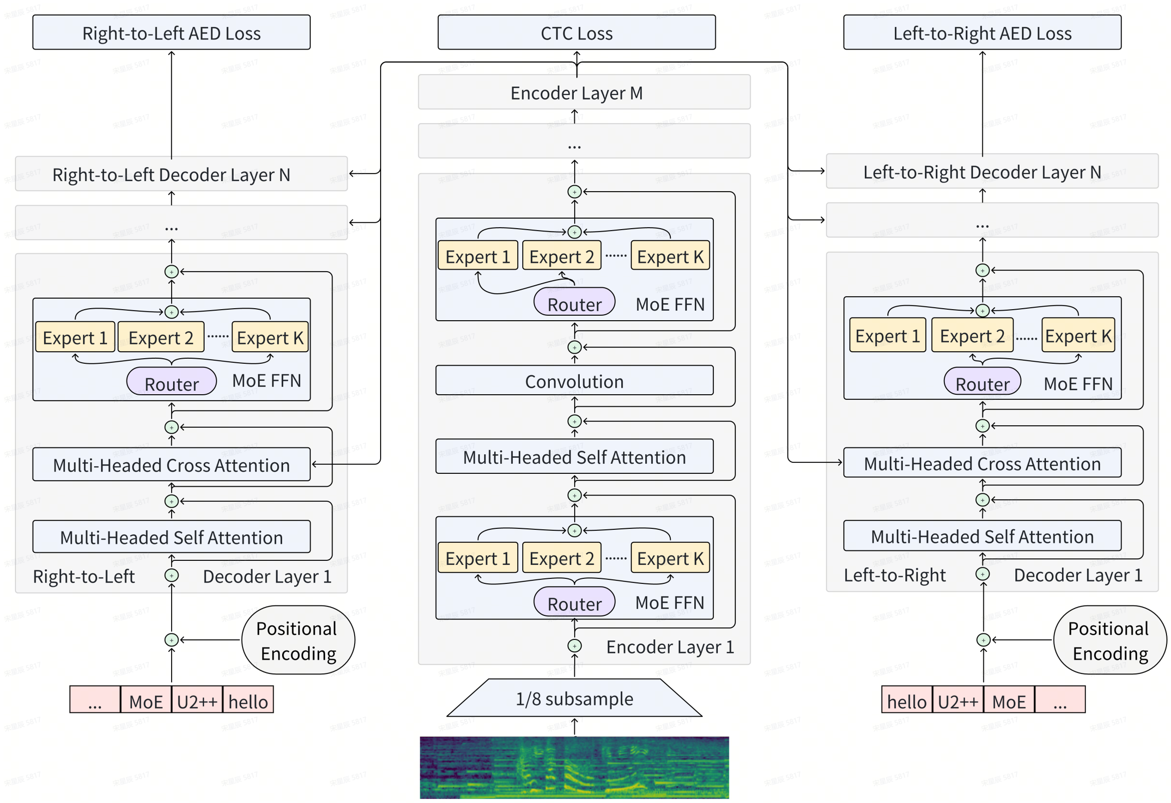

Fig. 1. The proposed U2++ MoE, a unified (streaming and non-streaming) two-pass (encoder for 1st pass decoding and decoder for 2nd pass rescoring) joint CTC/AED framework, enhanced with bidirectional decoders and Mixture-of-Experts. For efficient compression of speech frames, we employ 1/8 subsampling and structure our architecture with M encoder layers alongside $2N$ decoder layers, wherein equal divisions of N layers are allocated to both the right-to-left and left-to-right decoders.

routing, thus significantly streamlining the training optimization process.

2) Compared to [4, 5, 6], our MoE study also works without using any shared embedding networks, thereby simplifying the model architecture and enhancing its generality for model scaling.

3) Compared to all previous works [3, 4, 5, 6] that exclusively explored the application of MoE layers within the encoder, our study extends this innovation by integrating MoE layers into the decoder's FFN as well. Notably, *You et al.* [6] have also attempted to modify all FFN modules in encoder into MoE layers, but it fails to achieve a better performance (*detailed in [6], section 3.2, paragraph 1, last sentence*). In contrast, we are the first to demonstrate the effectiveness of MoE layer substitution across both encoder and decoder components.

4) We are the pioneers in demonstrating the streaming capability of the MoE. While *Hu et al.* [3] have made attempts to integrate MoE layers into a causal encoder to enable streaming recognition, their approach resulted in a notable deterioration in the average WER (*detailed in [3], section 5.1.1, paragraph 2, first sentence*). In stark contrast, our approach, which marries the MoE-based Conformer with the U2++ framework, successfully facilitates both streaming and non-streaming decoding modes within a singular MoE-based model.

5) Our research primarily emphasizes scaling models without a notable increase on RTF, diverging from prior efforts that predominantly concentrate on enhancing the accuracy of multi-lingual or multi-accent recognition [3, 4, 5, 6]. These studies lack a comprehensive analysis of inference latency, such as Dense-1B model v.s. MoE-1B model or Dense-225M model v.s. MoE-1B model. In this paper, however, we demonstrate that a MoE-1B model can achieve the accuracy of a Dense-1B model while maintaining the inference efficiency of a Dense-225M model.

In summary, our guiding principle has been to ***keeping MoE model as simple as possible and is thus more generic for scaling up models***. Our model do not require any auxiliary losses or any additional embedding networks. By applying 1) an embarrassingly simple replacement of all FFN layers with MoE layers and 2) the U2++ framework to

Conformer [15], we prove that MoE-1B model can achieve Dense-1B level accuracy with Dense-225M level inference cost, alongside the capability for streaming.

3. METHODOLOGY

Our model uses Conformer (for encoders) and Transformer (for decoders) as the main building block. A Conformer encoder layer [15] consists of a multi-headed self-attention and a convolution-based layer sandwiched by two FFN. A Transformer decoder layer [16] consists of a multi-headed self-attention, a multi-headed src-attention and one FFN. As shown in Fig.1, to incorporate experts, we use an MoE layer [1, 2] to replace all FFN in the encoders and decoders. Similar to [1, 2], the MoE layer consists of a routing network and multiple experts, each of which is an FFN.

We use the joint Connectionist Temporal Classification (CTC) loss [17] and Autoregressive Encoder Decoder (AED) loss [16] for training the proposed model. The combined loss has two hyper parameters (λ and α) to balance the importance of different losses (*more details can be found in [7], section 2.1*):

$$L = \lambda L_{CTC} + (1-\lambda)(\alpha L_{AED}^{right2left} + (1-\alpha) L_{AED}^{left2right}) \quad (1)$$

Similar to U2 [18], we adopt the dynamic chunk masking strategy to unify the streaming and non-streaming modes. Firstly, the input is split into several chunks by a fixed chunk size C and every chunk attends on itself and all the previous chunks, so the whole latency for the CTC decoding in the first pass only depends on the chunk size. When the chunk size is limited, it works in a streaming way; otherwise it works in a non-streaming way. Secondly, the chunk size is varied dynamically from 1 to the max length of the current training utterance in the training, so the trained model learns to predict with arbitrary chunk size.

4. EXPERIMENTS

4.1. Datasets

Our training corpus comprises mixed datasets gathered from a variety of application domains, amounting to a substantial 160k hours of large-scale, industrial-level training data. This corpus consists predominantly of Mandarin (90%) with the remainder in English (10%).

To evaluate the capabilities of the proposed method, we use the most widely used benchmark for the Mandarin ASR task, namely SpeechIO TIOBE ASR Benchmark [1]. SpeechIO test sets are carefully curated by SpeechIO authors, crawled from publicly available sources (Youtube, TV programs, Podcast etc), covering various well-known scenarios and topics (TV News, VLog, Documentary and so on), transcribed by payed professional annotators thus is exceptionally suitable for testing a model's general speech recognition capabilities. Cumulatively, the 26 publicly available SpeechIO test sets amount to 60.2 hours, averaging 2.3 hours of data across each domain.

4.2. Training Details

In all experiments, we utilize 80-dimensional log-mel filterbank features, computed using a 25ms window that is shifted every 10ms. Each frame undergoes global mean and variance normalization. For modeling Mandarin, we employ character-based representations, whereas for English, we utilize byte-pair encoding (BPE), culminating in a comprehensive vocabulary of 6000 units. All our experiments are conducted in WeNet toolkit [7] with DeepSpeed [19] enabled, all the models are trained using 8 * NVIDIA 3090 (24GB) GPUs.

We have developed three distinct models, as detailed in Table 1, all of which adopt the parameters $Head = 8$, $CNN_{kernel} = 15$, $\lambda = 0.3$, and $\alpha = 0.3$. In the context of the MoE layer, we configure it with 8 experts and enable only the top two experts during both the training and inference phases. For the decoding process, the CTC decoder initially generates the N-Best hypotheses during the first pass. Subsequently, these hypotheses are rescored by the attention decoder in the second pass to produce the final outcomes.

Table 1. Configuration of different models.

(a) Model	(b) M	(c) N	(d) d^{ff}	(e) d^{att}
Dense-225M	12	3	2880	720
Dense-1B	32	6	4096	1024
MoE-1B	12	3	2880	720

4.3. Main Results on 160k hours

In Table.2, we compare the performance of the three models from Table.1 under different conditions (such as the same number of training steps or the same training time), with the results indicating:

1) At the same number of training steps (263k steps), comparing columns (b), (d), and (e) reveals that the WER of the MoE-1B model (3.93) is slightly worse than that of the Dense-1B model (3.72), but both significantly outperform the Dense-225M baseline (4.50).

2) With the same training time (25.9 days), comparing columns (c), (d), and (f) shows that the WER of the MoE-1B model (3.80) is very close to that of the Dense-1B model (3.72), and both substantially surpass the Dense-225M model (4.18).

These results suggest that on a dataset of 160k hours, a larger number of parameters (from 225M to 1B) leads to bet-

[1] https://github.com/SpeechColab/Leaderboard

Table 2. Following the scaling law [20], we compare model WERs on a fixed dataset (160k hours) across equal training steps (236k steps) or compute time (25.9 days).

(a) TestSet	(b) Dense-225M 236k steps, 9.3 days	(c) Dense-225M 657k steps, 25.9 days	(d) Dense-1B 236k steps, 25.9 days	(e) MoE-1B 236k steps, 16.8 days	(f) MoE-1B 364k steps, 25.9 days
speechio_001	1.28	1.15	0.92	0.95	0.90
speechio_002	3.51	3.30	3.03	3.08	2.94
speechio_003	2.34	2.11	1.74	1.68	1.63
speechio_004	2.05	1.96	1.79	1.87	1.93
speechio_005	2.06	1.92	1.84	1.78	1.73
speechio_006	7.24	6.69	6.34	6.35	6.34
speechio_007	10.23	10.12	8.77	9.67	9.23
speechio_008	7.34	6.29	5.78	6.13	5.59
speechio_009	3.94	3.67	3.45	3.60	3.52
speechio_010	4.76	4.68	4.37	4.55	4.49
speechio_011	3.21	2.88	2.31	2.36	2.28
speechio_012	3.39	3.22	2.91	3.01	2.97
speechio_013	4.15	3.81	3.62	3.71	3.69
speechio_014	5.01	4.45	3.87	4.06	3.83
speechio_015	7.58	6.77	6.43	6.69	7.03
speechio_016	5.15	4.46	3.95	4.02	3.82
speechio_017	4.11	3.87	3.24	3.52	3.49
speechio_018	2.69	2.57	2.38	2.56	2.44
speechio_019	3.91	3.29	2.95	3.05	2.90
speechio_020	3.05	2.97	2.33	2.51	2.47
speechio_021	2.75	2.89	2.53	2.73	2.73
speechio_022	5.55	5.15	4.50	4.86	4.52
speechio_023	6.05	5.99	4.89	5.86	5.25
speechio_024	5.61	5.19	4.61	4.76	4.78
speechio_025	5.76	5.30	4.36	4.83	4.61
speechio_026	4.37	4.01	3.90	4.02	3.84
average	4.50	4.18	3.72	3.93	3.80

ter model performance. Moreover, when the number of parameters is the same, MoE models can achieve WER levels comparable to Dense models.

Furthermore, in Table.3, we compare the inference speeds of the three models, with the results showing:

1) Although the MoE-1B and Dense-1B have the same number of parameters, the former is 2.5 times faster than the latter.

2) Even though the parameter count of MoE-1B is 4.7 times that of Dense-225M, the absolute difference in RTF between the two is only around 0.03 (for cpu) or 0.0004 (for gpu).

Overall, combining the WER and RTF results, we can confirm that ***the MoE-1B model can achieve Dense-1B level accuracy with Dense-225M level inference cost***.

Table 3. RTF benchmark. When testing with a CPU, we set the batch size to 1 and perform inference on an int8 quantized model using a single thread on an Intel(R) Core(TM) i5-8400 CPU @ 2.80GHz. For GPU-based evaluations, we set the batch size to 200 and perform inference on an FP16 model using a single NVIDIA 3090. Please note that we do not include GPU RTF for decoder rescoring since the inference time for this process is dominated by the CTC prefix beam search running on the CPU, and therefore, it cannot objectively reflect the inference time on the GPU.

(a) Model	(b) ctc greedy decoding	(c) decoder rescoring
Dense-225M	0.1088 (cpu) / 0.0012 (gpu)	0.1524 (cpu)
Dense-1B	0.3155 (cpu) / 0.0028 (gpu)	0.4515 (cpu)
MoE-1B	0.1299 (cpu) / 0.0016 (gpu)	0.1826 (cpu)

4.4. Streaming Capability

Empirically, training a large model to accommodate both streaming and non-streaming modes simultaneously could potentially compromise performance. In response, this paper introduces a two-stage training pipeline. Initially, we train a non-streaming base model (such as MoE-1B and Dense-225M that is described in Section 4.2 and Table 1), which then serves as the foundation for initializing the proposed U2++-MoE-1B model (and also U2++-Dense-225M, U2++-Dense-1B). The MoE-1B model shares an identical architecture with the U2++-MoE-1B model, with the only distinction lying in their approach to chunk masking. While the MoE-1B employs a full chunk strategy, the U2++-MoE-1B adopts a dynamic chunk method as detailed in section 3. This approach stabilizes the training process for a unified system capable of handling both streaming and non-streaming functionalities.

In Table 4, by comparing three different streaming models, we can draw the same conclusion as in the non-streaming models (section 4.3), which is that our proposed MoE model significantly outperforms the Dense counterpart in terms of WER while maintaining a similar RTF. Please note that the WER for the U2++-Dense-1B model is not included. This is due to the frequent occurrence of gradient explosions during the training process, which, despite the initialization with a non-streaming Dense-1B model, made the training unsustainable.

Table 4. Averaged streaming results on SpeechIO test sets: WER Measured with a 640ms chunk size and RTF calculated using the same hardware (cpu) and methodology (decoder rescoring) as in Table 3. All models were initialized from their respective non-streaming baselines and subsequently trained for a total of 160k steps.

(a) Model	(b) WER	(c) RTF
U2++-Dense-225M	6.24	0.1937
U2++-Dense-1B	N/A	0.6015
U2++-MoE-1B	4.83	0.2436

5. CONCLUSION

The proposed U2++ MoE provides a clean setup and little task-specific design. Through the straightforward substitution of all FFN layers in the baseline model with MoE FFNs, coupled with the adoption of the U2++ training framework, we attain notable enhancements in WER alongside streaming recognition capabilities, all without a considerable increase in RTF.

6. ACKNOWLEDGEMENTS

We thank Wenpeng Li and Jianwei Niu for their feedbacks on this work.

7. REFERENCES

[1] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. 2017, OpenReview.net.

[2] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen, "Gshard: Scaling giant models with conditional computation and automatic sharding," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. 2021, OpenReview.net.

[3] Ke Hu, Bo Li, Tara N. Sainath, Yu Zhang, and Françoise Beaufays, "Mixture-of-expert conformer for streaming multilingual ASR," *CoRR*, vol. abs/2305.15663, 2023.

[4] Zhao You, Shulin Feng, Dan Su, and Dong Yu, "Speechmoe: Scaling to large acoustic models with dynamic routing mixture of experts," in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, Hynek Hermansky, Honza Cernocký, Lukás Burget, Lori Lamel, Odette Scharenborg, and Petr Motlícek, Eds. 2021, pp. 2077–2081, ISCA.

[5] Wenxuan Wang, Guodong Ma, Yuke Li, and Binbin Du, "Language-routing mixture of experts for multilingual and code-switching speech recognition," *CoRR*, vol. abs/2307.05956, 2023.

[6] Zhao You, Shulin Feng, Dan Su, and Dong Yu, "3m: Multi-loss, multi-path and multi-level neural networks for speech recognition," in *13th International Symposium on Chinese Spoken Language Processing, ISCSLP 2022, Singapore, December 11-14, 2022*, Kong Aik Lee, Hung-yi Lee, Yanfeng Lu, and Minghui Dong, Eds. 2022, pp. 170–174, IEEE.

[7] Binbin Zhang, Di Wu, Zhendong Peng, Xingchen Song, Zhuoyuan Yao, Hang Lv, Lei Xie, Chao Yang, Fuping Pan, and Jianwei Niu, "Wenet 2.0: More productive end-to-end speech recognition toolkit," *CoRR*, vol. abs/2203.15455, 2022.

[8] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al., "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, pp. 9, 2019.

[9] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, Eds., 2020.

[10] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, Eds. 2023, vol. 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, PMLR.

[11] Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, Zhong Meng, Ke Hu, Andrew Rosenberg, Rohit Prabhavalkar, Daniel S. Park, Parisa Haghani, Jason Riesa, Ginger Perng, Hagen Soltau, Trevor Strohman, Bhuvana Ramabhadran, Tara N. Sainath, Pedro J. Moreno, Chung-Cheng Chiu, Johan Schalkwyk, Françoise Beaufays, and Yonghui Wu, "Google USM: scaling automatic speech recognition beyond 100 languages," *CoRR*, vol. abs/2303.01037, 2023.

[12] Sneha Kudugunta, Yanping Huang, Ankur Bapna, Maxim Krikun, Dmitry Lepikhin, Minh-Thang Luong, and Orhan Firat, "Beyond distillation: Task-level mixture-of-experts for efficient inference," in *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wentau Yih, Eds. 2021, pp. 3577–3599, Association for Computational Linguistics.

[13] William Fedus, Barret Zoph, and Noam Shazeer, "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity," *J. Mach. Learn. Res.*, vol. 23, pp. 120:1–120:39, 2022.

[14] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James R. Glass, "An unsupervised autoregressive model for

speech representation learning," in *20th Annual Conference of the International Speech Communication Association (Interspeech 2019)*, Gernot Kubin and Zdravko Kacic, Eds., Graz, Austria, 2019, pp. 146–150, ISCA.

[15] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, Helen Meng, Bo Xu, and Thomas Fang Zheng, Eds. 2020, pp. 5036–5040, ISCA.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems (NeurIPS 2017)*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, Eds., Long Beach, USA, 2017, pp. 5998–6008, ACM.

[17] Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *23rd International Conference on Machine Learning (ICML 2006)*, William W. Cohen and Andrew W. Moore, Eds., Pittsburgh, USA, 2006, pp. 369–376, ACM.

[18] Zhuoyuan Yao, Di Wu, Xiong Wang, Binbin Zhang, Fan Yu, Chao Yang, Zhendong Peng, Xiaoyu Chen, Lei Xie, and Xin Lei, "Wenet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit," in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, Hynek Hermansky, Honza Cernocký, Lukás Burget, Lori Lamel, Odette Scharenborg, and Petr Motlícek, Eds. 2021, pp. 4054–4058, ISCA.

[19] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He, "Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash, Eds. 2020, pp. 3505–3506, ACM.

[20] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei, "Scaling laws for neural language models," *CoRR*, vol. abs/2001.08361, 2020.