∞-MoE: Generalizing Mixture of Experts to Infinite Experts

Anonymous ACL submission

Abstract

The Mixture of Experts (MoE) selects a few feed-forward networks (FFNs) per token, achieving an effective trade-off between computational cost and performance. In conventional MoE, each expert is treated as entirely independent, and experts are combined in a discrete space. As a result, when the number of experts increases, it becomes difficult to train each expert effectively. To stabilize training while increasing the number of experts, we propose ∞-MoE that selects a portion of the parameters of large FFNs based on continuous values sampled for each token. By considering experts in a continuous space, this approach allows for an infinite number of experts while maintaining computational efficiency. Experiments show that a GPT-2 Small-based ∞-MoE model, with 129M active and 186M total parameters, achieves comparable performance to a dense GPT-2 Medium with 350M parameters. Adjusting the number of sampled experts at inference time allows for a flexible trade-off between accuracy and speed, with an improvement of up to 2.5% in accuracy over conventional MoE.

1 Introduction

Large language models (LLMs) have recently achieved remarkable performance across a broad range of natural language processing tasks, such as machine translation, question answering, and code generation (Chen et al., 2021; Liu et al., 2021). These advances are primarily driven by scaling up model parameters, training data, and compute resources (Kaplan et al., 2020). However, simply increasing model size leads to substantial computational and memory overheads, motivating research into more efficient strategies for scaling.

Mixture of Experts (MoE) (Shazeer et al., 2017) stands out for its ability to expand parameter count while maintaining relatively low per-token compute costs. By routing each input to a subset of specialized experts, MoE-based architectures can efficiently store large amounts of knowledge sparsely (Dai et al., 2024; Jiang et al., 2024). Recent large-scale models such as DeepSeek(Dai et al., 2024), Mistral(Jiang et al., 2024), and Phi(Abdin et al., 2024) have successfully adopted MoE designs, demonstrating that sparse routing can significantly improve performance without incurring prohibitive computational expense.

A notable trend in recent MoE research is to aggressively increase the number of experts for finer-grained specialization. Empirical evidence shows that larger expert pools improve overall capacity and often yield higher accuracy with similar or reduced compute costs (Fedus et al., 2022; Lepikhin et al., 2020). For instance, PEER (He, 2024) can handle millions of experts, while recent theoretical work (Clark et al., 2022) confirms that MoE performance scales predictably with the expert count.

Following this trend, a natural question arises: **can we achieve even better performance by further increasing the number of experts to infinity?** In principle, having more experts should allow for even more specialized representations, potentially boosting generalization across diverse tasks.

We introduce ∞-MoE, which moves from a discrete set of experts to a continuous domain, allowing theoretically unbounded expert capacity. In this framework, each input samples from a continuum of experts, taking the concept of "increasing experts" to the extreme. Despite the potential for an infinite number of experts, our proposed ∞-MoE remains computationally tractable due to its sparse activation of only a small number of sampled experts at any given time. This design preserves the efficiency of sparse routing while offering significantly enhanced model capacity. Through experiments on GPT-2 Small/Medium (Radford et al., 2019), we observe that the GPT-2 Small-based ∞-MoE variant (129M active parameters, 186M total) achieves comparable performance to a dense

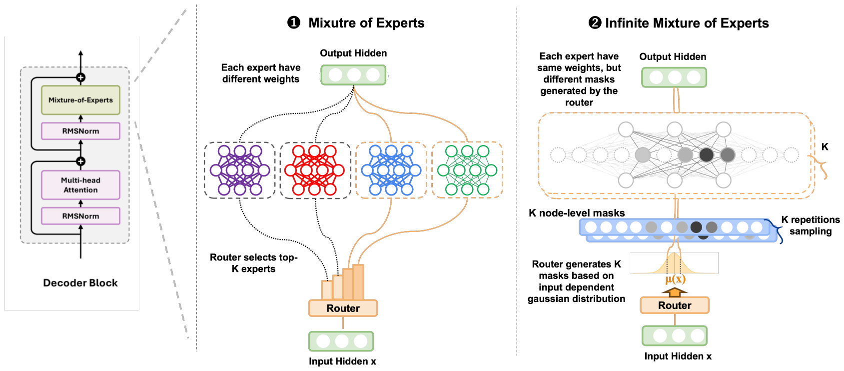

Figure 1: Overview of the proposed Infinite Mixture of Experts (∞-MoE). The router outputs a continuous distribution over the expert space, and each sample selects a unique expert.

GPT-2 Medium model with 350M parameters. Furthermore, increasing the number of samples during inference yields additional accuracy gains, while reducing it still maintains a 2.5% accuracy improvement over standard MoE, enabling flexible trade-offs between speed and accuracy.

2 Related Work

MoE was first proposed to split a problem space into multiple specialized expert networks (Jacobs et al., 1991), and has lately gained popularity for LLMs.

A central advantage in LLMs is that routing each token to just a few experts can greatly expand parameter capacity without a matching increase in compute (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022). For instance, GShard (Lepikhin et al., 2020) and Switch Transformer (Fedus et al., 2022) employ sparse expert activation to train models with hundreds of billions of parameters, though they typically rely on a small expert pool (16 to a few hundred) that restricts specialization.

Recent work addresses this by substantially raising the expert count. PEER (He, 2024) scales up to a million experts, demonstrating richer specialization via novel routing mechanisms. Theoretically, increasing experts improves performance without linearly increasing compute (Clark et al., 2022; Ludziejewski et al., 2024), but router overhead can grow large or over-compressed experts may degrade accuracy (Ludziejewski et al., 2024).

Our approach selects experts at the level of individual FFN nodes or small clusters, offering practically unlimited scalability while keeping routing overhead low. This strategy heightens representation power and scalability without imposing excessive compute costs.

3 Proposed Method

This section presents our ∞-MoE framework. We first introduce a generalized MoE formulation for the standard case, then detail the ∞-MoE model, which extends MoE to a continuous expert space.

3.1 Generalized Expression of MoE

Let $\mathcal{Z} = \{1, 2, \dots, n\}$ be a discrete index set of n experts. Let $x \in \mathbb{R}^{d_{\text{in}}}$ denote the input. Each expert is a function:

$$f(x, i) : \mathbb{R}^{d_{\text{in}}} \times \mathcal{Z} \to \mathbb{R}^{d_{\text{out}}},$$

where $i \in \mathcal{Z}$ indexes the expert. A router produces a probability distribution $p(i|x)$ over experts.

The MoE output is the expected expert output:

$$y \;=\; \sum_{i=1}^{n} p(i \mid x)\, f(x, i) \qquad (1)$$

Connection to Standard MoE. Standard MoE can be seen as a special case where the general expert function $f(x, i)$ simply selects the i-th expert from a set of n pre-defined expert functions, $\{e_1(x), \dots, e_n(x)\}$; that is, $f(x, i) = e_i(x)$. The router typically uses a softmax function to compute the probability of selecting expert i:

$$p(i|x) = \text{softmax}(TopK(g(x)))_i \qquad (2)$$

where $g(x) \in \mathbb{R}^n$ is a vector of scores produced by the router network. With a top-k operation selecting a subset K of experts, the final output is:

$$y = \sum_{i \in K} p(i|x)\, e_i(x). \qquad (3)$$

This clearly demonstrates the standard MoE is special case of this discrete formulation.

Table 1: Zero-shot performance on various benchmarks(BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2021), ARC-e/c (Boratko et al., 2018), OpenBookQA (Banerjee et al., 2019), RACE-high (Lai et al., 2017)). "Active/Total Param" indicates the approximate number of parameters used during forward vs. total parameters.

Model	Active/Total Param	BoolQ(↑)	HellaSwag(↑)	WinoGrande(↑)	ARC-e(↑)	ARC-c(↑)	OBQA(↑)	RACE-high(↑)	Avg(↑)
GPT-2 Small									
Dense	124M/124M	0.601	0.292	0.508	0.431	**0.194**	0.152	0.513	0.385
Switch Transformer	124M/181M	0.601	0.292	0.512	0.431	0.180	0.144	0.513	0.382
MoE	124M/181M	**0.605**	0.295	0.515	0.446	0.185	0.158	0.513	0.388
∞-MoE	129M/186M	0.596	**0.298**	**0.542**	**0.460**	0.189	**0.176**	**0.523**	**0.398**
GPT-2 Medium									
Dense	350M/350M	0.607	0.314	0.488	0.471	0.201	0.176	0.531	0.398
Switch Transformer	350M/556M	0.584	0.315	0.500	0.480	0.200	0.162	0.552	0.399
MoE	350M/556M	**0.593**	0.327	0.507	0.483	0.206	0.178	0.527	0.403
∞-MoE	362M/568M	0.566	**0.337**	**0.516**	**0.497**	**0.215**	**0.188**	**0.570**	**0.413**

3.2 ∞-MoE: Infinite Experts

∞-MoE extends the discrete MoE to a continuous, potentially uncountably infinite, expert space $\mathcal{Z} \subseteq \mathbb{R}^{d_z}$. The router defines a probability density $p(z|x)$ over $\mathcal{Z}$. The model output is:

$$y = \int_{\mathcal{Z}} p(z \mid x)\, f(x, z)\, dz \tag{4}$$

We approximate this integral via Monte Carlo sampling: we sample $z \sim p(z|x)$ and use $f(x, z)$ as an unbiased estimator of y.

Router Design. We use a Gaussian density for the router:

$$p(z \mid x) = \mathcal{N}(z \mid \mu(x), \Sigma(x)), \tag{5}$$

where a small neural network predicts $\mu(x)$ and $\Sigma(x)$ (i.e., all off-diagonal entries are zero) from x. During training, we sample $z^{(k)} \sim p(z \mid x)$ K times ($k = 1, \ldots, K$), allowing the router to learn to allocate probability mass to appropriate regions of $\mathcal{Z}$.

Expert Design. We treat z as a continuous expert index sampled from the router. Intuitively, each distinct value of z corresponds to a different expert in an infinite expert space. Our FFN is then modulated by a mask that "turns off" certain neurons in the intermediate layer, allowing the model to dynamically select which subset of parameters is active.

Formally, let $W_z \in \mathbb{R}^{d_{\text{ff}} \times d_z}$. Given z sampled from Equation 5, we apply a top-$N\%$ operator on intermediate neurons $\hat{m}_i = W_z\, z$, which keeps only the largest $N\%$ of nodes and sets the rest to 0:

$$\text{mask}(z)_i = \begin{cases} \hat{m}_i & \text{if } \hat{m}_i \text{ is top } N\% \text{ values,} \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Because the retained entries preserve their original values, the resulting mask is partially "soft" for the selected positions, while all other positions become strictly zero.

Given this mask, the expert output $f(x, z)$ is computed as:

$$f(x, z) = W_2\Big(\text{Act}(W_1 x) \odot \text{mask}(z)\Big), \tag{7}$$

where $\text{Act}(\cdot)$ is a non-linear activation, $\odot$ is element-wise multiplication, and $W_1 \in \mathbb{R}^{d_{\text{ff}} \times d_{\text{in}}}$, $W_2 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{ff}}}$ are learnable weight matrices. Through this mechanism, each sampled z effectively activates a distinct subset of the FFN's neurons, mirroring the sparsity in conventional MoE models but generalized to an infinite expert space.

4 Experiments

We evaluate the effectiveness of ∞-MoE using GPT-2 Small ($\sim$124M parameters) and GPT-2 Medium ($\sim$350M parameters) on a broad range of natural language understanding tasks.

4.1 Setup

Data. We pre-train our models on a large-scale web corpus called FineWeb (Penedo et al., 2024), from which we extract 10 billion tokens. For fine-tuning or direct evaluation, we use the zero-shot setting on standard NLP benchmarks.

Compared Methods. We compare four architectures:

- **Dense (FFN):** A standard Transformer with a single FFN layer shared by all inputs.
- **Switch Transformer (Top-1):** Routes each token to exactly one expert.
- **MoE (Top-2):** A classic sparse MoE setting that

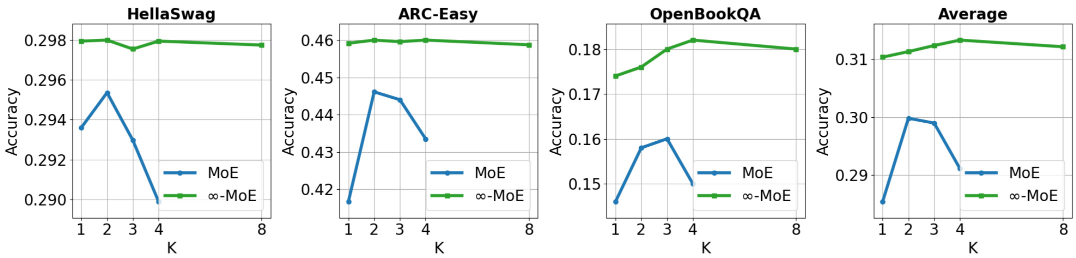

Figure 2: Comparison of MoE and ∞-MoE models on several tasks while varying the number of experts $K \in \{1, 2, 3, 4, 8\}$.For GPT-2 small, $K = 2$ yields 124M active parameters.∞-MoE consistently achieves strong accuracy across a wide range of K, even with fewer experts.Results for additional tasks are presented in the appendix.

activates the top-2 experts for each token. In this configuration, the total number of experts is fixed at 4.

- **∞-MoE:** Our proposed method with an infinite expert space. During both training and testing, two samples are drawn (i.e., $K = 2$); with one sample, only 25% of the overall expert space is active.

4.2 Results

Table 1 presents zero-shot performance on GPT-2 Small and GPT-2 Medium. Across all tasks, ∞-MoE consistently outperforms the Dense baseline, Switch Transformer, and standard MoE. Notably, for GPT-2 Small, ∞-MoE achieves the highest average score of 0.398 versus 0.385 (Dense), 0.382 (Switch), and 0.388 (MoE). We observe similar improvements with the GPT-2 Medium variant, where ∞-MoE again attains the best average accuracy (0.413).

5 Ablations

5.1 Scaling with sampling (K)

Figure 2 compares ∞-MoE with standard MoE across multiple tasks by varying K. In the conventional setup, increasing K can improve accuracy but may also introduce instability at high values. By contrast, ∞-MoE scales more smoothly with K, yielding robust gains and maintaining strong performance even at lower K (achieving a 2.5% improvement over standard MoE). Moreover, treating experts as a continuous space enables flexible inference, allowing users to adjust K based on hardware constraints or latency requirements.

These results demonstrate that ∞-MoE combines the expressiveness of an unbounded expert ensemble with the efficiency of sparse MoE, making it well-suited to a variety of runtime conditions.

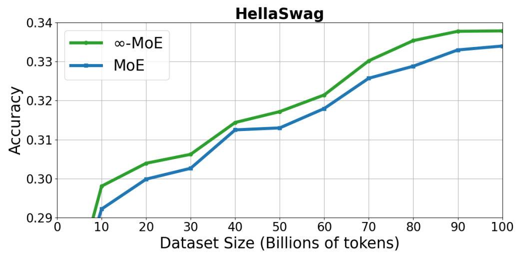

Figure 3: Accuracy on HellaSwag as a function of training data size (in billions of tokens). ∞-MoE is compared against a MoE baseline (GPT-2 small backbone).

5.2 Scaling with Dataset Size

To evaluate the effectiveness of our proposed method, ∞-MoE, under increasing dataset sizes, we conducted experiments using a GPT-2 small architecture as the base model. We measured the accuracy on the HellaSwag dataset, progressively increasing the training data size in increments of 10 billion tokens up to 100 billion. The results are plotted in Figure 3.

6 Conclusion

This paper introduces ∞-MoE, a novel framework that generalizes Mixture-of-Experts (MoE) models to a continuous, and potentially infinite, expert space. By defining a theoretically infinite number of experts, yet sparsely activating only a small, sampled subset, ∞-MoE achieves strong performance while maintaining computational efficiency comparable to standard MoE. Experiments at the scale of GPT-2 Small and Medium models demonstrate that ∞-MoE outperforms both Switch Transformers and standard MoE. Furthermore, ∞-MoE provides a flexible trade-off between inference speed and accuracy by adjusting the number of sampled experts (K) at inference time.

Limitations

While ∞-MoE offers a promising framework for extending Mixture-of-Experts (MoE) models to an infinite expert space, several open challenges remain:

1. **Scaling Beyond GPT-2 Medium.**
   Although our experiments focus on GPT-2 Small/Medium, the behavior of ∞-MoE when scaling to larger models (e.g., GPT-3 and beyond) is not yet fully understood. In particular, it is unclear how performance and efficiency will change when:

   - Increasing the *total* number of parameters while keeping the *active* (per-token) parameter count fixed,
   - Or scaling both active and total parameters in tandem.

   These scenarios raise questions about potential bottlenecks and trade-offs in both training and inference at extreme scales.

2. **Router Distributions.**
   Our current implementation employs a unimodal Gaussian router for simplicity. However, richer distributions—such as mixtures of Gaussians or nonparametric density estimators—could offer more expressive expert allocations, especially in high-dimensional expert spaces. While this may improve coverage of diverse input patterns, designing efficient sampling and sparse-inference mechanisms becomes more complex, and variance reduction in training remains an open challenge.

3. **Applicability to Other Domains.**
   Although our study highlights ∞-MoE's utility in language modeling, it remains unclear how readily this framework generalizes to other domains such as vision (e.g., ViT) or multimodal vision-language models (VLMs). Practical concerns include adapting continuous expert indices to handle different input modalities, ensuring sparse and efficient routing for high-resolution data, and maintaining competitive accuracy in tasks beyond NLP.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019. Careful selection of knowledge to solve open book question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6120–6129, Florence, Italy. Association for Computational Linguistics.

Michael Boratko, Harshit Padigela, Divyendra Mikkilineni, Pritish Yuvraj, Rajarshi Das, Andrew McCallum, Maria Chang, Achille Fokoue-Nkoutche, Pavan Kapanipathi, Nicholas Mattei, Ryan Musa, Kartik Talamadupula, and Michael Witbrock. 2018. A systematic classification of knowledge, reasoning, and context within the ARC dataset. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 60–70, Melbourne, Australia. Association for Computational Linguistics.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Aidan Clark, Diego De Las Casas, Aurelia Guy, Arthur Mensch, Michela Paganini, Jordan Hoffmann, Bogdan Damoc, Blake Hechtman, Trevor Cai, Sebastian Borgeaud, George Bm Van Den Driessche, Eliza Rutherford, Tom Hennigan, Matthew J Johnson, Albin Cassirer, Chris Jones, Elena Buchatskaya, David Budden, Laurent Sifre, Simon Osindero, Oriol Vinyals, Marc'Aurelio Ranzato, Jack Rae, Erich Elsen, Koray Kavukcuoglu, and Karen Simonyan. 2022. Unified scaling laws for routed language models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 4057–4086. PMLR.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. 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)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. 2024. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *Preprint*, arXiv:2401.06066.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Xu Owen He. 2024. Mixture of a million experts.

Robert Jacobs, Michael Jordan, Steven Nowlan, and Geoffrey Hinton. 1991. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

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

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *Preprint*, arXiv:2006.16668.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Jan Ludziejewski, Jakub Krajewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, Marek Cygan, and Sebastian Jaszczur. 2024. Scaling laws for fine-grained mixture of experts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 33270–33288. PMLR.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*. Accessed: 2024-11-15.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *Preprint*, arXiv:1701.06538.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

A Hyperparameter

Details are provided in Table 2.

Table 2: Model and training hyperparameters used in the experiments.

Parameter	GPT2-small	GPT2-medium
Model Hyperparameters		
Block size	1024	1024
Vocab size	50257	50257
Layers	12	24
Heads	12	16
Embedding dim	768	1024
Hidden dim	3072	4096
Gate dim(z dim)	256	256
Training Hyperparameters		
Total batch size	524288	
Gradient accumulation steps	1	
Optimizer	`adamw`	
Learning rate	0.0006	
Weight decay	0.1	
Warmup ratio	0.03	
Warmup iterations	700	
Data type	`bfloat16`	
ZeRO stage	1	

B Total Computation for Experiments

We executed the experiments mainly by running the training for each model using eight nodes, each equipped with eight NVIDIA H200 (141GB) GPUs.

C License

C.1 Model

- GPT-2 small/medium: Modified MIT License

C.2 Dataset

- FineWeb: Open Data Commons Attribution License (ODC-By) v1.0

D Additional Results

Figure 4 presents a comparison of MoE and ∞-MoE models on all tasks.

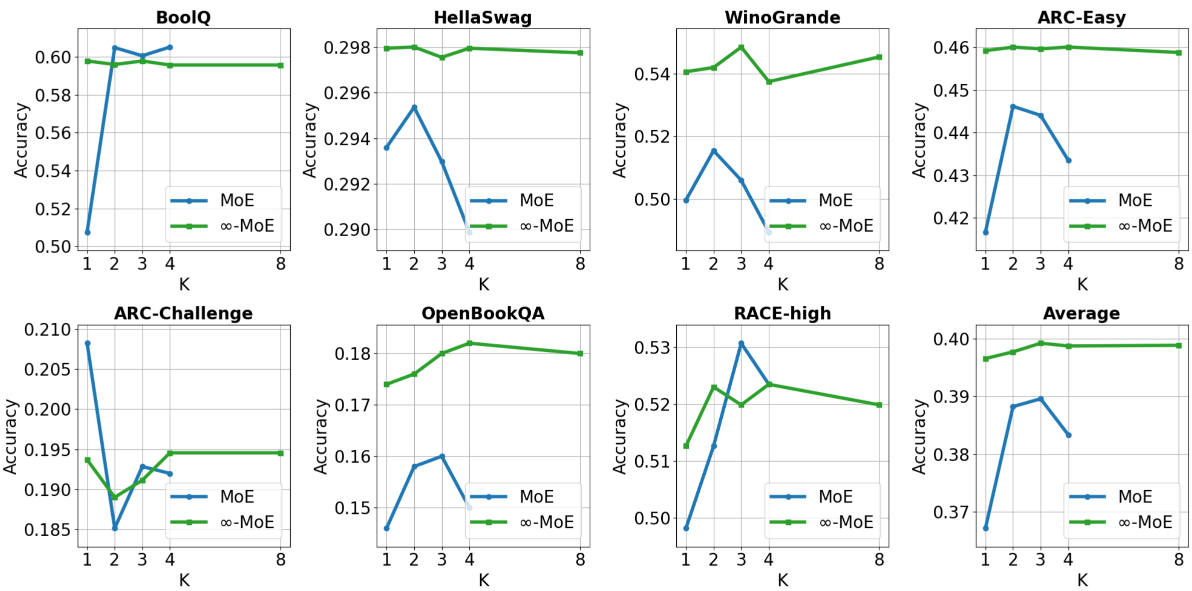

Figure 4: Comparison of MoE and ∞-MoE models on all tasks while varying the number of experts $K \in \{1, 2, 3, 4, 8\}$.