Infinite-parameter Large Language Model

Fei Ding*
AI Lab

Abstract

In the standard transformer architecture, increasing model parameters leads to linear growth in computational cost and activation memory. To address this issue, we propose a novel Infinite Parameter Large Language Model (IP-LLM) architecture that decouples model size from computational cost and device memory. Existing large language models are all fixed-parameter models, while human knowledge is infinite and expands daily. Finite parameters are inherently limited in their capacity to accommodate this boundless knowledge. Our IP-LLM architecture can potentially accommodate infinite knowledge, resolving this issue and laying the foundation for realizing a truly omniscient and omnipotent artificial general intelligence in the future.Our architecture surpasses MOE in performance while requiring significantly less memory.

1 Introduction

Scaling laws for neural language models show the power of scaling (Kaplan et al., 2020; Hoffmann et al., 2022): increasing the number of parameters, amount of training data, or the computational budget has proven to be a reliable way to improve model performance. However, there is a linear relationship between computational footprint, as measured by FLOPs and device memory consumption, and parameter count.

To decouple computational cost from parameter count, we group the parameters of large models, each group storing a specific type of knowledge. During inference, only the relevant parameter group participates in computation, reducing both computational load and device memory consumption.

We partitioned the data into 22 categories and designed a 24B-parameter model. The model comprises 24.5 billion parameters, of which 7.2 billion are dedicated to the base component, 0.7 billion to the routing component, and the remaining 16.6 billion are distributed across 22 distinct categories. During inference, only the 7.2B base, 0.75B router, and the parameters for a single data category (0.75B) are loaded into memory, totaling 8.7B. This represents a 65% reduction in inference memory consumption compared to a fixed 24.5B parameter model.

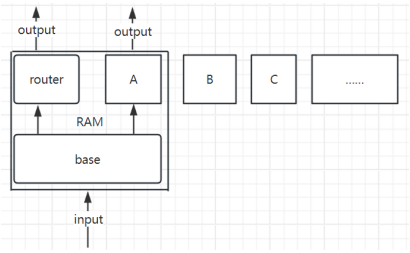

Figure 1: Parameters A, B, C, and D store knowledge for different categories. When reasoning about Category A problems, only parameter A needs to be loaded into memory, eliminating the need to load all parameters.

This paper makes the following contributions:

- Inspired by the routing mechanism of MoE, this paper proposes a novel approach that leverages all model parameters for routing, instead of a subset, significantly enhancing routing accuracy. Our approach first classifies the input text into a specific domain based on model predictions, subsequently employing parameters specialized for that domain to perform inference.

- We proposes a segmented pretraining framework, separating the pretraining process into two phases. The first phase emphasizes the acquisition of foundational linguistic knowledge, including lexical, grammatical, and syntactic elements, as well as basic world knowledge. The second phase then focuses on learning

*Corresponding author
email:dingfei@email.ncu.edu.cn

knowledge built upon this linguistic foundation.

- We propose a novel infinite-parameter large language model capable of lifelong learning without catastrophic forgetting, by strategically training new knowledge onto fresh parameters.

- This innovation results in a drastic reduction in both training cost and inference memory consumption for the large language model. We observe a significant decrease in training cost, while inference memory consumption is lowered by approximately 65%.

2 Related Work

Several recent works (Shazeer et al., 2017; Lepikhin et al., 2020; Fedus et al., 2022; Zhou et al., 2022) have adopted the Mixture-of-Experts (MoE) architecture to decouple computational cost from parameter count.

(Geva et al., 2021; Dai et al., 2022) argue that feedforward (FFW) layers store factual knowledge (Geva et al., 2021; Dai et al., 2022).These layers constitute approximately two-thirds of the total parameters in a transformer architecture. The Mixture-of-Experts (MOE) architecture deviates from the traditional single dense feedforward network (FFW) by utilizing a set of sparsely activated expert modules, frequently implemented as FFWs.

Clark et al. (2022) investigated the scaling properties of MoE language models, demonstrating that increasing the number of experts can effectively enhance performance without incurring additional inference costs. However, their experiments revealed that the efficiency gains offered by MoEs plateau after reaching a particular model size.

More recently, Krajewski et al. (2024) identified that this plateauing phenomenon was a consequence of using a fixed number of training tokens. Their findings demonstrate that when the number of training tokens is optimized for computational efficiency, MoEs consistently outperform dense models in terms of FLOPs (floating-point operations) per parameter. Furthermore, they introduced granularity, the number of active experts, as a novel scaling dimension. Their empirical studies revealed that employing higher granularity leads to improved performance.

3 Task Definition

Our model utilizes a mechanism for selective parameter loading during inference, enabling successful reasoning even under memory constraints. Only a small subset of parameters is required to be retained in memory.

We define a given model f that comprises three specified models f_{base},f_{router},f_A,f_B,f_C and a set of input-output pairs (x, y). we can define this process as:

$$x' = f_{base}(x_i) \tag{1}$$

f_{base} represents the inference process of the parameters in the base part. The input x is subject to parsing via the f_{base}

$$R = f_{router}(x') \tag{2}$$

f_{router} represents the inference process of the parameters in the routing part. After passing through the f_{router}, we obtain R, which signifies the category to which the input belongs.

$$f(x_i) = \begin{cases} f_A(x') & \text{if } R = TokenA \\ f_B(x') & \text{if } R = TokenB \\ f_C(x') & \text{if } R = TokenC \\ ... & \\ x' & \text{if } R \in other \end{cases} \tag{3}$$

f_A represents the inference process of the parameters that encode domain knowledge from A.f_B and f_C follow the same pattern. Based on the determined category, it select corresponding parameters for inference.

4 Training strategy

The dataset is comprised of two distinct components. The first part focuses on training a base model, emphasizing foundational linguistic knowledge including vocabulary, grammar, syntax, and basic world knowledge. The second part consists of domain-specific knowledge, used to train the router and specialized parameters for each domain.

As a first step,in consideration of computational resource constraints, we employ the Qwen1.5-beta-7B-Chat (Bai et al., 2023) model ,a pre-trained language model with strong performance in various tasks,as the base model.

Next, we introduce four additional transformer layers after the final layer of the base model. These

Model	MMLU	C-Eval	GSM8K	MATH
GPT-4	86.4	69.9	92.0	45.8
Llama2-7B	46.8	32.5	16.7	3.3
Llama2-13B	55.0	41.4	29.6	5.0
Llama2-34B	62.6	-	42.2	6.2
Llama2-70B	69.8	50.1	54.4	10.6
Mistral-7B	64.1	47.4	47.5	11.3
Mixtral-8x7B	70.6	-	74.4	28.4
Qwen1.5-7B	61.0	74.1	62.5	20.3
IPLLM-24B	75.2	86.7	80.3	35.5
Qwen1.5-32B	73.4	83.5	77.4	36.1
Qwen1.5-72B	77.5	84.1	79.5	34.1

Figure 2: Comparison

layers are then trained on domain-specific data to acquire specialized knowledge, while the remaining parameters are frozen. After training, these new transformer layers are updated, and the process is repeated for other domains.

After training, the new four transformer layers replace the previous four layers, and the process is repeated for other domains.

Finally, we add four transformer layers to the final layer of the base model to serve as a router. This router is trained using a dataset composed of all domain-specific data, where each data point is labeled with its corresponding domain.

5 Experiments

5.1 Datasets

To ensure fair comparison, we use our proprietary evaluation pipeline. Performance is assessed across a diverse array of tasks categorized as follows:

Popular aggregated results: MMLU (Hendrycks et al., 2020) (5-shot)

Math: GSM8K (Cobbe et al., 2021) (8-shot) with maj@8 and MATH (Hendrycks et al., 2021) (4-shot) with maj@4

The evaluation results, shown in Figure 2, demonstrate a significant performance improvement in our trained model compared to the original model.

6 Conclusion

In this paper, we introduce a novel architecture for large language models that offers significant advantages in terms of reduced device memory requirements for both training and inference, while also enabling the model to learn new knowledge without catastrophic forgetting.

7 Limitations

In this work, we did not train the base model from scratch due to computational constraints. Training the base model from scratch might further enhance performance. We will address the issue of multi-domain knowledge fusion in a subsequent paper.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*.

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*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. *arXiv preprint arXiv:1808.07036*.

Aidan Clark, Diego De Las Casas, Aurelia Guy, Arthur Mensch, Michela Paganini, Jordan Hoffmann, Bogdan Damoc, Blake Hechtman, Trevor Cai, Sebastian Borgeaud, et al. 2022. Unified scaling laws for routed language models. In *International Conference on Machine Learning*, 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. *arXiv preprint arXiv:1905.10044*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research*, 23(1):5232–5270.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2022. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

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. *arXiv preprint arXiv:2001.08361*.

Jakub Krajewski, Jan Ludziejewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, et al. 2024. Scaling laws for fine-grained mixture of experts. *arXiv preprint arXiv:2402.07871*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

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. *arXiv preprint arXiv:2006.16668*.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

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. In *International Conference on Learning Representations*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, , and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. *arXiv preprint arXiv:2305.13172*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix Yu, and Sanjiv Kumar. 2020. Modifying memories in transformer models. *arXiv preprint arXiv:2012.00363*.