Scaling Infrastructure to Support Multi-Trillion Parameter LLM Training

Mikhail Isaev
michael.v.isaev@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Nic McDonald
nimcdonald@nvidia.com
NVIDIA
Salt Lake City, Utah, USA

Richard Vuduc
richie@cc.gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

We wish to consider what software and system configurations might permit existing Large Language Models (LLMs), now at about 1 trillion parameters [8], to scale with greater efficiency to even larger model sizes.[1] Our analysis is driven by the continued success and efficacy of LLMs in a variety of applications [1, 2, 6, 8, 10, 12, 13] and motivated by the observation that Model FLOPS Utilization (MFU)—a common metric of efficiency for assessing how well specialized Artificial Intelligence (AI) accelerators are utilized during model training—can be as low as 50% or less [11]. A significant improvement to MFU will be necessary to increase model sizes by 10× (10 trillion parameters) or higher on architectures similar to current systems. With a space requirement of 20 bytes per parameter, to store just the model's weights and optimizer state we would need more than 200 TB of memory. For a system based on NVIDIA H100 [9] Graphics Processing Unit (GPU) with 80 GiB of high bandwidth memory (HBM) memory, we would need 2,500 GPUs and a fully model-parallel implementation to train such a model. No known model-parallelism technique at this scale would be able to provide anywhere near 50% MFU. Motivated by this example, we aim to establish the system limitations that prevent us from training multi-trillion parameter LLMs on large systems built using clusters of 8 interconnected GPUs, similar to NVIDIA DGX and HGX.

We start by presenting a methodology for choosing well structured multi-trillion parameter LLMs. We focus on the LLM's *aspect ratio* defined as the ratio between the hidden dimension of the transformer block to the number of blocks (a.k.a., transformer layers). Some recent research claims the ideal aspect ratio is a constant 128 [5], while others claim that the aspect ratio should increase exponentially with the number of blocks [7]. Both of these analyses were performed on LLMs 2 to 5 orders of magnitude smaller than today's production LLMs. In the absence of consensus among the LLM experts, we follow the current practice of extrapolating aspect ratios linearly with the number of transformer blocks.

For performance estimation we use Calculon [3], a fast open source analytical model of LLM training performance modeling that we developed.[2] Calculon can estimate the time and resource usage for a given LLM, system configuration, and software execution strategy in about 1 millisecond, allowing the exploration of large design spaces having many billions of such configurations. Calculon models LLM training with tensor parallelism (TP), pipeline parallelism (PP), and data parallelism (DP), allowing searches to determine optimal split-parallelism configurations. The system specification describes an accelerator-based distributed system with a two-level memory hierarchy connected to multiple networks.

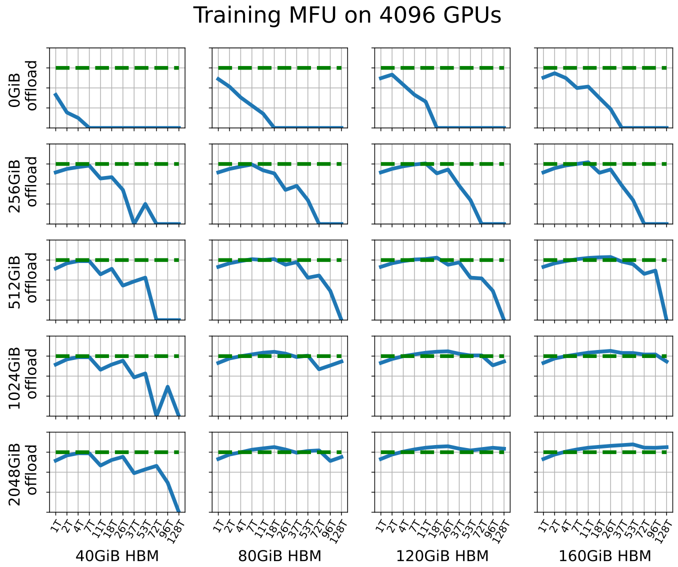

Figure 1: Model FLOPS Utilization (MFU) of LLMs ranging from 1 to 128 trillion parameters on systems with various HBM and offloadinig memory capacity. Green dashed line represents 75% MFU.

Overall, we find it will be critical to co-design the LLM, software, and hardware to attain high performance and efficiency. Noticing that large LLM training performance bottleneck is usually memory, we propose a novel system design with larger pools of slower memory for tensor offload based on reuse patterns. After searching a space of billions of system configurations and execution strategies that provide best performance with given hardware configuration, we found out that current H100 GPUs with 80 GiB of HBM enabled with 512 GiB of tensor offloading capacity allows scaling to 11T-parameter LLMs; and getting to 128T parameters requires 120 GiB of HBM and 2 TiB of offloading memory, yielding 75%+ MFU, which is uncommon even when training much smaller LLMs today.

In conclusion, our findings can be summarized as follows:

(1) Training a hundred-trillion parameter LLM is feasible but requires a secondary memory pool up to 1 TiB per GPU with a bandwidth of 100 GB/s bidirectionally.
(2) Strong scaling for a 1T model stalls around 12,288 GPUs, as matrix multiplication becomes small, inefficient, and unable to overlap with communication.

[1] This work was presented at ASSYST, a non-archival workshop co-located with ISCA. Full workshop paper [4] is available at https://openreview.net/forum?id=rqn2v1Ltgn0

[2] The full description of Calculon will be available in a paper by Isaev M. et al. "Calculon: a methodology and tool for high-level co-design of systems and large language models" accepted as part of SC'23 technical program.

(3) Scaling beyond 10T models requires more first-level memory, with HBM size scaling with model size.
(4) Growing model and system size beyond 10T parameters and 10k GPUs demands a larger fast-network domain and more targeted software optimizations.

REFERENCES

[1] 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. 2020. Language Models Are Few-Shot Learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) *(NIPS'20)*. Curran Associates Inc., Red Hook, NY, USA, Article 159, 25 pages.

[2] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. PaLM: Scaling Language Modeling with Pathways. arXiv:2204.02311 [cs.CL]

[3] Mikhail Isaev and Nic McDonald. 2022. Calculon. https://github.com/calculon-ai/calculon

[4] Mikhail Isaev, Nic McDonald, Jeffrey Young, and Richard Vuduc. 2023. Scaling Infrastructure to Support Multi-Trillion Parameter LLM Training. *Architecture and System Support for Transformer Models (ASSYST @ISCA 2023)* (2023). https://openreview.net/forum?id=rqn2v1Ltgn0

[5] 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:2001.08361 [cs.LG]

[6] Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. 2023. GPT-4 Passes the Bar Exam. *SSRN Electronic Journal* (2023). https://ssrn.com/abstract=4389233

[7] Yoav Levine, Noam Wies, Or Sharir, Hofit Bata, and Amnon Shashua. 2020. Limits to Depth Efficiencies of Self-Attention. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 22640–22651. https://proceedings.neurips.cc/paper_files/paper/2020/file/ff4dfdf5904e920ce52b48c1cef97829-Paper.pdf

[8] Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Patwary, Vijay Korthikanti, Dmitri Vainbrand, Prethvi Kashinkunti, Julie Bernauer, Bryan Catanzaro, Amar Phanishayee, and Matei Zaharia. 2021. Efficient Large-Scale Language Model Training on GPU Clusters Using Megatron-LM. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (St. Louis, Missouri) *(SC '21)*. Association for Computing Machinery, New York, NY, USA, Article 58, 15 pages. https://doi.org/10.1145/3458817.3476209

[9] NVIDIA. 2022. NVIDIA H100 Tensor Core GPU Architecture. https://images.nvidia.com/aem-dam/en-zz/Solutions/data-center/nvidia-ampere-architecture-whitepaper.pdf

[10] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[11] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon Emissions and Large Neural Network Training. https://doi.org/10.48550/ARXIV.2104.10350

[12] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019).

[13] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs.CL]