Editorial

The power of big models: Unleashing opportunities for cloud computing

Yanying Lin

Shenzhen Institute of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 51800, China; yy.lin1@siat.ac.cn

Abstract: The proliferation of deep models characterized by an abundance of parameters has catalyzed research enthusiasm in the domain of AI systems. The emergence of novel computational modalities has brought forth numerous fresh challenges within the realm of cloud computing, encompassing aspects such as cost, performance, elasticity, and the intricate tradeoffs entailed therein.

Keywords: Big model; Cloud computing; Deep learning; Model inference.

Introduction and Observation

The recent advancements in the field of deep learning, such as ChatGPT[1,2], BERT[3], and LLaMA[4] have spurred research in the realm of cloud computing pertaining to the exploration of large-scale models. These deep models, characterized by their formidable parameter magnitudes, have exhibited a noteworthy propensity for generalization. Concurrently, the amelioration of their overall capabilities has engendered a substantial demand for bespoke solutions. Consequently, cloud computing is poised to assume the role of the principal undergirding infrastructure for customized largescale model inference in the forthcoming era. Nonetheless, this development has engendered pervasive apprehensions regarding the adaptability of extant cloud computing infrastructure, which is primarily tailored to accommodate lightweight applications, such as microservices, to the shifting paradigms encapsulated within this burgeoning landscape[5–9].

Large-scale models typically require substantial computational resources, thereby imposing high infrastructure requirements when executed within on premises data centers, often necessitating the deployment of thousands of accelerator cards. Consequently, the utilization of cloud computing infrastructure, such as distributed clusters, has become a prevalent approach for conducting both inference and training tasks. By leveraging the exceptional elasticity and pay-as- you-go resource-sharing capabilities of cloud computing, it becomes economically feasible to ensure high-performance operations for large-scale models, encompassing desirable attributes such as low response latency and high throughput capacities. As the demand for increasingly specialized and tailored services continues to rise, the adoption of cloud computing for computational purposes emerges as an inexorable trend. Nonetheless, the cloud computing infrastructure initially designed to cater to traditional service models encounters challenges when confronted with the distinct computational patterns associated with large-scale models.

One formidable challenge that looms large in the realm of large-scale model inference is the constricting elasticity of cloud computing engendered by the specter of cold starts. While long-term online traffic exhibits discernible periodicity, the capricious nature of transient bursts eludes precise prognostication. Thus, the expeditious scalability to contend with unforeseen upswings becomes an imperative concomitant to the act of downscaling. Moreover, the quantitative quantification of resource interference and the consequent implementation of robust interference isolation mechanisms unfurl as latent hurdles. Ergo, the quandary of whether and when to effect resource amalgamation crystallizes as a topic of discernible import warranting meticulous discourse.

Received: 6 July 2023 Accepted: 13 July 2023 Available online: 20 Novermber 2023

During the training or finetuning stages of large-scale models, it is customary to synchronize an extensive array of parameters at each iteration. This practice, in turn, engenders potential challenges on the data center networking front. Furthermore, owing to the requisite persistence of parameters at every training step, storage conundrums ensue within the domain of cloud computing, encompassing a confluence of localized and centralized storage paradigms. In the inference phase, data transmission assumes a relatively parsimonious guise, albeit the aspect of latency assumes paramount importance. For instance, empirical observations attest that the inference time for a GPT model boasting 175 billion parameters typically hovers within the realm of 300 milliseconds. Consequently, the aegis of averting SLO transgressions becomes a panoramic discourse, spanning the breadth of the landscape.

Contemporary approaches primarily concentrate on optimizing the utilization of cluster resources during the training phase[9–16], with a specific emphasis on harnessing model parallelism to enhance resource efficiency. This entails decomposing the model into sub-models structured as chains or directed acyclic graphs to bolster parallelism or harnessing intra-operator parallelism to expedite response speed. However, this approach may prove inadequate for the forthcoming era of customized large-scale model cloud computing. Within the realm of cloud computing, a salient discernment lies in the envisioning of a manifold of models, epitomizing an unparalleled tapestry of diversity, underscored by their bespoke and expertly tailored attributes.

Concomitantly, an abundance of proprietary models graces the landscape, ushering forth the sine qua non of data privacy and endowing the fabric of security isolation with resolute fortitude. Directly applying training methodologies to inference in customized models does not yield the desired optimization performance. The characteristics imposed by processes like backpropagation, parameter aggregation, and parameter parallelism during model training create constraints that render existing optimization methods unsuitable for the inference stage.

The characteristics of the training phase, including back- propagation, parameter merging, and parameter parallelism, impose constraints that render existing optimization methods inapplicable to the inference stage. Nevertheless, inference already holds an overwhelming numerical advantage in current cloud services, as exemplified by Amazon Web Services, which reports allocating 90% of its resources to inference. Presently, research on optimizing model inference performance remains focused on single-cluster scenarios. Strategies such as traffic prediction, request consolidation, and memory tensor merging are explored to enhance resource utilization and efficiency[11,17–21].

The systems research domain calls for a holistic solution that strikes a balance between performance and cost. This paradigm engenders an open-ended discourse, replete with an expansive terrain awaiting exploration, thereby constituting a pivotal focal point for the next phase of inquiry within the systems research domain.

Conflict of Interest

The authors declare no conflict of interest.

References

1. Brown TB, Mann B, Ryder N, et al. Language models are few-shot learners. arXiv 2020; arXiv:2005.14165. doi: 10.48550/arXiv.2005.1416.
2. Bubeck S, Chandrasekaran V, Eldan R, et al. Sparks of artificial general intelligence: Early experiments with GPT-4. arXiv 2023; arXiv:2303.12712. doi: 10.48550/arXiv.2303.12712.
3. Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv 2019; arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805
4. Touvron H, Lavril T, Izacard G, et al. LLaMA: Open and efficient foundation language models. arXiv 2023; arXiv:2302.13971. doi: 10.48550/arXiv.2302.13971
5. Jain P, Kumar S, Wooders S, et al. Skyplane: Optimizing transfer cost and throughput using cloud-aware overlays. In: Proceedings of the 32nd USENIX Security Symposium (USENIX 2023); 9–11 August 2023;

Anaheim, CA, USA. pp. 1375–1389.
6. Li C, Yao Z, Wu X, et al. DeepSpeed data efficiency: Improving deep learning model quality and training efficiency via efficient data sampling and routing. arXiv 2023; arXiv:2212.03597. doi: 10.48550/arXiv.2212.03597
7. Li Z, Zheng L, Zhong Y, et al. AlpaServe: Statistical multiplexing with model parallelism for deep learning serving. arXiv 2023; arXiv:2302.11665. doi: 10.48550/arXiv.2302.11665
8. Yu M, Cao T, Wang W, Chen R. Following the data, not the function: Rethinking function orchestration in serverless computing. In: Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation (ONDI 2023); 17–19 April 2023; Boston, MA, USA. pp. 1489–1504.
9. Zhang H, Tang Y, Khandelwal A, Stoica I. SHEPHERD: Serving DNNs in the wild. In: Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2023); 7–19 April 2023; Boston, MA, USA. pp. 787–808.
10. Bai Z, Zhang Z, Zhu Y, Jin X. PipeSwitch: Fast pipelined context switching for deep learning applications. In: Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2020); 4–6 November 2020; Banff, Alberta, Canada. pp. 499–514.
11. Gujarati A, Karimi R, Alzayat S, et al. Serving DNNs like clockwork: Performance predictability from the bottom up. In: Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2020); 4–6 November 2020; Banff, Alberta, Canada. pp. 443–462.
12. Huang Y, Cheng Y, Bapna A, et al. GPipe: Efficient training of giant neural networks using pipeline parallelism. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS 2019); 8–14 December 2019; Vancouver, Canada. pp. 103–112.
13. Narayanan D, Harlap A, Phanishayee A, et al. PipeDream: Generalized pipeline parallelism for DNN training. In: Proceedings of the 27th ACM Symposium on Operating Systems Principles (SOSP 2019); 27–30 October 2019; Huntsville, ON, Canada. pp. 1–15.
14. Yu GI, Jeong JS, Kim GW, et al. Orca: A distributed serving system for transformer-based generative models. In: Proceedings of the 16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2022); 11–13 July 2022; Carlsbad, CA, USA. pp. 521–538.
15. Zheng L, Li Z, Zhang H, et al. Alpa: Automating inter- and intra-operator parallelism for distributed deep learning. In: Proceedings of the 16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2022); 11–13 July 2022; Carlsbad, CA, USA. pp. 559–578.
16. Zhou Z, Wei X, Zhang J, Sun G. PetS: A unified framework for parameter-efficient transformers serving. In: Proceedings of the 2022 USENIX Annual Technical Conference (USENIX ATC 2022); 11–13 July 2022; Carlsbad, CA, USA. pp. 489–504.
17. Bhattacharjee A, Chhokra AD, Kang Z, et al. BARISTA: Efficient and scalable serverless serving system for deep learning prediction services. In: Proceedings of the 2019 IEEE International Conference on Cloud Engineering (IC2E); 24–27 June 2019; Prague, Czech Republic. pp. 23–33.
18. Choi S, Lee S, Kim Y, et al. Serving heterogeneous machine learning models on Multi-GPU servers with spatio-temporal sharing. In: Proceedings of the 2022 USENIX Annual Technical Conference (USENIX ATC 2022); 11–13 July 2022; Carlsbad, CA, USA. pp. 199–216.
19. Kosaian J, Rashmi KV, Venkataraman S. Parity models: Erasure-coded resilience for prediction serving systems. In: Proceedings of the 27th ACM Symposium on Operating Systems Principles (SOSP 2019); 27–30 October 2019; Huntsville, ON, Canada. pp. 30–46.
20. Li J, Zhao L, Yang Y, et al. Tetris: Memory-efficient serverless inference through tensor sharing. In: Proceedings of the 2022 USENIX Annual Technical Conference (USENIX ATC 2022); 11–13 July 2022; Carlsbad, CA, USA.
21. Romero F, Li Q, Yadwadkar NJ, Kozyrakis C. INFaaS: Automated model-less inference serving. In: Proceedings of the 2021 USENIX Annual Technical Conference (USENIX ATC 2021); 14–16 July 2021. pp. 397–411.