Making LLM More Efficient by Compressing Low-rank Matrices

Pingwei Sun

psunah@connect.ust.uk

Abstract

This project aims to explore efficient compression methods by factorizing the low-rank weight matrices of LLM. The project mainly consists of the following two steps: **1)** Analyze the structure of generative large models and identify locations within the model where the many redundant parameters exist; **2)** Explore methods that satisfied the balance between accuracy and compression rate. If time permits, optimizations for HPC will also be explored.

1 Project Introduction

Since the advent of ChatGPT, large models have demonstrated remarkable capabilities across various domains. However, the extensive parameter size of these models has posed challenges in both training and deployment. This project aims to leverage the low-rank property of the weight matrices in LLM to compress the model, thereby accelerating its time performance.

The current compression strategies (Xu and McAuley, 2022) mainly fall into three directions:**(1) Knowledge Distillation,** which is commonly applied in industrial field. However, it always takes more training steps and the performance of student model highly depends on the set of hyper parameters.**(2) Pruning** can well maintain the original performance of the model, but the irregular structure may disable the accelerated optimization.**(3) Low-Rank Factorization** is a method that compresses the model and reflects the interpretability of the model to some extent. Early research has conducted experiments of a series methods, such as SVD and Kronecker. Though being a commonly used low-rank factorization strategy, SVD tends to result in poor performance when the compression rate is high. Recently, numerical optimization methods like Fisher (Hua et al., 2022) have been proposed, but they are not time efficient.

2 Target and workload

Analysis of LLM structure. At present, a series of generative large models are constructed by stacking of decoders. It is known that modules located at different depths handles different levels of knowledge. Aiming at pruning parameters without significantly drop of the accuracy, preliminary experiments should be conducted to find the bottleneck structure.

Efficient compression method. Search for decomposition methods that can achieve a trade-off between compression ratio and accuracy.

Further optimization. If time permits, the compression algorithm can also be optimized in the aspects of both logical and mathematical to increase its parallelism for HPC.

3 Grading Criteria

Considering the requirements of MSBD5014 and this research-oriented topic, the specific grading criteria consist of the following components

- **A more detailed proposal: 10%**
- **Group meetings and regular reports: 30%**
- **Final report: 60%**

References

Ting Hua, Yen-Chang Hsu, Felicity Wang, Qian Lou, Yilin Shen, and Hongxia Jin. 2022. Numerical optimizations for weighted low-rank estimation on language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1404–1416, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Canwen Xu and Julian McAuley. 2022. A survey on model compression and acceleration for pretrained language models.