Project for Team 5

Title
Training a Large Transformer Model with Limited GPU Resources

Introduction
Transformer models are widely being applied to many domains in science and society. Unfortunately, the prohibitive training costs of large Transformer models hinder their broad application. In practice, training large Transformer models with billions or even trillions of parameters can take days, weeks, or even months, depending on the available computational resources.

Objective
This project aims to cost-efficiently train the large Transformer models with limited GPU resources by appropriately combining multiple small models. This is motivated by the fact that it usually consumes a shorter time and fewer resources to train small Transformer models, even though these small models are less powerful and only suitable for a simpler language.

Methodology
The methodology for this project involves several key steps:
- Data collection: collect two English sentence datasets: the first is extracted from level-0 reading books for preschool children and the second is extracted from level-1 reading books for students in the first grade.
- Small Transformer model training: Use the first English sentence dataset to build/train one or more copies $m_1, m_2, \cdots, m_n$ of a small Transformer model with the dimension d_0 ($< d_1$).
- Large Transformer model training: Use the second English sentence dataset to train a Transformer model M_0 with the dimension d_1.
- Model combination: Design a cost-efficient method of combining well-trained small models to generate the initial weights of a large Transformer model M_1' with the dimension d_1 and then refine M_1' through training with the second English sentence dataset to generate the well-trained large Transformer model M_1.
- Performance evaluation: Compare two training methods for large Transformer model generation in prediction accuracy and training cost, and conduct the profile-based analysis on the training latency, memory usage, and GPU core consumption of training large and small Transfer models with Nsight Systems/Compute.

Expected Outcomes
Students will understand how to design, optimize, and implement LLMs targeting a specific platform. They will learn how to balance prediction accuracy and cost efficiency, which are crucial in real-world applications. Also, they will have a deep insight into the resource consumption of different models.

Reference

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention is All You Need. Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, December 2017.
[2] Wenyu Du, Tongxu Luo, Zihan Qiu, Zeyu Huang, Yikang Shen, Reynold Cheng, Yike Guo, and JieFu. Stacking Your Transformers: A Closer Look at Model Growth for Efficient LLM Pre-Training. https://arxiv.org/pdf/2405.15319
[3] Peihao Wang, Rameswar Panda, Lucas Torroba Hennigen, Philip Greengard, Leonid Karlinsky, Rogerio Feris, David D. Cox, Zhangyang Wang, and Yoon Kim. Learning to Grow Pretrained Models for Efficient Transformer Training. https://arxiv.org/pdf/2303.00980
[4] Yu Pan, Ye Yuan, Yichun Yin, Zenglin Xu, Lifeng Shang, Xin Jiang, Qun Liu. Reusing Pretrained Models by Multi-Linear Operators for Efficient Training. Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS), 2023
[5] Sheng Shen, Pete Walsh, Kurt Keutzer, Jesse Dodge, Matthew Peters, and Iz Beltagy. Staged Training for Transformer Language Models. Proceedings of the 39th International Conference on Machine Learning (ICML), 2022.
[6] Peihao Wang, Rameswar Panda, and Zhangyang Wang. Data Efficient Neural Scaling Law via Model Reusing. Proceedings of the 40th International Conference on Machine Learning (ICML), 2023.
[7] DL-GPU workshop tutorial on “CUDA Kernel Profiler and Performance Analysis Tool”

Project presentation: 30-minute team presentation on Friday, June 7th, 2024
Paper draft completion deadline: August 2nd, 2024

Paper draft: a 10-page paper including references in the IEEE 2-column format
Paper format should follow IEEE Computer Society Proceedings Manuscript Formatting Guidelines (see the link to "formatting instructions" below).
https://www.ieee.org/conferences/publishing/templates.html