Identifying computation and communication patterns of MLPerf Training workloads

Saheli Bhattacharjee, Anton Lokhmotov

Abstract

The MLPerf benchmark, established in 2018, has become an industry-standard suite of machine learning (ML) workloads for evaluating the performance of ML systems across training and inference tasks. Given its prominence, vendors invest considerable effort in developing highly optimized implementations to maximize performance. However, understanding the underlying computation and communication patterns of these workloads remains a critical aspect of improving ML system efficiency.

In this presentation, we introduce our early work on identifying the computation and communication patterns of optimized MLPerf Training workloads. We recognise that existing implementation choices are largely dictated by current system capabilities and constraints, particularly in networking and parallelism. As system architectures evolve, revisiting these patterns will be crucial to achieving optimal performance.

Our approach involves building a suite of tools to capture traces of such workloads e.g. tensor shapes involved in computation and primitives involved in communication. Captured traces can be used as input to simulation tools. First, to calibrate existing simulation frameworks to ensure they accurately reflect the performance of today's systems; and second, to explore "what-if" scenarios that guide system design at the server-, rack- and datacentre- levels.

Our findings reveal that MLPerf Training workloads exhibit patterns similar to those of Berkeley dwarfs—a class of stencil computations that are known to present challenges in efficient parallelization. We analyze these patterns through detailed profiling on both GPU and TPU platforms, uncovering distinct trade-offs between computation and communication efficiency. Our profiling results indicate that:

- GPUs demonstrate higher efficiency for data-parallel workloads due to their superior memory bandwidth and parallel execution capabilities.
- TPUs outperform GPUs for model-parallel workloads, benefiting from their specialized hardware designed for tensor processing.
- Both GPUs and TPUs encounter inefficiencies in communication-heavy workloads, highlighting the need for advancements in interconnect technology and communication-aware scheduling strategies.

By understanding these workload characteristics, we can inform the design of next-generation ML hardware and software ecosystems. Our work provides actionable insights into optimizing parallelism strategies, improving interconnect architectures, and refining workload scheduling to enhance overall ML training performance.

This presentation will showcase our methodology, key findings, and their implications for the broader ML systems community. By characterising the computational and communication properties of MLPerf Training workloads, we aim to contribute to the evolution of efficient ML system architectures. Our work has broad implications for hardware designers, ML researchers, and system architects seeking to push the boundaries of scalable and high-performance machine learning training.