Fan Zhang

zfan38551@gmail.com · https://github.com/zfan2356

Education

B.S. in Software Engineering at **Northeastern University**, NE, China Sep, 2021 – Jun, 2025

Selected courses: Operating Systems, Computer Networks, Computer Architecture

Work Experience

Tencent, WXG WeChat Technology Group Dec, 2024 – Present

Technology Architecture Department, Pre-training/Post-training Engineer of Base Model

- Responsible for the framework development of pre-training/post-training of base model in WeChat, including model training acceleration and memory optimization.
- Participated in the reimplementation of DeepSeek Infra, using ThunderKittens to refactor DeepGEMM, and responsible for the development of **80%** of Kernels in the framework.

Related Projects

Distributed Pre-training Engine for Large Language Models Dec, 2024 – Present

A large-scale pre-training engine developed for WeChat's next-generation foundation model WELM, supporting multi-node multi-GPU training of trillion-parameter language models with industry-leading distributed training capabilities, perfectly adapted to the WeChat ecosystem.

- **High-Performance Parallelism**: Implements 3D parallelism (TP+DP+PP) and ZeRO for dense models, with EP parallelism support for MoE models, enabling distributed training across multiple machines and GPUs.
- **Customized Kernels**: Integrated forward and backward computation for most operators using Triton/ThunderKittens, with multiple operator fusion implementations to optimize memory usage and accelerate training efficiency.
- **FP8 Precision Support**: Supports FP8 low-precision training with flexible precision selection through Quant/Dequant operations for efficient low-precision training.
- **Cutting-edge Technology Integration**: Supports Dualpipe/DeepEP to achieve computation and communication overlap.

RDMA Server Dec, 2024 – Present

RDMA Server is a parameter distribution engine used by Pre-Train/Post-Train frameworks for efficient parameter distribution. Based on InfiniBand RDMA for high-performance reading, supporting multi-node multi-GPU parameter synchronization and forwarding, it is a Linux high-performance communication engine for large model distributed training.

- **High-Performance Communication**: Based on InfiniBand, uses RDMA to optimize communication. GPUs can read memory data from other GPUs bypassing CPU, avoiding CPU intervention and achieving high-performance communication. Also provides UDP communication support.
- **Python Binding**: Uses PyBind11 to wrap C++ code, providing easy-to-call Python interfaces.
- **Asynchronous Support**: Implemented stackless coroutine model based on C++20 coroutine standard, transforming asynchronous I/O operations into sequential awaitable expressions through co_await suspension points. While maintaining event-driven high performance, eliminates callbacks through compile-time state machines, achieving synchronous semantics for asynchronous execution flow.
- **High-Performance Logging**: Lock-free queue asynchronous batch log processing combined with thread-local buffering and double-buffering mechanisms. Achieves zero dynamic memory allocation based on memory pre-allocation, coupled with SSD-optimized sequential write strategy and real-time log compression, delivering tens of millions of log writes per second while maintaining nanosecond-level latency.

Skills

- **ACM** (Association for Computing Machinery), member since Dec, 2023
- **Programming Languages**: multilingual (not limited to any specific language), especially experienced in C/C++, Python, Go, comfortable with Java, Rust (in random order).
- **LLM Algorithms**: Proficient in large model pre-training algorithms, familiar with cutting-edge Dense and MoE technologies. Deep understanding of model architectures including Llama, DeepSeek, Qwen, and comprehension of principles and designs of components like Rotary, RMSNorm, Swiglu. Experienced in large model post-training algorithms, familiar with principles and designs of DPO, PPO, GRPO.
- **Proficient in LLM Infrastructure**: Have studied source code of frameworks including Megatron-LM, DeepSpeed, FSDP, Verl, understanding their principles and designs. Familiar with distributed computing/communication libraries such as Ray, NCCL, and proficient in PyTorch.
- **System & Hardware**: Understanding of Linux low-level principles, familiar with GPU/CPU hardware knowledge, proficient in X86-64, ARM64, RISC-V architecture knowledge, skilled in NVIDIA GPU underlying architecture and PTX assembly, proficient in writing CUDA-related code.

Misc

- Personal blog: ai-router, recording learning experiences and technical sharing.
- Languages: English - fluent, Chinese - native speaker