许宏鑫

191-2921-2198 | xuhx56@mail2.sysu.edu.cn | xhx1022.github.io

教育经历

中山大学 | 计算机科学与技术，计算机学院 | 学术型硕士研究生 2024.09—2027.06（预计）

导师张献伟，主要研究方向为 **MLSys**，在大模型推理系统方面有一定的研究和工程经验。

主要研究成果为：在投 A 会两篇，一篇一作，一篇二作

华南理工大学 | 计算机科学与技术，计算机学院 | 工学学士 2020.09—2024.06

GPA: 3.84/4.0(专业前 10%)，获校级学业奖学金、企业奖学金多次，并保研至中山大学。

技术能力

- **编程语言**: Python,C++,Shell
- **工具**: Linux, Git, Docker, Nsight System
- **技术栈**: 有大模型推理框架的实践经验，熟悉 Pytorch、SGLang、vLLM。

项目经历

基于动态层重分配的 LLM 高效流水线并行服务系统 | 在投论文一作 2025.03—2025.05

该项目聚焦于大模型推理中流水线并行的 **inter-stage** 不平衡问题，系统通过实时预测计算与采样延迟，动态调整各阶段的层分配，有效缓解因尾部阶段采样开销造成的流水线气泡与阶段失衡，显著提升硬件利用率。在多种负载下，端到端推理延迟降低了 10% 至 49%，优于现有主流推理框架。其核心机制包括：(1) **延迟预测器**：实时根据负载预测前向计算和采样开销，用于调度器决策；(2) **气泡感知调度器**：根据阶段执行时间差异自适应调整层分配，打破传统平均分配策略，缓解流水线气泡现象；(3) **异步 KV 缓存迁移机制**：支持推理过程中的非阻塞重分配，保持流水线运行连续性。

自研高效大模型流水线推理框架 | 开发者之一 2024.06—2025.01

参与开发针对流水线并行的高效大模型推理系统，支持多种主流开源模型，集成了 **PagedAttention**、**Chunk Prefill**、**Prefix Caching** 等优化技术；同时构建了**异步运行时系统**，采用多进程架构与非阻塞通信机制，通过预调度元数据与激活值传输解耦，有效减少主控进程的调度开销与 CPU 资源占用；此外提出 **Token Throttling 调节机制**，可根据实时请求量与 KV 缓存压力动态调整 prefill 与 decode 阶段的 token 数量，平衡各批次之间的计算负载，显著减少 **inter-batch** 流水线气泡。

基于 SLO 满足率的混合负载调度优化 | 独立实现 2025.01—2025.02

针对大模型推理中输入输出分布高度多样化的混合负载场景，独立设计并实现基于 SLO 满足率的调度优化机制，主要包括：(1) 长度感知优先级排序：根据请求的输入输出长度，按去除 prefix 后的实际 prefill 长度从小到大排序，在保证长请求 SLO 的前提下，优先调度短 prefill 请求，使更多 decode 请求与 prefill 请求能够并行混合执行，从而提升整体吞吐；(2) 窗口调度机制：引入调度窗口，仅对窗口内请求进行排序，防止长请求长期被延后调度，兼顾响应公平性与效率；(3) 动态调整 chunk 大小：根据当前系统负载与 SLO 限制动态调整执行执行粒度，进一步优化资源利用与响应性能。

基于 GPU 空间-时间协同编排的高效大模型推理系统 | 在投论文二作 2024.11—2025.03

该系统针对 prefill 和 decode 阶段在 GPU 上资源使用不均的问题，提出了空间-时间协同编排方案，使两阶段能够并发执行，并根据实际需求动态划分 GPU 的计算资源。设计并实现一种针对大模型推理中 prefill 与 decode 阶段资源使用不均的问题的空间-时间协同编排机制，使两个阶段能在 GPU 上并发运行，并根据负载动态调整计算资源分配。系统识别并解决了 prefill 阶段中由于 wave quantization 与注意力机制瓶颈导致的低计算利用率问题，以及 chunked prefill 策略中因延迟优先引发的吞吐下降和资源浪费问题。提出即时资源重配置方案，预配置多种 SM 分区策略，支持毫秒级切换；并通过 GPU 映射实现 KV cache 零拷贝跨进程共享，进一步降低通信开销与状态迁移成本。

多智能体间 KV Cache 复用优化 | 独立实现 2024.09—2024.11

在多智能体系统中，一个智能体的输入往往包含其他智能体的输出，当前主流方法需重新执行 Prefill 阶段，导致计算冗余与推理延迟显著增加。针对这一问题，提出并实现了一种基于部分重计算的 KV Cache 复用优化策略。核心思路是，注意到注意力机制中仅部分 Token 在部分层中产生显著的交叉注意力，因此设计了 “逐层筛选 + 选择性重计算” 的方法，在每一层动态评估交叉注意力分数，仅对关键 Token 在关键层进行重计算，其余部分复用原有的 KV Cache，从而实现在精度不下降的情况下，智能体间能够共享 KV Cache，大幅减少重复计算开销。

YatCC-AI: 中山大学智能编译教学平台 | 助教 2025.02—至今

作为核心助教，负责模型部署与平台服务的技术支持与优化：(1) 在春节后第一时间完成 DeepSeek-R1 模型上线，确保学生教学场景可直接调用；(2) 搭建 Prometheus+Grafana 监控系统，实现模型运行状态与资源使用的实时可视化；(3) 构建基于大模型的 RAGFlow 检索增强生成系统，用于帮助同学快速检索编译课程实验文档。