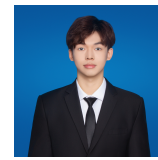


# 张浩东 (Haodong Zhang)

邮箱: 1010761880@qq.com | 电话: 13148796380 | GitHub: holeeast979.github.io



## 教育经历

香港中文大学 (CUHK)	人工智能 (MSc in AI)	2026 秋季入学 (已获 Offer)
暨南大学 (Jinan University)	软件工程	学士
		2022.09 - 2026.06

- GPA: 3.85/5.0 (专业排名前 5%)
- 核心课程: 强化学习与最优控制 (97)、机器学习 (96)、人工智能原理 (92)、算法分析与设计 (90)

## 专业技能

- 编程语言: Python (精通), C/C++, Java, SQL AI 框架: PyTorch, LangChain, LangGraph, Hugging Face
- LLMOps & 工具: Docker, Langfuse, Prometheus, Grafana, RocketMQ, Git, Linux
- 核心能力: LLM 应用开发 (RAG/Agent), 多模态模型 (LMM) 架构与性能分析 (Profiling)

## 荣誉奖项

- 国家奖学金 (National Scholarship): 本科生最高荣誉, 全校前 1% | 暨南大学, 2025
- APMCM 亚太地区大学生数学建模竞赛优胜奖: 核心队员, 负责大模型相关算法设计与实现 | 2025
- 蓝桥杯全国软件大赛省级二等奖: Python 程序设计 A 组, 算法与数据结构竞赛 | 2024
- 暨南大学优秀学生二等奖学金: 连续两年获奖 (2022-2024)

## 实习经历

十方融海科技有限公司	AI 平台开发实习生 (LLMOps 方向)	2025.09 - 2025.12
<ul style="list-style-type: none"><li>核心职责: 负责企业级多智能体编排平台的 LLMOps 基础设施建设, 支撑下游业务的高并发调用。</li><li>Agent 评测流水线: 设计并容器化部署基于 RocketMQ 的高并发评测套件, 开发 Mock Server 模拟海量用户请求, 实现 Agent 系统的端到端压力测试; 量化系统在不同并发下的 TPS 与 Latency 瓶颈, 为模型上线提供性能准入标准。</li><li>全链路可观测性: 主导 Langfuse 在生产环境的深度集成, 将 Tracing 植入 LangGraph 工作流, 实现从用户 Query 到 LLM Token 生成的全链路透视; 解决多智能体协作中的“黑盒”调试难题, Agent 错误归因效率提升 50%+。</li><li>系统监控: 基于 Prometheus + Grafana 构建实时监控看板, 定义“队列积压率”、“消费延迟”等关键指标, 实现异常流量毫秒级预警。</li><li>技术栈: Python, LangGraph, Docker, RocketMQ, Prometheus, Grafana, Langfuse</li></ul>		

## 项目经历

端侧多模态大模型 (LMM) 推理加速研究	独立负责人	2024.10 - 至今
<ul style="list-style-type: none"><li>项目背景: 针对 LMM 在边缘设备 (Jetson Orin) 上显存占用大、推理延迟高的问题, 开展基于异构计算的推理加速研究。</li><li>瓶颈定位: 开发基于 PyTorch Hooks 的非侵入式耗时分析工具, 实验发现 ViT Encoder 占据 92%-94% 的推理延迟, 推翻了“视频解码是主要瓶颈 (仅占 18%)”的传统假设, 明确了优化重点。</li><li>架构对比: 验证 Q-Former 架构在视频推理上的优势, 通过 Token 压缩机制将视频推理速度提升 9 倍 (923ms -&gt; 105ms)。</li><li>采样策略: 揭示端侧设备的“边际效应递减”现象——帧数翻倍导致延迟增加 77% 但精度收益微乎其微, 确立以动态关键帧提取替代均匀采样的路线。</li><li>后续计划: 研究 FlashAttention 端侧适配及 Token Pruning 算法, 目标推理速度提升 40%+。</li><li>技术栈: PyTorch, Hooks, Profiling (Nsight Systems), Transformers, ViT</li></ul>		
高质量 RAG 基准数据集构建与评估	核心开发者	2025.06 - 2025.08
<ul style="list-style-type: none"><li>数据工程: 针对 Natural Questions 数据集, 设计并实现一套基于 DeepSeek API 的自动化数据增强流水线, 成功将“黄金段落”数量从 13,419 扩充至 42,205 个 (+214%), 显著增强数据集对长尾知识的覆盖度。</li><li>检索优化: 提出并实现 Re-ranking 优化算法, 在基准测试中将 Retrieval Recall@10 提升 3-5 个百分点。</li><li>技术栈: Python, DeepSeek API, Milvus, Spacy, Data Processing</li></ul>		