雷记新 - AI 应用开发

教育背景：重庆邮电大学-软件工程　学历：本科-2026 届

电话：15126251889　　邮箱：2452143632@qq.com

荣誉奖项

- 全国大学生数学建模竞赛国家级二等奖（2023）
- 国家励志奖学金和校级奖学金

专业技能

- 了解大模型工作原理、主流模型对比及定制化三种方式,了解 Agent 本质——从工具调用到自主决策的推理机制
- 掌握 Prompt 五要素设计，熟悉 CoT、ReAct、Few-Shot 等提示策略；有 System Prompt 结构化设计经验，能通过工具描述语义边界调优提升 Agent 工具选择准确率
- 了解 Spring AI 核心模块(ChatClient、Advisor 机制、RAG 流程),能使用 Spring AI 构建问答应用和 RAG 检索增强系统
- 掌握 LangChain4j 核心机制，能构建 AI Agent；熟悉 @Tool 工具定义、ChatMemory 多轮对话管理、TokenStream 流式输出，了解 ReAct 执行循环控制
- 了解 RAG 整体架构 (文档加载 → 切块 → 向量化 → 检索 → 生成),了解 Embedding 模型选型、向量数据库选型、TopK 与相似度阈值调优,了解混合检索、RRF 融合排序等进阶方案；
- 了解 MCP 协议架构，能开发 MCP Tools Server 了解 A2A 协议与 AgentSkill 概念，了解多框架 (LangChain4j / Spring AI / AgentScope）生态互通方式
- 熟悉使用 Claude/Codex Agent 模式辅助项目搭建、代码重构、单元测试生成；了解 AI 辅助开发的最佳实践与边界；
- 了解 AI 应用延迟分析、Token 成本控制、并发限流与 API 配额管理；了解结构化日志指标采集；

实习经历

上海捷羿软件系统有限公司-技术研发部-AI 应用后端开发工程师　　**2026.02-至今**

AI 融合开发平台

AI 融合开发平台是一个集需求智能梳理、交互优化、PRD 原型生成于一体的高效开发辅助平台，专为产品研发团队、中小企业设计，简化需求梳理与原型制作流程，降低产品研发前期沟通成本，提升需求落地效率。平台核心支持用户需求与 AI 交互迭代，最终生成可打包导出的 PRD 原型，目前已全面接入主流国产 AI 模型，适配多场景需求开发。

技术栈：SpringBoot、MybatisPlus、Mysql、Redis、国产 AI 模型 API、FastAPI、MinIO、Vue3（后端适配）

主要负责 AI 交互与原型生成模块：

- 负责用户需求与国产 AI 模型的交互对接，设计并实现需求提交、AI 智能梳理、用户二次编辑的全流程接口，支持用户与 AI 实时交互调整需求，确保需求梳理的准确性与高效性
- 基于梳理确认后的需求，设计 PRD 原型自动生成逻辑，集成原型模板引擎，实现需求到 PRD 原型的一键生成，支持原型在线预览、编辑与导出打包，将原型制作时间从 2 天优化至 2 小时
- 基于 Spring AI 框架集成各类国产 AI 模型，封装统一的模型调用接口，简化模型接入流程，兼容不同国产模型的请求格式，优化接口调用超时、重试机制，提升 AI 交互响应速度与系统稳定性，保障多用户并发使用场景下的流畅性

上海明奇网络科技有限公司-技术研发部-后端研发实习工程师　　**2025.07-2025.10**

惠服 IT 智能系统

惠服 IT 智能系统是一个微服务架构的企业级 IT 服务管理平台，专为 IT 服务外包公司、连锁企业、大型集团等组织设计。提供从工单创建、派发、处理到关闭的完整解决方案，同时集成客户管理、供应商协调、数据分析等核心功能。

技术栈：SpringBoot、SpringCloud、Mysql、Redis、MybatisPlus、RocketMQ、阿里云 OSS

主要负责工单模块：

- 基于 **Redis 分布式锁**的工单创建防重复提交机制，通过参数+用户标识生成唯一锁 Key，有效防止重复工单产生
- 通过自定义**线程池+CountDownLatch** 并发调用多个微服务 API，将 10 万条工单统计时间从 15 分钟优化至 3 分钟，性能提升 5 倍，显著改善用户体验
- 设计并实现了基于 RocketMQ 的**异步消息推送**系统，支持微信公众号、企业微信、短信等通知类型，通过异步消息队列实现高并发推送，显著提升用户体验和系统稳定性

项目经验

智能销售数据分析 Agent **大模型应用开发** 2026.01-2026.03

项目描述

基于 AI Agent 构建的销售数据分析系统，目标是用自然语言对话替代传统"一个需求一个接口"的报表开发模式。用户直接用中文提问，Agent 自主决策调用哪些工具、调用几次，将复杂查询意图映射到有限的工具能力集合，生成包含数据和分析的完整回答。项目部署在学院服务器，供组内同学试用和测试。

技术架构

SpringAI、Langchain4j、Mysql、Redis、Guava、prompt、memory

项目亮点

- 基于 LangChain4j，设计覆盖查询/统计/趋势/图表/异常预警五类场景的工具集，Agent 通过 ReAct 循环自主规划调用链路，单次提问最多串联 4 步工具调用完成复杂分析任务，全程无硬编码流程控制
- 设计结构化 Prompt 工程体系，工具描述采用"正向能力 + 反向排除"结构，消除相似工具语义混淆；System Prompt 注入当天日期解决模型时间感知盲区，声明能力边界防止越界推理；
- LLM 友好型工具输出设计，工具返回值统一格式化为自然语言，降低模型推理认知负担，提升多步推理答案质量；图表工具通过 CHART_JSON: 前缀内嵌协议，前端按标记分流渲染，无需独立接口
- 实现持久化多轮对话记忆 + Token 成本三重控制，实现多 Session 隔离，限定 20 条上下文 + Redis 缓存复用热点查询 +System Prompt 控制在 200 Token 以内，三重策略叠加将 Token 日消耗控制在预算内，并实时监控
- 数据权限从 Prompt 约束升级为代码硬保证，依赖 System Prompt 声明"只查自己数据"是不够的，模型行为不可完全预测，Prompt 约束在推理链路中可能被忽视。项目采用 ThreadLocal 传递用户角色信息，与模型推理过程完全解耦
- 用 Micrometer 在工具调用前后埋点，分别记录每次请求的 Input Token 和 Output Token，通过分析发现 System Prompt 和工具描述合计占 Input Token 的 60% 以上，由此驱动 System Prompt 压缩和工具描述精简，将固定成本降低约 30%

随享听书 **java 项目开发** 2025.05-2025.06

项目描述

随享听书是一个有声书音频分享平台，用户可以在平台创建书籍音频，随时分享好声音。项目模块主要包括：登录模块、账户模块、后台管理模块、专辑详情模块、订单模块、搜索模块、支付模块

技术栈：SpringBoot、SpirngCloud、Redis、MySQL、Mybatis、Rabbitmq、Minio、Cannal、xxl-Job

项目亮点

- 采用 JWT 双 token 机制避免登录频繁过期问题，结合 SpringAOP 实现接口权限统一校验，通过 **ThreadLocal** 传递用户上下文实现请求链路身份共享
- 使用**自定义线程池+CompletableFuture 异步编排**+双缓存架构优化专辑详情信息页面，响应时间从 1249ms 提升到 5ms
- 使用**分布式锁+lua 脚本**避免专辑详情页缓存击穿问题，使用布隆过滤器防止缓存穿透问题
- 使用 Canal 订阅 BInlog 日志异步删除缓存 + 消息队列重试机制，**解决缓存不一致问题**以保障数据可靠性
- 使用 **RabbitMQ+死信队列**实现超时自动关闭订单
- 使用 Seata 中的 **AT 模式**解决账户微服务和订单微服务的数据强一致性