林翰平

求职意向：大模型算法工程师 | 成都 | 18k-22k | 一个月内到岗

年　　龄：23岁　　性　　别：男

工作年限：3年经验　　电　　话：13219318632

邮　　箱：woshijilin@foxmail.com

教育背景

2020-06 ~ 2024-09　　**西南石油大学**　　**软件工程（本科）**

主修课程:

- 计算机科学基础：数据结构与算法、操作系统、计算机组成原理、计算机网络
- 软件开发技术：面向对象程序设计、软件架构设计、设计模式、敏捷开发
- 系统与安全：数据库系统、分布式系统、软件安全与测试
- 应用开发：Web开发（前端/后端）、移动应用开发、云计算与微服务
- 工程管理：软件项目管理、软件需求工程、软件质量保证

外语水平:

英语: CET-4

毕业论文:

《基于大模型的NLP情感分析系统》

工作经验

2024-04 ~ 至今　　**北京领为军融科技有限公司成都分公司**　　**中级算法工程师**

- 负责大模型应用设计和开发，基于 vLLM 部署企业内大模型推理服务，支持 Qwen、DeepSeek 等主流模型的统一接入与动态调度。
- 主导文档智能生成系统的全栈开发，基于 LangChain 构建生成流程，结合 Milvus 实现知识检索与上下文增强，支持模板化、自动化文档输出。
- 推动领域知识的结构化沉淀，构建领域知识图谱，并设计 AI 智能辅导系统后端架构，融合大模型能力提供个性化学习支持。

2023-03 ~ 2023-09　　**成都云徙科技**　　**中级开发工程师**

- 前端组件库开发：参与公司内部 React 组件库的封装与维护，提升代码复用率，优化团队开发效率。
- ToB 商城全栈开发：

  负责 珠江啤酒、SKG 等企业级电商平台的前后端开发，采用 React + Ant Design 构建前端界面，使用springboot 开发后端接口。

  实现响应式布局、动态路由、状态管理（Redux/MobX），确保高性能用户体验。
- 工程化与部署：

  参与 CI/CD 流程搭建，基于 Docker 实现阿里云自动化部署。

  熟悉 Git 协作开发，遵循 Agile/Scrum 流程，提升团队协作效率。

项目经验

2025-04 ~ 至今　　**K12 教育AI错题引导系统**　　**算法设计、后端开发**

- 项目背景

  作为部门在 AI 教育方向的创新尝试，目标是探索 AI 在初升高衔接阶段的个性化教学能力。学生使用流程为：① 学习知识点；② 获得推送题目；③ 手写答题，错题进入 AI 错题引导，直至掌握。

- 核心设计

AI 老师模块是核心，包含三大功能：

a. 个性化管理：根据学生水平动态调整讲题语气、题目难度；
b. 手写内容识别：支持公式、草图等复杂输入；
c. 答题分析与错误引导：判断正误并提供针对性讲解。

前端采用 React，后端采用 Go，大模型选用 Gemini-2.5-pro 与 Qwen-Max，实现高质量教学交互。

- 我负责的工作
- 我负责 AI 老师模块的完整设计与开发：
- 设计个性化教学逻辑与难度调控机制
- 实现手写内容识别与结构化解析
- 构建错题分析流程与引导策略
- 完成后端 API 与模型服务集成
- 技术亮点
- 项目验证了 AI 在教育中“因材施教”的可行性，为后续智能教学产品积累了技术经验与用户反馈。

2025-02 ~ 2025-04　　医疗病历OCR手写识别系统　　算法设计、架构

- 项目背景
- 与医疗部门合作，旨在解决传统 OCR 在病历归档中识别效果差、归档流程复杂、缺乏专业语义理解等问题。目标是实现手写病历的高效、准确、结构化电子化。
- 核心设计
- 我设计了一套融合 OCR 与大模型的多模态识别方案：

a. 使用微调后的百度飞桨 Handwriting v1 模型进行初步文本识别；
b. 同步调用 Qwen2.5-VL 多模态模型，结合病历来源、科室、疾病类型等属性生成上下文提示词，提升语义理解能力；
1. 通过 reasoning 模型（如 DeepSeek-R1）对两路识别结果进行合并分析，输出最终电子病历表与结构化 metadata。
a. 系统还集成“文转表”功能，实现批量病历自动归档。

- 我负责的工作
- 我主导整体方案设计与后端架构搭建：
- 完成技术选型与多轮实验验证，确定最优融合策略
- 设计系统框架并交由实习生补全代码
- 亲自审阅所有核心模块代码
- 负责部署至阿里云服务器，并与医疗部门完成交接
- 技术亮点
- 项目上线后大幅提升病历归档效率与准确性，获得医疗部及合作医院高度评价，成为标准化归档工具。

2024-08 ~ 2025-01　　内网大模型文档生成系统　　算法设计、开发

- 项目背景
- 公司涉及军方合作项目，部分资源仅限内网访问，导致无法使用外部大模型服务。项目面临三大挑战：① 需在内网部署大模型；② 需构建专属知识库支持文档生成；③ 需支持知识库自动更新。项目由我与一位资深算法工程师共同负责。
- 核心设计
- 申请硬件采购并部署 4 张 A100 显卡（3 张用于大模型服务）。选用 Qwen2.5-72B 大模型，基于 vLLM 框架部署，实现高性能推理。前后端采用 React + FastAPI 快速搭建。知识库采用 Milvus + MySQL 架构：对已有文档抽取 metadata 存入 MySQL，embedding 向量存入 Milvus。
- 文档生成流程为：输入主题、背景信息、参考信息 → 生成初步大纲 → 用户可在前端交互式修改大纲 → 大纲存入大纲库（支持复用）→ 完整大纲进入生成模块。生成支持两种模式：① 按大章节快速生成；② 按子章节详细生成。结合 RAG 策略，在生成过程中捕获关键信息（主旨+关键词），检索知识库中 TOP3 参考文档进行增强生成。最终文档支持在线 AI 辅助编辑（集成文转图、文转表、润色、翻译、语气更改等功能），并可导出为 Markdown 或 DOCX 格式。

我负责的工作

- 我主导前后端开发与 AI 模块的完整实现：
- 负责 FastAPI 后端服务设计与 API 开发
- 搭建 Milvus + MySQL 知识库，实现文档向量化与元数据管理
- 基于 LangChain 进行提示词管理与生成管线编排
- 二次开发 CKEditor，集成 AI 编辑功能
- 选用 Qwen2.5-72B 与 DeepSeek-R1 reasoning 模型，优化生成质量
- 技术亮点
- 项目历时 6 个月完成，成功在内网环境下实现文档智能生成，满足军工项目对数据安全与合规性的高要求，已稳定服务于多个项目团队。

2024-04 ~ 2024-07　　CodeGen AI 代码生成项目　　算法设计、开发

- 项目背景
- 针对公司内部控制系统频繁迭代、开发成本高的问题，团队启动 AI 驱动的低代码平台项目，目标是让非技术人员也能参与系统构建。传统低代码平台依赖预设组件，难以应对复杂业务逻辑，而大模型的引入为“从逻辑到代码”的自动化提供了新路径。
- 核心设计
- 采用 Next.js 全栈架构作为模板，设计端到端生成管线。首先对公司现有系统进行解构分析，抽象出 BO（Business Object）与 DO（Data Object）的概念。业务人员通过前端拖拽方式描述业务逻辑（生成伪代码），构建出 BO；中台大模型基于 BO 分析数据结构，自动设计 DO 与数据库表关系；再由大模型根据页面逻辑分块生成 React 组件，最后通过前端专家模型完成组件拼接与适配性检查，输出可运行的全栈页面。
- 我负责的工作
- 我负责大模型在 BO-DO 映射环节的核心设计与实现，包括：
- 设计从客户端输入到 AI 处理再到数据库建模的数据桥接流程
- 构建生成管线（Pipeline）的编排逻辑
- 编写并持续调优 Prompt，确保大模型能准确理解业务语义并生成合理的数据结构
- 设计大模型二次检查机制，提升生成结果的稳定性与一致性
- 技术亮点
- 该系统区别于传统低代码平台，业务人员只需关注业务流程本身，只要逻辑正确，大模型生成的代码即可符合预期，显著降低了开发与沟通成本，实现了“业务即代码”的闭环。

自我评价

我是一名具备强工程落地能力的大模型应用工程师，专注于 **AI 与业务场景的深度融合** 。在多个从 0 到 1 的 AI 项目中，我始终以解决实际问题为导向，不仅关注技术实现，更重视对业务需求的理解与价值闭环。

我具备扎实的全栈开发能力，熟悉 React、Next.js、Go、FastAPI 等前后端技术，能够独立完成系统架构设计与开发落地。在大模型方向，我深入实践了 **vLLM 部署、LangChain 管线编排、RAG 增强生成、多模态识别、Prompt 工程优化** 等核心技术，具备将大模型集成到复杂系统中的实战经验。

我参与过多个高复杂度项目，如 AI 代码生成系统、内网文档智能生成平台、医疗病历手写识别系统等，涵盖军工、医疗、教育等多个领域。在项目中，我不仅负责关键技术模块的设计与实现，更注重从 **业务痛点出发** ，设计符合用户习惯的技术方案，推动 AI 能力真正落地并产生价值。

我具备良好的系统思维与跨团队协作能力，能够与算法、产品、业务方高效沟通，将模糊需求转化为清晰技术路径。同时，我持续关注大模型领域的前沿进展，具备快速学习与技术选型能力，致力于构建**可维护、可扩展、可解释的 AI 应用系统** 。