Hyacehila

教育背景

武汉大学

西安电子科技大学

AI Agent 研发工程师（实习） 2026.5 – 至今

网易互娱（上海）

- **游戏 UI 生产链路 Agent**：围绕策划 GUI、UIP 与程序开发链路，研究从 Figma/PSD 设计稿生成 NeoX 工程侧.uiprefab 文件 (类似.prefab .csd) 的 Agent 工具；通过多模态嵌入检索、项目模板库与开发规范，为模型注入组件结构、资源绑定、命名规则和工程合法性约束，使其在既有工程习惯内完成 UI 还原与 UIP 接入，并持续攻关状态组合、动效一致性与复杂界面连贯性问题。
- **UI 程序自动化与真实反馈闭环**：开发面向 NeoX 引擎与内部 UI 编辑器的 Agent Skill，接入项目知识、服务端与客户端 MCP，将程序开发从一次性代码生成改造为“生成–运行–观察–修复”的多轮闭环；Agent 可通过模拟点击和真实界面反馈定位 UI 逻辑、资源绑定与交互问题，使策划能够完成在不阅读代码的情况下完成初步 UI 程序开发，程序侧主要负责代码检查与最终合入把关。
- **项目级知识库与智能问答系统**：针对自研引擎及 UI 编辑器长期迭代产生的文档滞后与隐性知识流失问题，设计并落地基于 LLM Wiki 与 Agentic RAG 的自我演化知识库方案。将零散分布与口头传递的项目经验转化为可检索、可追溯的动态上下文系统，大幅降低新人 Onboarding 门槛及项目组内部技术沟通成本。

算法研究员（实习） 2025.12 – 2026.3

绿盟科技（武汉）

- **漏洞挖掘 Agent 与 CodeQL 验证闭环**：构建面向代码漏洞挖掘的 Single Agent / Harness，将漏洞情报检索、源码定位、污点流建模、CodeQL 查询生成与引擎验证组织为可回溯的多轮分析闭环；重点设计状态表达、工具调用协议和验证反馈，而非预设复杂角色分工，使模型能够围绕候选 source/sink、失败路径、工具输出和验证结果持续迭代，缓解长链路分析中的路径依赖和伪阳性结论。CodeQL 污点分析结果作为外部 verifier 提供最终反馈。
- **训练数据与 Agent 轨迹沉淀**：基于 GitHub 开源项目与内部数据库构建漏洞数据清洗、实体校验和标注流程，提取并校验 8,000+ CVE 实体，清洗约 4,000 条高质量样本，完成 CWE/OWASP 多维标注与语言/漏洞类型配比控制；从 Agent 执行过程中沉淀 2,500 条高置信 Tool-use SFT 轨迹和 300 条经验证污点流数据，用于冷启动、评测集构建、奖励设计与后续 RL 训练探索。

项目经历

可信舆情分析多 Agent 系统

国家自然科学基金原创探索计划项目

- **多 Agent 运行时与白盒监控**：基于 PocketFlow 设计数据准备、协同分析、报告生成三阶段运行架构，通过 MCP 将统计分析工具统一封装为标准接口，降低 Agent 分析链路中工具调用分散、扩展困难与状态难审计的问题；同时开发 Streamlit 白盒化前端，暴露步骤级 trace、tool I/O 与节点调度路径，实现 Agent 分析全流程可观测，将复杂任务的异常定位时间从数小时压缩至 10 分钟。

- **协同编排与冲突裁决**：围绕开放域舆情分析中单 Agent 结论片面、证据来源单一的问题，设计 ForumHost 作为显式调度中枢，统一协调 DataAgent 基于标注数据与量化工具执行统计分析，SearchAgent 基于 Tavily 补充实时事实与上下文；通过多轮补证、交叉质疑、冲突检测与停止条件，将多 Agent 协作从自由对话收敛为可控工作流，缓解长链路任务中的上下文污染与目标偏移。在代表性事件对照验证中，相较单 Agent（仅 DataAgent）基线，关键事实错误率由约 20% 降至 0，最终报告的事实部分无需再人工修订。
- **结构化报告与引用约束**：针对长链路报告生成中常见的“雪球式幻觉”、自由扩写与引用漂移问题，将结构化输出 Schema、内联引用、来源回溯与可靠性标注嵌入逐章生成流程，要求核心结论必须绑定证据、来源与置信度，并引入核查 Agent 对关键结论及溯源证据进行二次校验；最终报告引用覆盖率达到 95%，错误引用率降低至 3%。
- **高并发数据处理管线**：面向海量无序原始博文的预处理难题，基于 Asyncio 构建高并发清洗与语义标注流水线，结合 few-shot 提示驱动 LLM 节点完成文本清洗、标签补全与语义标准化，并通过失败重试、断点续跑与并发控制缓解 API 限流瓶颈，为后续 Agent 分析提供稳定数据底座；在 40–60 并发约束下，将 2 万条数据的处理时间压缩至 12 小时以内，并通过缓存 few-shot 示例与系统提示，将输入 token 成本降低约 60%。

Novel Evaluation：基于 LLM Rubric 的小说结构化评测工具

个人研究项目 · *LLM as Judge / LLM Evaluation*

- **项目背景**：面向小说质量评测这一高度主观且难以标准化的问题，项目尝试引入 LLM Rubric / LLM as Judge，将原本依赖经验与直觉的文本判断拆解为多个可解释维度，对其进行分层评估与量化表达。希望在保留文学判断弹性的同时，让评测过程更系统化且可迭代改进。
- **技术实现**：项目围绕分阶段评测流程构建，包含输入预检查、8 轴 LLM Rubric 评分、一致性整理、聚合与最终结果投影。8 轴主要覆盖平台契合程度、商业性、核心优点、核心缺点等关键维度，并通过严格 JSON 与 schema 约束保证 LLM 输出结果可被利用。系统层面则完成前端界面、后端任务执行与数据存储。
- **项目成果**：形成了包含完整前端交互界面、后端任务执行与数据库的评测工具，支持任务创建、状态跟踪、结果展示与历史回访。可输出总体判断、平台候选、市场判断等 8 轴评价结果与总体评价。验证了将 LLM Rubric 方法引入到文学评价这一高度主观且评价角度分散领域的可行性。

BLOG

- 事实层与界面层：Markdown 与 HTML 不是替代关系
- Context is All You Need：智能体的上下文工程
- 从强化学习智能体到语言智能体：建模不确定性
- 从智能体的认知结构到智能体框架
- 从反馈回路看 Agent 如何把生成变成搜索
- LLM 推理与训练的本质：从 Surrogate 到强化学习的几何空间
- The Statistical Crisis in Science
- 从 MCP 到 Agent Skills：为什么 Agent 又需要一种新的上下文工程协议？
- 从 Working Memory 到长期记忆：Agent Memory 的全景图
- AI Agent 的工程化：给 LLM 戴上确定性枷锁的外围工程
- Model Is Good Enough：2026 年，AI 真正稀缺的是应用而不是更大的模型
- 游戏行业如何引入 AI Agent

奖项与证书

- 全国大学生统计建模大赛陕西省一等奖
- 中国高校 SAS 数据分析大赛全国三等奖
- 美国大学生数学建模竞赛二等奖
- CET4: 510 | CET6: 513