贾一平

LLM Agent • AI for Software Engineering

jiayiping513@gmail.com | 个人主页：yiping.ai

教育经历

加拿大女王大学（Queen's University） 2021.09 – 2027.03（预计）

博士研究生，计算机学院（School of Computing），软件演化与分析实验室

研究方向：**人工智能驱动的软件工程（AI for Software Engineering）**；聚焦代码生成智能体、多智能体协作、智能体记忆力机制与模型不确定性分析。

加拿大戴尔豪斯大学（Dalhousie University） 2019.01 – 2021.08

计算机科学荣誉学士（Bachelor of Computer Science, Honours） **GPA：4.04/4.3**

荣誉研究项目：**基于计算机视觉的自动驾驶车道检测系统。**

工作经历

IBM 加拿大实验室（高级研究中心 CAS）｜人工智能研究实习生（全职） 2024.03 – 2026.11（预计）

参与 IBM Bob、仓库级修复智能体与 WCA4Z 研发，负责代码智能体架构设计（MCP 工具与记忆解耦）、基于不确定性的代码质量评估与模型路由，以及 COBOL 代码解释的评估与模型优化（已上线）；研究成果产出 2 篇一作论文。

清华大学–科大讯飞医疗语言处理技术联合研究中心｜算法实习生 2020.12 – 2021.03

面向多家医院的电子病历开展数据挖掘与特征工程，结合领域知识从病历文本和结构化字段中提取 20 余项描述性特征与量化指标；训练病历质量预测模型，实现潜在质量问题提前预警与骗保风险辅助识别。

项目经历

IBM Bob ｜代码生成风险预测与动态模型路由 2026.04 – 2026.11（预计）

- 面向 IBM Bob 智能体的代码生成与多模型协同推理场景，研究利用生成过程中的不确定性信号预测代码正确性，为智能体的候选代码选择与模型路由提供依据。
- 设计代码结构感知的不确定性聚合方法，结合 AST 节点类别、代码嵌套深度等结构信息，将 token-level entropy 逐行聚合为风险特征，捕捉生成过程中的不确定性与代码结构之间的关联。
- 结合 Random Forest 与 LSTM 构建混合风险预测模型，按代码长度选择分类策略，兼顾短代码结构特征与长代码逐行序列依赖；将代码正确性预测 ROC-AUC 从基于序列级 P90 熵训练的 RF 基线的 0.63 提升至 0.71。
- 基于代码正确性预测设计动态级联多模型路由策略，指导智能体在基座模型与专家模型间自适应选择；仅增加相当于两模型成本差距约 35% 的开销，即获得相当于两模型性能差距约 76% 的提升，改善生成质量与推理成本权衡。

IBM ｜仓库级软件问题自主修复智能体研发 2025.03 – 2025.10

- 面向仓库级自动修复中工具调用回执、执行日志与中间结果持续累积导致上下文冗余和推理成本增加的问题，设计动作执行与主干记忆解耦的智能体架构，构建问题分析、代码检索、故障定位、补丁生成与测试验证的自动闭环。
- 将代码检索、补丁执行、测试验证与失败恢复等工作流封装为 MCP 工具，在工具内部处理动作执行上下文，仅将任务相关结果与关键诊断证据返回 Agent 主体；通过动态维护主干记忆，减少冗余工具回执与中间日志的累积，使上下文增长速度约为 OpenAI Codex CLI 的 21%，成本约为其一半。
- 采用定制 RAG 方案检索外部知识与历史修复经验，结合自动测试反馈迭代调整修复策略；设计提前终止与人工交接机制，根据执行轨迹评估继续修复的预期收益，支持复杂问题的人工接续处理。
- 基于 DeepSeek V4.1 Flash 基座模型，在 SWE-bench Verified 上达到 91.7% 的问题解决率，相较基线方案降低 43% 的成本；通过消融实验验证记忆维护、错误恢复与知识检索对修复效果的贡献。

IBM WCA4Z ｜ COBOL 代码解释：数据构建、评估体系与模型优化 2024.03 – 2024.10

- 参与研发 IBM WCA4Z（watsonx Code Assistant for Z）的 COBOL 代码解释功能，该功能已在产品中正式上线。针对 COBOL 公开语料稀缺、合作企业因合规风险不提供源码、高质量标注几近空白的问题，负责搭建覆盖数据采集、质量评估、提示词优化、数据合成与模型微调的完整链路。
- 开发爬虫全网采集公开 COBOL 代码并去重，构建无标注语料库；以 IBM 领域专家精标的 golden set（由 20 条扩充至 50 条，每条为完整 COBOL 文件及其解释）作为有标注评测基准。
- 为弥补 BLEU 等指标的不足，设计融合自动指标与 LLM-as-a-Judge 的评估体系：在 BLEU、ROUGE、CodeBLEU 之外，逐个变量和代码块调用大模型，判断仅凭解释能否理解其含义与业务逻辑；有标注数据按自动指标、变量、代

码块各 1/3 加权，无标注数据按变量、代码块各 1/2 加权，使评估可扩展至无标注数据。

- 基于该评估体系设计遗传算法驱动的提示词自动优化：以人工调优的提示词为初始种群，由 LLM 对候选提示词进行语义级交叉与变异，并在无标注数据上迭代筛选；独立测试集 BLEU 从约 0.12 提升至 0.43（相对提升约 258%）。
- 将优化后的提示词应用于全量语料并逐条评分，产出 1600 条高质量 COBOL 代码解释数据，并基于该数据使用 LoRA 对 IBM Granite 进行参数高效微调。

面向跨毒性端点排序的多模态文献提取 Agent 工作流 2026.08 – 至今

- 与哈佛大学公共卫生学院 Gao Lab 成员合作，针对毒理学文献中毒性端点（endpoint）繁杂、指标换算依赖主观判断、不同物质毒性难以统一比较的问题，开发多模态文献提取 Agent 工作流，为基于 Learning-to-Rank 的跨端点毒性比较构建排序数据集。
- 将 PDF 文本、图片与表格的多模态信息抽取、数据融合及多轮检验流程，抽象为 Skill、Goal 与 Validation 三部分，分别定义可复用的操作方法、任务目标与校验规则，结合通用 Agent 执行框架（general-purpose harness）完成文献解析与排序信息提取。
- 使用多个不同模型独立重复执行同一任务，对抽取结果进行跨模型一致性比对，达到 72% 的完全一致率和 89% 的多数一致率，为数据筛选与后续复核提供依据。
- 将模块化抽取与校验机制重构为确定性 LangGraph 状态图工作流，收敛开放式多轮探索并串联抽取、融合与校验节点；在保持相近结果一致性的前提下，将单篇文献处理时间从 10–15 分钟缩短至 3–6 分钟。

MemoCoder：具备长期记忆的多智能体代码生成与自动修复框架 2025.01 – 2025.07

- 设计由 Planner、Code Writer、Test Executor 与 Mentor 组成的多智能体框架，通过角色协作完成代码生成与自动修复。
- 构建基于可执行测试反馈的迭代修复闭环，将编译错误、运行异常、断言失败与超时反馈给 Mentor，指导 Code Writer 定向修正。
- 提出持久化 Fixing Knowledge Set；Mentor 跨任务归纳错误模式，将成功修复经验固化为可检索的长期记忆，并采用定制 RAG 方案检索相关经验、总结高层修复策略。
- 在 MBPP、HumanEval 与 LiveCodeBench 上优于零样本生成和 Self-Repair，Pass@10 相对提升最高达 12.1%，Pass@50 相对提升最高达 14.5%。

学术论文

- **Jia, Y.**, & Zou, Y. (2026). Structure-Aware Uncertainty Estimation for Code Correctness Prediction and Adaptive Model Routing (under review).
- **Jia, Y.**, Noei, S., & Zou, Y. (2026). Engineering End-to-End Repair Loops for Repository-Level Issue Resolving (under review).
- **Jia, Y.**, Jiang, Z. M., Noei, S., & Zou, Y. (2025). MemoCoder: Automated Function Synthesis using LLM-Supported Agents. *arXiv:2507.18812*.
- **Jia, Y.**, Hassan, S., & Zou, Y. (2026). Enhancing Software Maintenance: A Learning to Rank Approach for Co-changed Method Identification. *ACM TOSEM* **(CCF-A)**. doi:10.1145/3820772.

技术能力

- 编程与工程开发：熟练使用 Python，掌握 Java、C++、R；熟悉 Linux、Git、Docker、自动化测试与 CI/CD，具备 AI 应用开发与部署能力。
- 智能体系统设计：具备端到端 Agent 系统设计与开发能力，涵盖任务规划、多智能体协作、工具集成、记忆与上下文管理、执行反馈及异常恢复；熟悉 LangGraph、MCP 和可复用 Skill 开发。
- 大模型训练与应用：熟悉 PyTorch、Hugging Face Transformers 与 PEFT，具备 LoRA 及多种 Adapter 微调实践；能够开展提示词优化、多模态模型应用及本地与云端模型部署。
- 智能体评估与优化：能够设计自动化评估流程，结合任务指标、LLM-as-a-Judge 与一致性检验评估模型表现；具备不确定性分析、多模型路由及推理质量、成本与延迟优化能力。
- 语言能力：英语流利，具备学术文献阅读与写作能力；法语 B2；日语 N2。