Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

基于大语言模型的事件根因分析

硕士研究生 刘栋涵

2024年11月24日

问题回溯

- 相关内容
  - 2024.03.31 杨宗源 《LLM的强化学习》
  - 2023.02.12 邵思源 《自动化漏洞挖掘初探》

内容提要

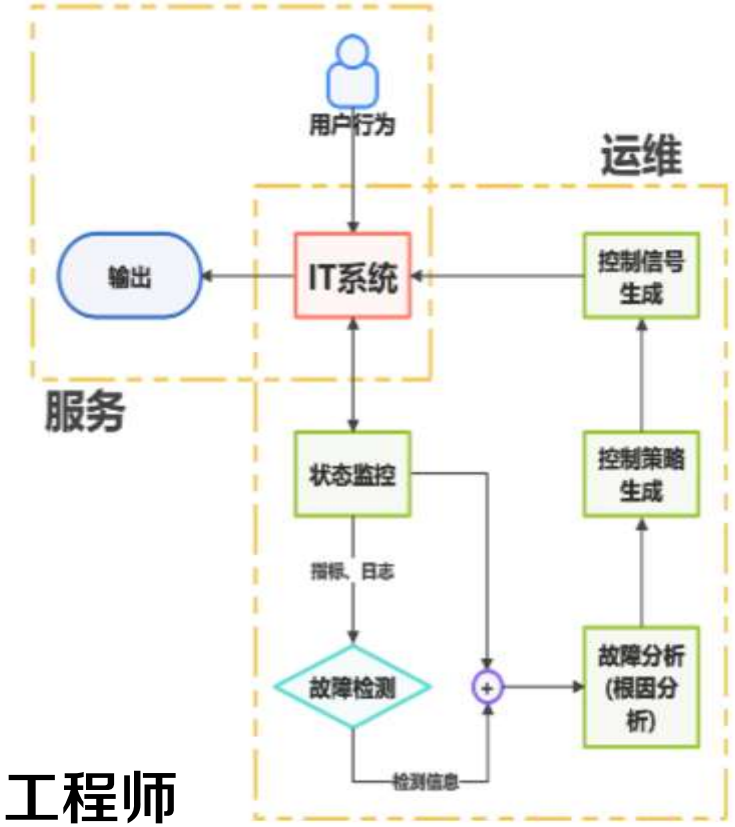
- 预期收获
- 内涵解析与研究目标
- 研究背景与意义
- 研究历史与现状
- 知识基础
- 算法原理
  - RCACopilot
  - RCAgent
- 特点总结与未来展望
- 参考文献

预期收获

- 预期收获
  - 了解事件根因分析的研究历史与现状
  - 掌握事件根因分析基本概念和内涵
  - 理解事件根因分析的过程及其原理
  - 了解事件根因分析的未来发展方向

内涵解析与研究目标

- 内涵解析
  - 根因分析(RCA)：系统性地收集和分析与事件相关的数据，识别导致系统或服务器事件的根本原因
    - 日志、指标、追踪信息和警报等
    - 中断或性能下降
  - 事件：任何导致服务中断或质量下降的情况
    - 服务无法响应、性能严重下降、用户体验受影响等
- 研究目标
  - 实现数据收集的自动化以及输出根因类别
  - 提供清晰的分析过程，让分析结果易于理解，并能被相关工程师或管理者直接使用



研究背景

- 研究背景
  - 服务器系统的复杂性和动态性
    - 系统容易受到多种故障的影响，传统RCA方法过程繁琐、容易出错，且难以应对系统的动态变化
    - 快速而准确的根因分析对于及时恢复服务并防止问题复发至关重要
  - 现有基于大规模语言模型的RCA方法普遍存在隐私风险
    - 调用外部LLM接口需要将系统数据上传到外部服务器，这增加了数据泄露的可能性
    - 云计算环境的复杂性和动态性使得数据分布变化频繁，这进一步加剧了敏感信息暴露的风险

研究意义

- 研究意义
  - 精准识别问题的根本原因，使得模型作出解释性输出，帮助工程师迅速采取有效措施恢复系统的正常运行，从而减少服务中断时间
  - 现代云服务系统复杂性高、数据多样且动态变化，传统手动方法难以满足需求，根因分析的研究为解决多源数据整合、海量信息筛选以及动态事件适应提供了理论和技术支持
  - 对复杂系统中的问题进行深层次剖析，找到系统中薄弱环节，从而增强系统的稳定性和鲁棒性

研究历史与现状

2010 Lou等人提出了“基于机器学习的根因分析”方法。这标志着根因分析从传统规则方法向智能化方向发展的重要一步

2015 Dapper提出以“追踪请求路径”为核心的根因分析方法。这种方法通过对跨节点调用的全面追踪和统计分析，能够快速定位服务器中故障的根因

2016 Aguilera等人提出了基于“时间相关性”的根因分析方法，利用分布式跟踪系统收集的时间戳信息，通过事件的因果关系来定位故障来源

2020 Zhang进一步发展了“因果推理” 技术，用于分布式服务器中的根因分析，他们结合基于图模型的因果推理方法和现代深度学习技术，实现了复杂场景下的高效根因定位

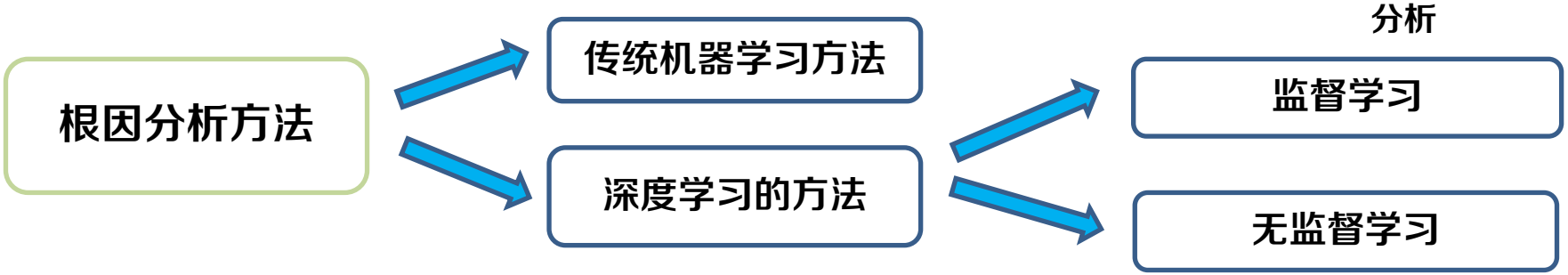
2022 Foxreb团队提出了利用大模型的方法，通过观察系统输入和输出的变化应用GPT进行根因分析

2023 Liang等人提出了PSqueeze方法，该方法引入了广义涟漪效应（GRE）的概念，通过概率聚类和启发式搜索技术实现了高效的故障定位

2023 张伟教授团队结合日志文本、系统指标和网络流量等多种数据源，开发多模态深度学习模型，实现对复杂系统的全面监控和异常检测

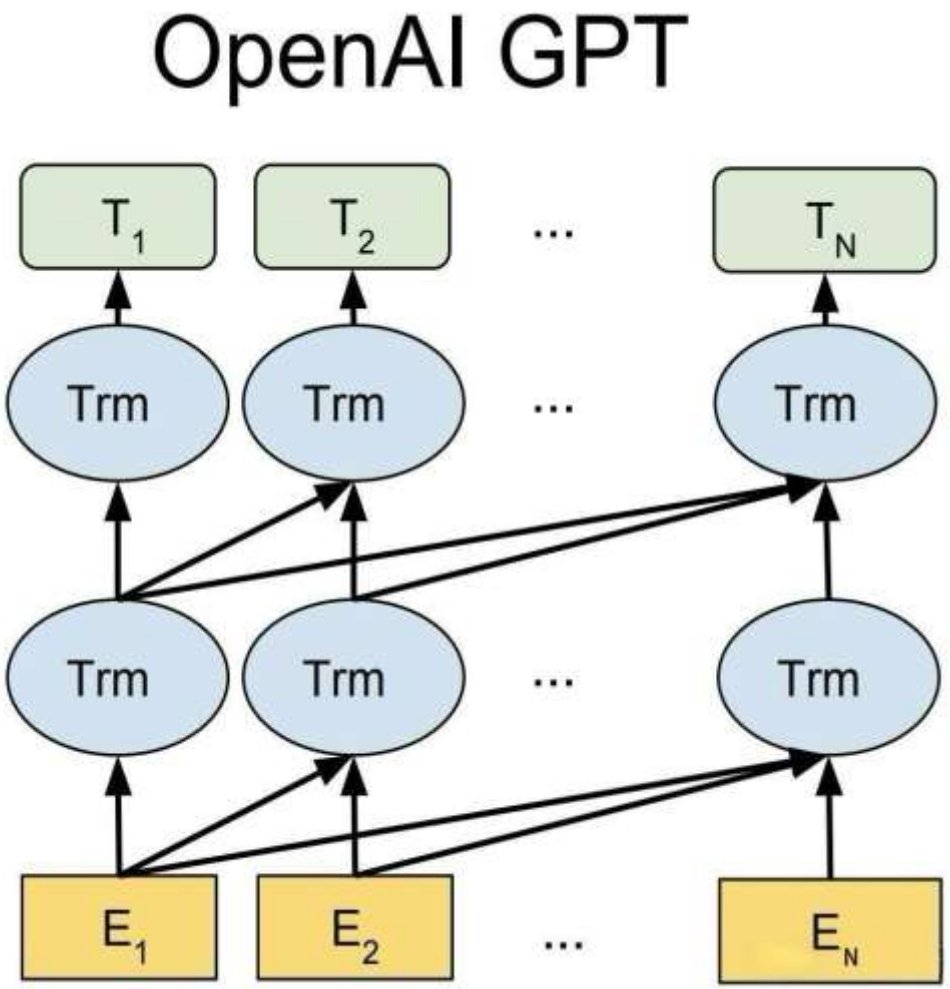
2024 Zhang 等人提出了CloudRCA框架，该框架结合了多个数据源，并使用分层贝叶斯网络模型进行根因定位，能够实现高效准确的故障分析

2024 Chandy 等人提出了CausalRCA方法。该方法通过因果结构学习，并结合根因推断技术，能够精确定位故障和相关指标



知识基础　大语言模型

- 定义
  - 大规模自然语言处理模型，具有强大的语言理解、生成和推理能力，特点是参数规模大、训练数据广泛、架构复杂，能够在各种语言任务上表现出卓越的性能
- 工作原理
  - 预训练：在大量非结构化文本数据上进行无监督学习
  - 微调：使预训练模型适应下游应用
- 分类
  - 自回归模型（GPT）
  - 自编码模型（BERT）



知识基础 ReAct

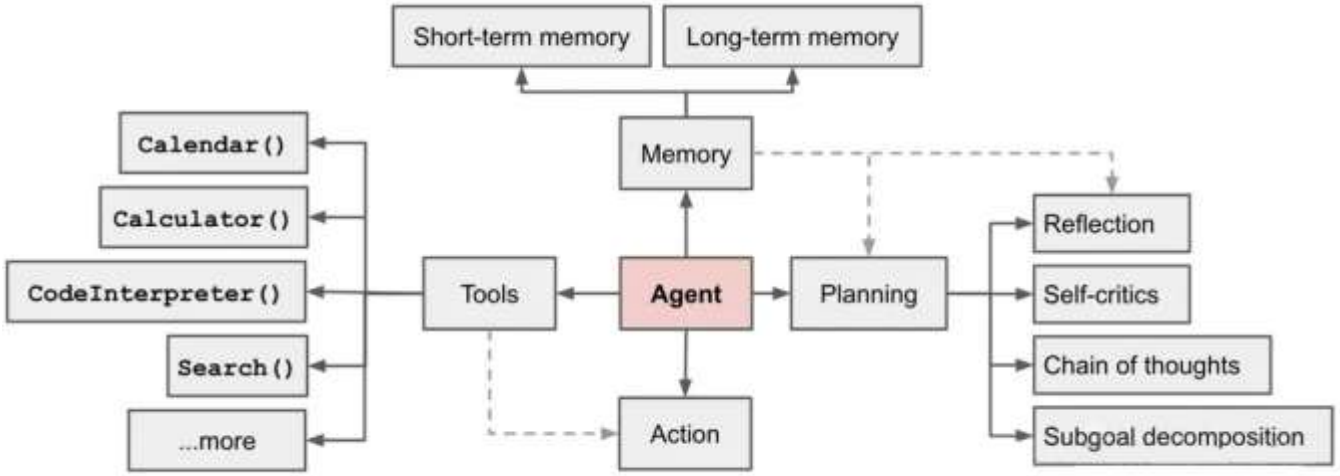
- 定义
  - 结合推理(Reasoning)和行动(Acting)的人工智能方法
- 核心理念
  - 模型生成一个**思考过程**描述对问题当前状态的**理解**
  - 模型基于思考，生成一个**行动指令**，例如查询**外部信息**、**调用API**或**访问外部网址**
  - 获取行动结果后，模型生成新的**思考过程**，再进行下一步
- 优点
  - 使得模型在执行动作时可以考虑到之前的推理结果

知识基础　思维链

- 定义
  - 通过生成一系列中间推理步骤，引导模型逐步解决复杂任务的技术
- 核心思想
  - 分步推理
    - 将问题分解为多个子问题
    - 每一步基于前一步的结果进行推导
  - 显式过程
    - 输出中包含中间推理步骤，而不是黑盒式预测
- 优点
  - 更清晰地捕捉输入数据中的逻辑关系
  - 避免复杂任务中的错误累积或遗漏关键信息
  - 提供更高的可解释性，通过显式展示中间推理路径

知识基础　智能体

- 定义
  - 一种基于大语言模型的，具有规划思考能力、记忆能力、使用工具能力，能自主完成给定任务的实体
- 关键部分
  - 规划：智能体将大型任务分解为子任务，并规划执行任务的流程
  - 记忆：分为长期记忆和短期记忆
  - 工具使用：网页搜索工具、代码执行器



Automatic Root Cause Analysis via Large Language Models for Cloud Incidents

RCACopilot TIPO

T	目标	实现数据的自动化收集，提高效率和准确率
I	输入	与事件相关的多源数据，653个真实事件
P	处理	1. 根据警报类型，自动化收集多源数据，完成特定数据的提取和整合 2. 利用嵌入模型将事件信息转换为向量，选择最相关事件作为示例输入 3. 通过思维链和诊断信息，以及历史事件进行相似匹配
O	输出	输出事件的根因类别并作出解释性说明

P	问题	1.单一数据源不足支持高效诊断 2.新类型事件是诊断过程中的最大挑战
C	条件	需要ChatGPT（版本：GPT-4）
D	难点	1.多源数据可能存在噪声、缺失或不一致，增加了数据预处理的复杂性。 2.对于没有历史参考的事件，模型的预测准确性和适应能力存在挑战
L	水平	2024 SCI 1区

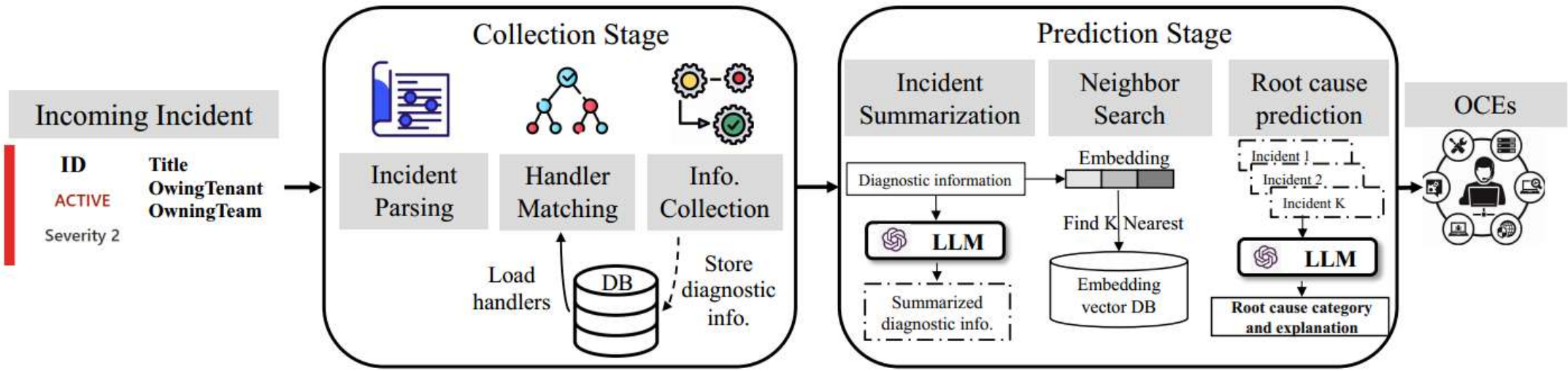
RCACopilot 现存难题

- 现有方法存在问题
  - 工程师需要手动筛选和分析海量数据，效率低下和容易出现错误，更难以快速的定位复杂系统的根因
  - 静态的故障排查指南(TSGs)难以覆盖动态的系统，对于未见过的事件类型，工程师缺乏参考案例，诊断效果低下
  - 云服务系统产生的日志、追踪信息和性能指标等多源数据体量庞大，且格式不一致，单一数据源难以全面反映问题。

- 解决方法
  - 多源数据整合
    - 针对不同报警类型自动化收集日志、追踪信息和性能指标
    - 使用动态决策树操作，保证数据收集的全面性和高效性
  - 思维链
    - 将任务分解成明确的中间步骤，并在提示中逐步列出这些步骤
    - 要求模型一步步给出答案，例如，在异常检测任务中，可以引导模型先判断变量是否符合特定条件，再决定日志是否异常

RCACopilot 算法原理

- 算法原理
  - 诊断信息收集阶段
    - 输入：事件，包括基础信息，例如报警类型和事件类型
    - 输出：一个包含多源数据的诊断信息文件，包括日志片段、追踪路径等
  - 根因预测阶段
    - 输入：系统收集到的诊断信息文件
    - 输出：当前事件的根本原因和详细解释



- 根因分析评价指标
  - Micro-F1
    - 定义：计算每个样本的正确预测数量与总预测数量的比值
    - 计算公式：$Micro-F1=\frac{2\times Micro-Precision\times Micro-Recall}{Micro-Precision+Micro-Recall}$
    - 特点：衡量系统的总体性能
  - Macro-F1
    - 定义：对每个类别分别计算F1分数，然后求平均
    - 计算公式：$Macro-F1=\frac{1}{N}\sum_{i=1}^{N}F1_i$
    - 特点：反映模型在各类别上的均衡性

实验设计　实验设置

- 模型设置
  - 实验使用ChatGPT4的API（8K token版本）作为主要推理模型
- 对比方法
  - 基线模型：XGBoost，FastText，Fine-tune GPT
  - 变体模型：GPT-4 Prompt，GPT-4 Embed
- 评价指标
  - Micro-F1和Macro-F1，模型的性能指标
  - 训练时间和推理时间，模型的效率指标
- 数据集
  - 653个真实事件，由经验丰富的工程师标注根因类别

实验设计　对比实验

– 根因分析

- 性能对比
  - RCACopilot（GPT-4）在 Micro-F1 和 Macro-F1 上均表现最佳，远远高于其他基线模型和变体模型

- 效率对比
  - RCACopilot（GPT-4）的效率表现良好，远远高于基线模型，和变体模型相差无几，可以满足生产环境的实时需求

Method	F1-score		Avg. Time (s)	
	Micro	Macro	Train.	Infer.
FastText [61]	0.076	0.004	10.592	0.524
XGBoost [3]	0.022	0.009	11.581	1.211
Fine-tune GPT [1]	0.103	0.144	3192	4.262
GPT-4 Prompt	0.026	0.004	–	3.251
GPT-4 Embed.	0.257	0.122	1925	3.522
RCACopilot (GPT-3.5)	0.761	0.505	10.562	4.221
RCACopilot (GPT-4)	**0.766**	**0.533**	10.562	4.205

算法总结 RCACopilot

- 算法贡献
  - 多源数据整合
    - 解决了传统方法中单一数据源难以提供全面信息的问题，提高了诊断信息的覆盖率
  - 自动化与实时性
    - 提供端到端的自动化流程，从诊断信息收集到根因预测实现全链路自动化

- 算法不足
  - 尽管使用思维链可以增强对新事件的处理能力，但对完全无先例的事件，模型仍然缺乏足够参考信息
  - RCACopilot的推理时间优化到4.2秒，但GPT-4的高开销仍可能成功大规模部署中的瓶颈，尤其是在事件频发的环境下

RCAgent: Cloud Root Cause Analysis by Autonomous Agents with Tool-Augmented Large Language Models

RCAgent TIPO

T	目标	在数据庞大时候，保证根因分析的准确率和可解性
I	输入	阿里云平台的5000个异常事件，选取其中的161个
P	处理	1.使用快照键机制，将输入压缩为可处理得短片段，同时保存完整信息 2.在思维-行动的同时，使用JSON修改机制解决工具调用因格式导致的错误 3.利用轨迹自治性聚合，选择语义一致性最高的结果
O	输出	事件发生的主要原因以及针对异常问题的修复建议

P	问题	1.现有方法通常涉及多种类型的大量数据，无法有效的整合利用 2.模型在工具调用的时候，可能会出现配置错误、格式错误等问题
C	条件	强大的计算资源和存储数据的数据库
D	难点	1.如何对长文本进行处理，使得不超过LLM的输入限制，而且不遗漏信息 2.如何保证在复杂的情况下，保证调用的工具不会失效
L	水平	2024 ICSE

RCAgent 现存问题分析

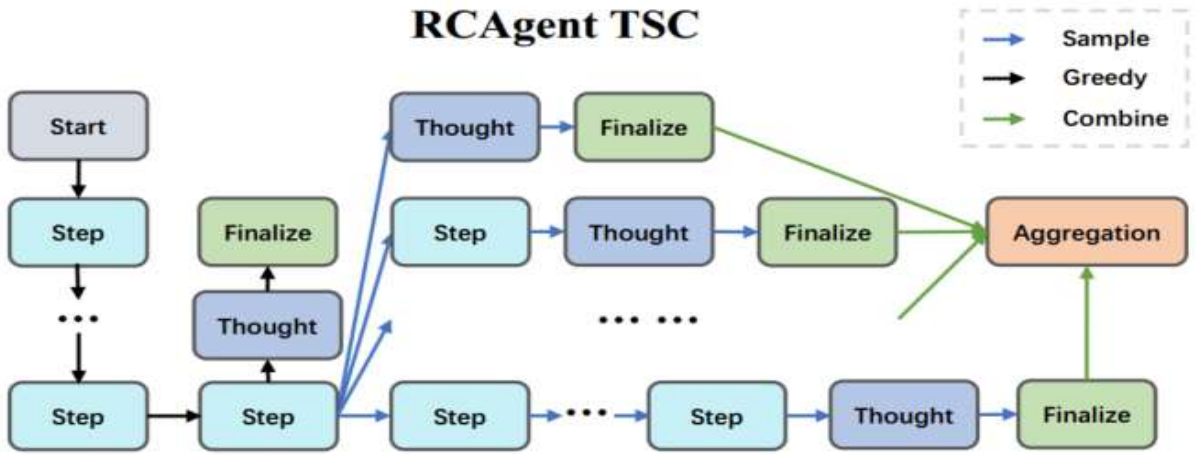
- 现有方法存在问题
  - 上下文长度问题
    - 根因分析任务通常涉及多种类型的大量数据，包括日志、代码片段、数据库查询结果等，这些数据往往非常冗长
  - 行动无效性
    - 模型在推理过程中调用工具或采取的操作，可能因为工具调用的输入错误、格式问题或参数无效，从而使得模型可能进行无关或重复的行动，浪漫资源，并影响任务效率

RCAgent 创新点分析 观测快照键

- 解决方法
  - 观测快照键（OBSK）
    - 目标
      - 减少上下文占用:将长观测结果压缩为短摘要，降低输入长度
      - 保留信息完整性:通过键值存储关联完整观测内容
    - 基本构成
      - 摘要：提取观察内容的关键信息，作为LLM的直接输入，例如错误类型，时间戳等
      - 快照键：为观测内容生成完整唯一的标识（如哈希值）,快照键与完整数据存储在键值系统中，可以随时检索，如果模型需要更详细的信息，可以根据快照键从键值存储总检索完整内容

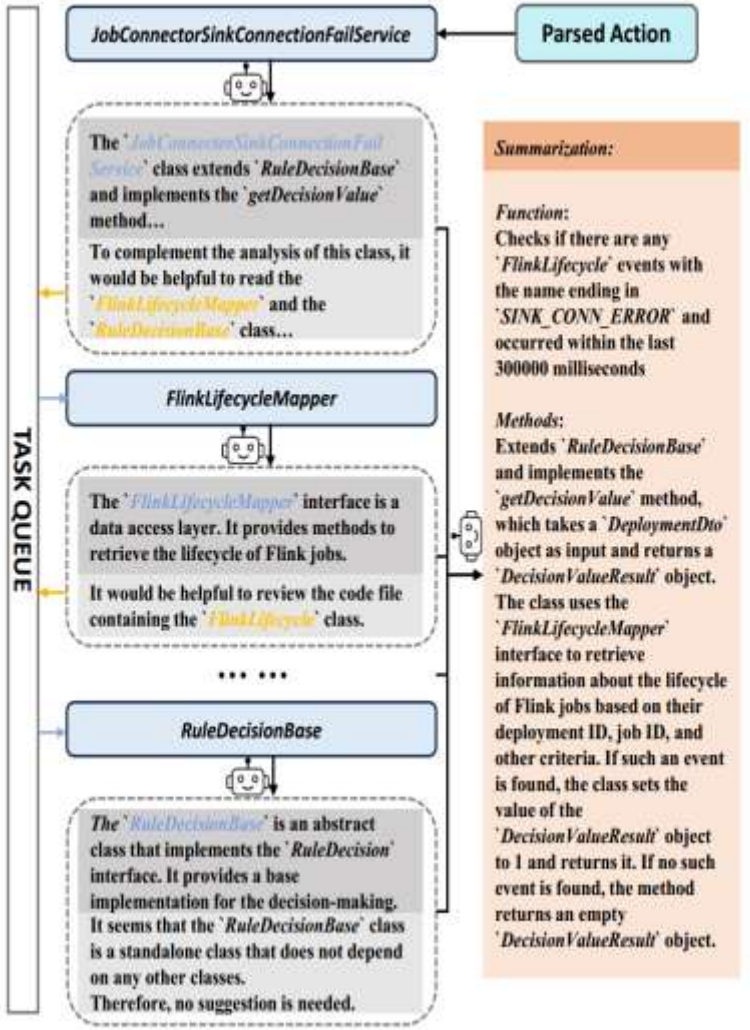
RCAgent 创新点分析 JSON和轨迹级自一致

- 解决方法
  - JSON修复机制（JsonRegen）
    - 格式检测：在每次工具调用前，检测生成的JSON数据是否符合标注
    - 自动修复:若发现错误，系统会通过清理非法字符或重新生成JSON数据进行修复
  - 轨迹级自一致性（TSC）
    - 轨迹多样性：在任务终止阶段，采用多条完整的行动轨迹
    - 轨迹语义投票：使用语义嵌入计算各轨迹之间的一致性，并选择最优的轨迹



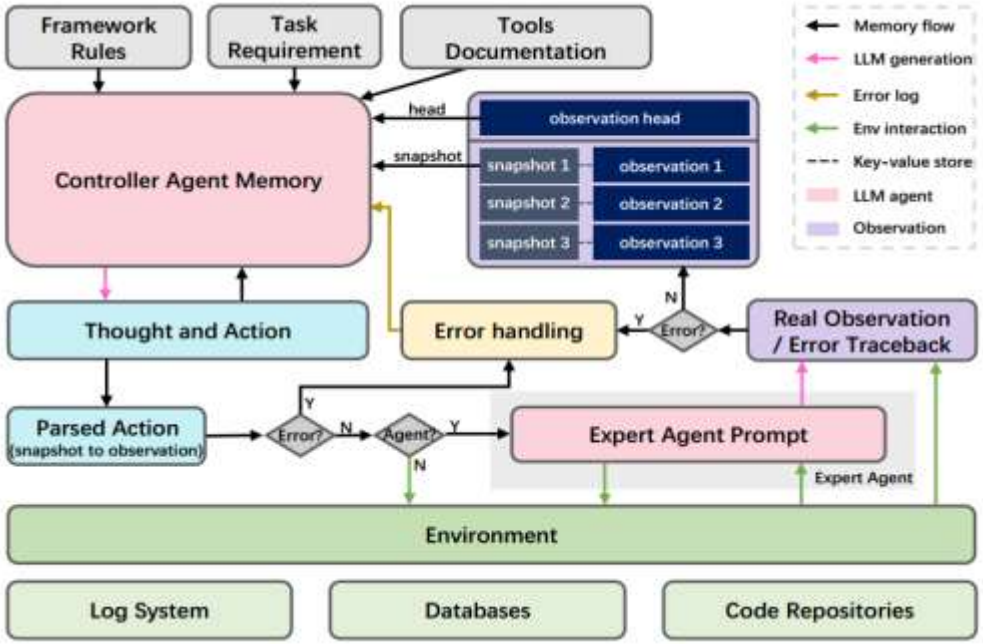
RCAgent 创新点分析 分析工具

- 解决方法
  - 分析工具
    - 代码分析工具
      - 输入：输入数据中的代码部分
      - 输出：通过LLM对复杂的代码文件内容进行自动化提取和汇总，减少人工干预
    - 日志分析工具
      - 输入：输入数据中的日志部分
      - 输出：使用LLM对每个分块进行详细分析，生成日志分析和异常诊断结果



RCAgent 算法原理

- 算法原理图
  - 控制器代理：通过思维将任务进行拆分，然后确定下一步行动
  - 专家代理：负责处理复杂的任务，利用针对性大语言模型生成高级的解决方案
  - 环境交互：从日志系统、数据库和代码仓库中获取信息
  - 错误处理：检测并修复执行过程中的错误，若无法处理，就调用专家代理



实验设计　数据资源

- 数据集
  - 包含 5000个异常任务，经过筛选和标注，最终生成 161 个多样性样本，覆盖多种异常类型
- 对比模型
  - 非基于智能体的模型:XGBoost， Finetune T5， LLM Summary
  - 基于智能体的模型:ReAct
- 评价指标
  - METEOR、BLEURT、BASTScore：衡量文本生成的质量
  - H-Helpfulness：通过人工评估，衡量生成结果对实际任务的帮助度
  - G-Correctness：由 LLM 判断根因预测的语义正确性

- 评价指标
  - METEOR
    - 定义：它通过匹配单词和短语来评估生成文本与参考文本的相似性，考虑了词形变化和语义信息
    - 公式：$METEOR = F(1 - Penalty)$
  - NUBIA
    - 定义：用于评估生成文本和参考文本之间的语义一致性和信息完整性
    - 公式：$NUBIA = \frac{E_{candidate} \times E_{reference}}{\|E_{candidate}\| \times \|E_{reference}\|}$
  - BLEURT
    - 定义：基于BERT的生成文本质量评估指标，通过训练一个模型来预测生成文本和参考文本之间的评分

实验设计　对比实验

- 实验结论
  - RCAgent在云环境中处理事件异常时，相较于其他模型具有显著优势，尤其是在根因预测、责任归属、人工帮助度等方面，RCAgent均表现出色
  - TSC的引入进一步优化了RCAgent的性能，使其在复杂任务中的表现更为出色

Model	Root Cause					Responsibility	Human
	METEOR	NUBIA	BLEURT	BARTScore	G-Correctness	Precision	H-Helpfulness
XGBoost	-	-	-	-	-	77.65	-
Finetune T5	4.18	8.20	19.32	-6.61	2.62	77.85	-
LLM Summary	7.88	14.57	25.40	-6.00	3.58	77.21	-
ReAct	5.21	10.38	20.33	-6.25	2.24	73.53	$1.36_{\pm 0.03}$
RCAgent	13.77	19.48	31.52	-5.59	3.82	80.74	$2.47_{\pm 0.17}$
w/ TSC (LLM)	**$15.72_{\pm 0.61}$**	**$26.79_{\pm 2.54}$**	**$35.72_{\pm 0.58}$**	**$-5.29_{\pm 0.03}$**	**$4.36_{\pm 0.01}$**	**$82.06_{\pm 0.42}$**	**$2.92_{\pm 0.21}$**

实验设计　消融实验

- 评估RCAgent各功能的有效性
  - w/o LLM Experts：移除专家代理后，性能大幅下降，表明专家代理在复杂任务分析中的重要性
  - w/o OBSK：去掉快照键机制后，长上下文的处理能力下降，导致 BLEURT 降低至 29.67
  - w/o Json：移除 JSON 修复功能后，任务稳定性受影响， G-Correctness 从 5.22 降至 4.19

模型设置	Root Cause (根因分析)	Responsibility (责任归属)	Human (人工评估)
	METEOR	NUBIA	BLEURT
完整 RCAgent	15.15	19.48	31.57
w/o LLM Experts	9.60	14.20	27.77
w/o OBSK	12.37	16.55	29.67
w/o JsonRegen	13.89	17.72	27.72

算法总结 RCAgent

- 算法贡献
  - 提出了首个基于 LLM 的工具增强型智能体框架，用于解决云系统中的根因分析任务
  - 引入了一系列增强方法，包括自洽性聚合、稳定性增强和快照键机制
  - 在本地部署，对隐私和信息安全起到保护作用

- 算法不足
  - 轨迹级自一致性（TSC）虽然有效，但增加计算开销，特别是在复杂任务中，生成多个路径可能导致资源浪费
  - 现有工具可能难以处理新的数据类型或不同的云环境

特点总结与未来展望

特点总结与未来展望

- 特点总结
  - RCACopilot
    - 在针对不同事件类型，利用事件处理器，从多种数据源中收集相关诊断数据，弥补单一数据源信息不足
    - 实现了云事件管理的端到端自动化，从数据收集到根因预测均由系统自主完成
    - 对于未见过的新型事件，系统能够自动生成新的根因类别，并提供相应解释，帮助OCES理解根因
  - RCAgent
    - 本地部署，避免依赖外部API，保障云平台数据的隐私和安全
    - 通过OBSK压缩上下文长度，提高效率
    - 使用JSON修复机制，提高模型在复杂任务的稳定性

参考文献

[1] Sun Y, Huang Y, Yu Y, et al. DivLog: Log Parsing with Prompt Enhanced In-Context Learning[C]. Proceedings of the 46th International Conference on Software Engineering (ICSE 2024). ACM, 2024: 1-12

[2] Liu, Y., Tao, S., Meng, W., Wang, J., Ma, W., Zhao, Y., Chen, Y., Yang, H., Jiang, Y., & Chen, X. (2023). LogPrompt: Prompt Engineering Towards Zero-Shot and Interpretable Log Analysis :2308.07610.

[3] Meng W, Zhao Y, Chen Y, et al. Automatic Root Cause Analysis via Large Language Models for Cloud Incidents[J]. IEEE Transactions on Cloud Computing, 2023, 11(4): 1123-1135.

[4] Ma, L., Yang, W., Xu, B., Jiang, S., Fei, B., Liang, J., Zhou, M., & Xiao, Y. (2024). KnowLog: Knowledge Enhanced Pre-trained Language Model for Log Understanding. Proceedings of the 46th International Conference on Software Engineering (ICSE 2024), 1-12.

知人者智，自知者明。胜人者有力，自胜者强。知足者富。强行者有志。不失其所者久。死而不亡者，寿。

谢谢！