

MicroRCA-Agent: 基于大模型智能体的微服务根因定位方案

队伍: 男团910

演讲人: 汤攀

单位: 上海大学



主办单位: 中国计算机学会 (CCF)

承办单位: 中国计算机学会互联网专委会、中国科学院计算机网络信息中心、中国移动研究院、清华大学

协办单位: 华为2012实验室、阿里云、中兴通讯、中国移动九天团队、南开大学、西安电子科技大学、清华大学计算机科学与技术系、神州灵云

目录 CONTENTS

第一章节 团队介绍

第二章节 赛题分析

第三章节 具体方案设计

第四章节 总结与思考

第一节 团队介绍

团队介绍

团队名称:

- 男团910

团队成员:

- 汤攀、唐世祥、浦桓齐

所在单位:

- 上海大学

初赛成绩:

- 初赛第6

特别鸣谢:

- 苗志清 华东师范大学 (讨论协作和服务支持)
- 汪治兴 北京理工大学 (讨论协作和服务支持)



第二章节

赛题分析

基于大模型智能体的微服务（三种模态数据）根因定位

大模型如何设计使用？



- ★ 上下文长度
- ★ 哪些地方使用？

日志



调用链



指标



三种模态数据如何处理？

- ★ 不同数据形式不同



压缩各模态有效信息

agent 总结 metric

多模态中 agent 定位根因

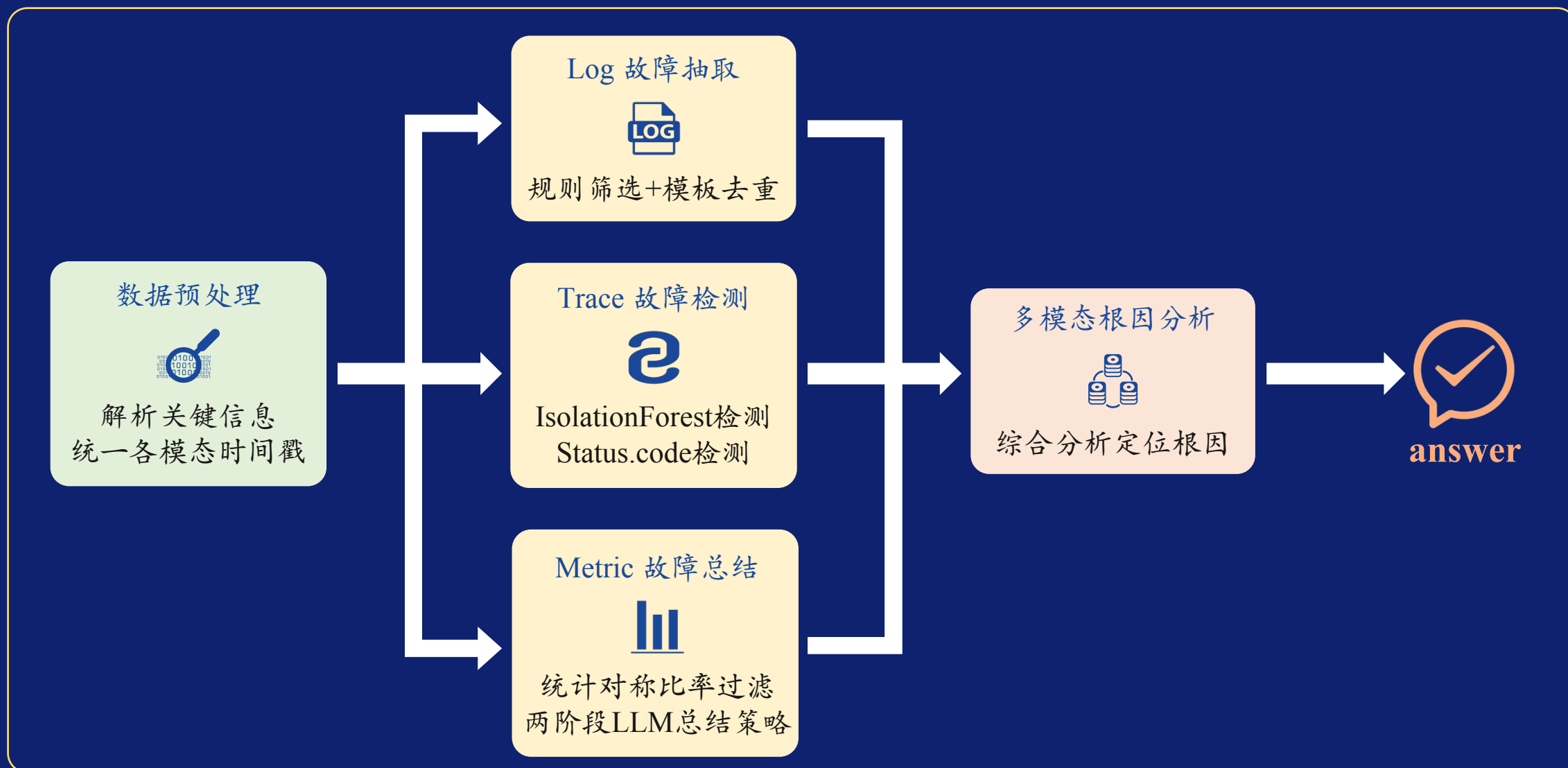
Log: 模板化+规则化

Trace: 无监督学习+规则增强

Metric: 两阶段指标总结分析

第三章 具体方案设计

- 方案总体框架包括数据预处理、log故障抽取、trace故障检测、metric故障总结和多模态根因分析五个模块。



数据预处理包含输入数据解析和多模态时间戳统一：解析问题中关键信息，并统一各模态数据时间戳以进行时间对齐。

输入数据 (input.json) 解析

```
{
  "Anomaly Description": "The system experienced an anomaly from 2025-06-05T16:10:02Z to 2025-06-05T16:31:02Z. Please infer the possible cause.",
  "uuid": "345fbe93-80"
}
```



多模态时间戳统一

Log数据

...	@timestamp	...
...	2025-06-05T16:10:02Z	...



...	timestamp_ns	...
...	1749139802000000000	...

Trace数据

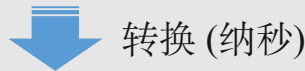
...	time	...
...	2025-06-05T16:10:02Z	...



...	timestamp_ns	...
...	1749139802000000000	...

Metric数据

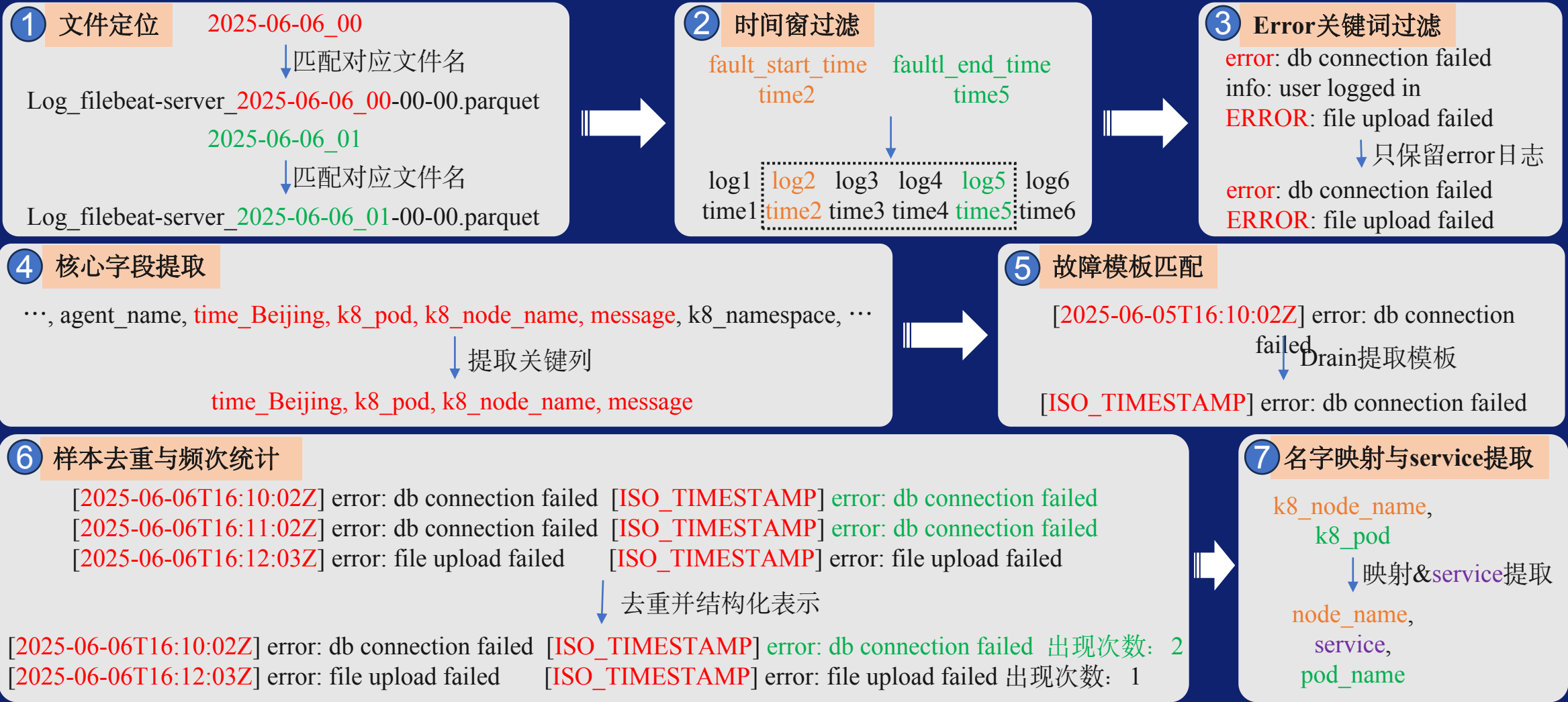
...	startTime	...
...	1749139802000000	...



...	timestamp_ns	...
...	1749139802000000000	...

Log故障抽取

Log采用“规则化+模板化”方式，通过多重规则筛选和模板去重，实现对error日志信息的抽取和压缩。



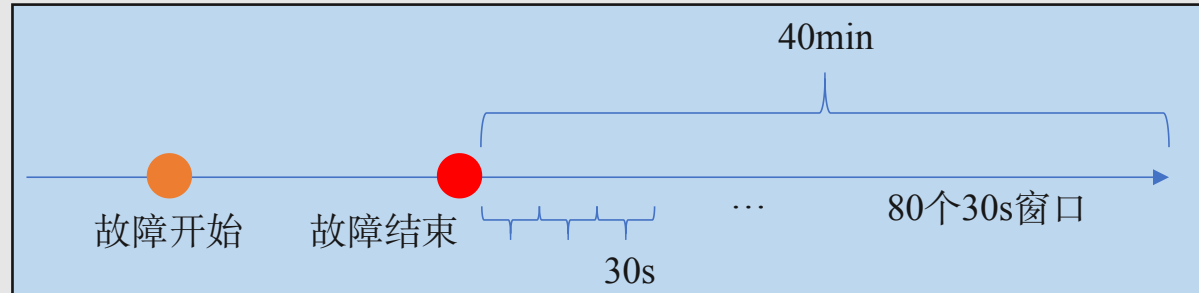
Log故障抽取结果示例:

node_name,	service_name,	pod_name,	message,	occurrence_count
aiops-k8s-04,	frontend,	frontend-2,	{"error": "could not retrieve product: ...",	出现次数: 9

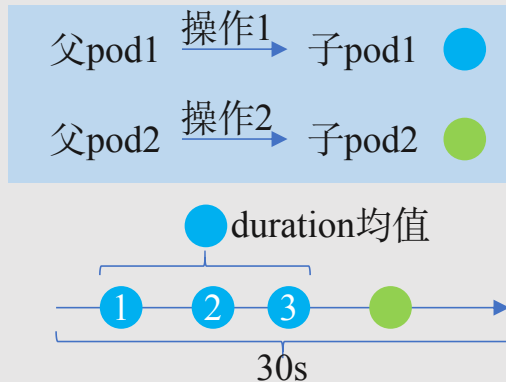
Trace故障检测

Trace 采用 “无监督学习+规则增强” 的混合异常检测策略。

- 无监督学习：利用 IsolationForest 检测 duration 异常，抽取50个样本异常结束后40分钟的数据训练多组模型
- 规则增强检测：利用 status.code≠0 和 对应的 status.message 检测状态异常



50个样本



50个样本 × 80个30s窗口的均值=4000个duration均值（如果每个30s都有数据）

父pod

子pod

操作名称

frontend-0_frontend-0_hipstershop.ProductCatalogService/GetProduct,

frontend-0_frontend-0_hipstershop.CheckoutService/PlaceOrder,

...

frontend-2_cartservice-2_POST /hipstershop.CartService/AddItem,

调用duration列表 (<4000个值)

[113.19, 125.13, 2349.66, ...]

[148.41, 18672.41, 153.59, ...]

[3544.81, 3925.68, 3876.12, ...]

训练模型

IsolationForest1 检测

IsolationForest2 检测

IsolationForestn 检测

Isolation异常

+

Status异常

Trace故障检测
结果示例
(取前20) :

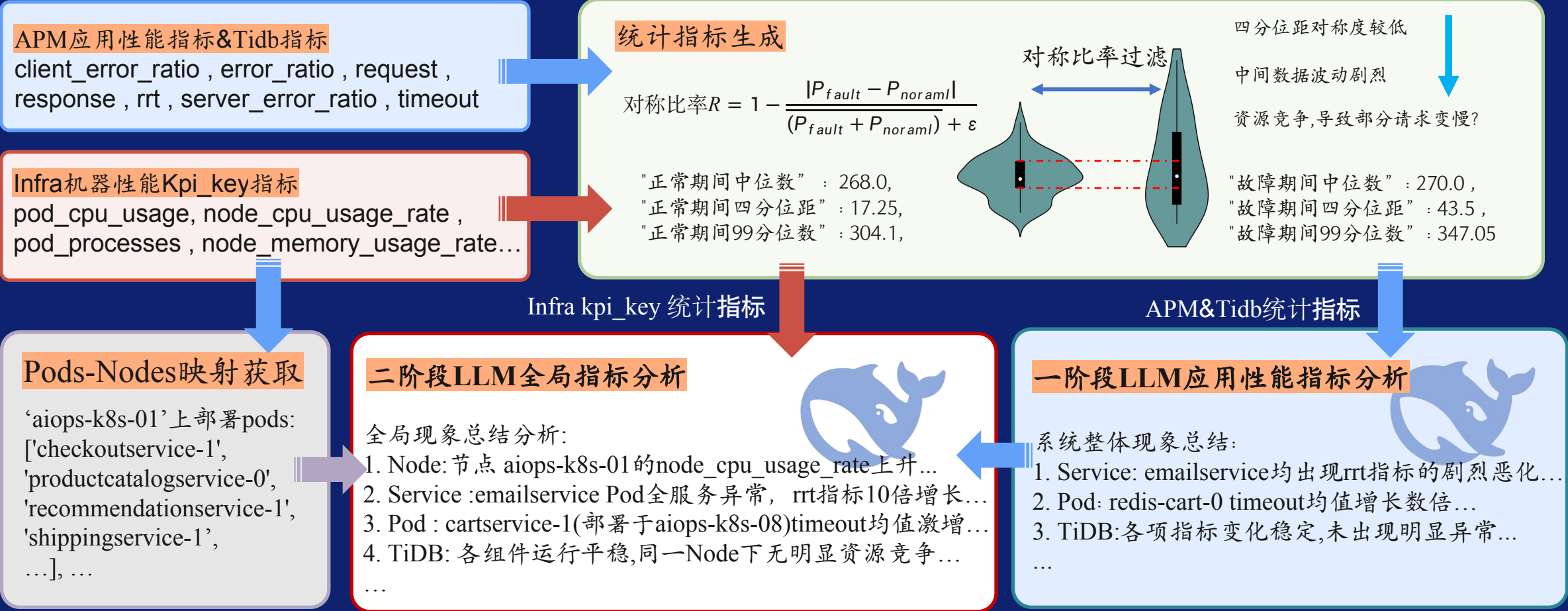
node_name	service_name	parent_pod	child_pod	operation_name	normal_avg_duration	anomaly_avg_duration	occurrence_count
aiops-k8s-04, frontend,	N/A,	frontend-1,	histershop.Frontend/Recv.,	11203.63505,	56375295.28801,		出现次数: 32
aiops-k8s-04, frontend,	N/A,	frontend-1,	histershop.Frontend/Recv.,	13, HTTP status cde: 500,			出现次数: 161

Status异常
Status.code ≠ 0
Status.message1:
HTTP status code: 500
Status message2:
context canceled

Metric故障总结

Metric 故障总结模块设计了“统计对称比率过滤+层次化两阶段的LLM分析”策略。

- 统计对称比率过滤：将海量数据通过统计指标总结，对比是否统计对称，过滤正常数据，大幅度降低输入token。
- 层次化两阶段的LLM分析：将多层级指标分次总结，降低上下文推理难度。



Log故障数据、Trace异常模式和Metric 现象总结进行数据整合, 构建最终根因定位Agent, 并进行输出格式规范处理

Log故障抽取结果

node_name, service_name, pod_name, message, occurrence_count
aiops-k8s-04, frontend, frontend-2, {"error": "could not...", 出现次数: 9

Trace故障检测结果

node_name, service_name, parent_pod, child_pod, operation_name, ...
aiops-k8s-04, frontend, N/A, frontend-1, histershop.Frontend, ...

Metric故障现象总结结果

Node: aiops-k8s-01的node_cpu_usage_rate相较于正常期间显著上升...
Service :emailservice Pod全服务异常, 相较正常段rrt指标10倍增长...
...

```
{  
  "component": "adservice-0",  
  "uuid": "8c1e8ce9-237",  
  "reason": "kpi_key `pod_cpu_usage` surged and kpi_key `pod_memory_working_set_bytes` increased significantly in adservice-0",  
  "reasoning_trace": [  
    {  
      "step": 1,  
      "action": "LoadMetrics(adservice-0)",  
      "observation": "kpi_key `pod_cpu_usage` surged and kpi_key `pod_memory_working_set_bytes` increased in metric"  
    },  
    {  
      "step": 2,  
      "action": "TraceAnalysis('frontend -> adservice')",  
      "observation": "hipstershop.AdService/GetAds duration spiked in trace"  
    },  
    {  
      "step": 3,  
      "action": "LogSearch(adservice-0)",  
      "observation": "DeadlineExceeded error found in log"  
    }  
  ]  
}
```

Agent
推理结果

大模型输出不稳定
导致格式残缺(缺失括号)

多模态根因定位prompt

请根据提供的日志数据、链路追踪数据、系统指标数据, 进行综合故障分析, 识别最有可能的单一故障原因。不要包含任何其他解释或文本。要求:

1. 综合多种监控数据进行分析, 优先考虑数据间的关联性
2. 只返回一个最可能的故障分析结果
3. 故障级别判断标准: ...
...

可用的监控数据:

日志异常数据: ...
微服务链路异常数据: ...
Metric故障总结分析结果: ..

```
{  
  "component": "adservice-0",  
  "uuid": "8c1e8ce9-237",  
  "reason": "kpi_key `pod_cpu_usage` surged and kpi_key `pod_memory_working_set_bytes` increased significantly in adservice-0",  
  "reasoning_trace": [  
    {  
      "step": 1,  
      "action": "LoadMetrics(adservice-0)",  
      "observation": "kpi_key `pod_cpu_usage` surged and kpi_key `pod_memory_working_set_bytes` increased in metric"  
    },  
    {  
      "step": 2,  
      "action": "TraceAnalysis('frontend -> adservice')",  
      "observation": "hipstershop.AdService/GetAds duration spiked in trace"  
    },  
    {  
      "step": 3,  
      "action": "LogSearch(adservice-0)",  
      "observation": "DeadlineExceeded error found in log"  
    }  
  ]  
}
```

最终标准化输出
写入result.json

正则表达补齐

三种模态处理方案数据量压缩统计（平均）

Log 数据压缩统计				
匹配到log文件行数	时间过滤后行数	Error字段过滤后行数	Drain模板去重后行数	Token 统计
288242.34行	67324.19行	1248.38行	14.74行	约 2400 tokens

Trace 数据压缩统计				
匹配到trace文件行数	时间过滤后行数	IsolationForest检测到行数	分组聚合后行数	Token 统计
308926.08行	69887.22行	139.17行	18.16行	约 1000 tokens
		包含status的trace行数	Status异常记录行数	Status异常分组后行数
		66441.70行	772.64行	5.03行
				Token 统计
				约 300 tokens

Metric 数据压缩统计				
匹配到metric行数	metric文件数量	使用的文件数量	输出文本	Token 统计
1045282.06行	112个	72个	1500-2000字	约 1500 tokens

第四章

总结与思考

总结



多重规则筛选过滤
drain模板去重



IsolationForest异常检测
status状态异常检测



统计对称比率过滤
两阶段LLM现象总结

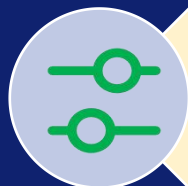
思考



常见故障类型设计sop



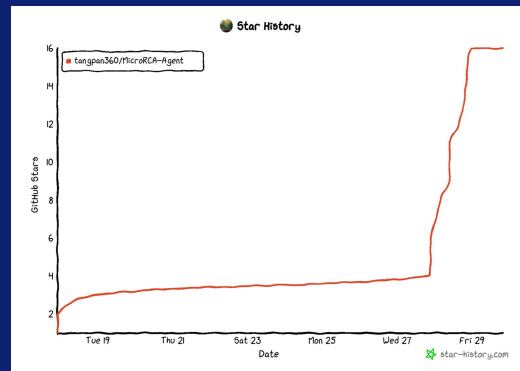
设计多智能体对话系统



调整参数，提示词等
(几乎未调优)

开源仓库:

- <https://github.com/tangpan360/MicroRCA-Agent.git>



Q&A

OpenAIOps AIOPS | 2025 CCF国际AIOps挑战赛
2025 CCF International AIOps Challenge

THANKS

主办单位：中国计算机学会（CCF）

承办单位：中国计算机学会互联网专委会、中国科学院计算机网络信息中心、中国移动研究院、清华大学

协办单位：华为2012实验室、阿里云、中兴通讯、中国移动九天团队、南开大学、西安电子科技大学、清华大学计算机科学与技术系、神州灵云