

10.20 AIOps故障定位挑战赛 副本

赛题介绍

赛题构建真实的云原生电商系统，部署在跨 AZ 的可用区下：

- 系统有背景负载以及模拟用户下单交易行为。
- 系统会有实时多模态可观测数据产生（Log、Trace、Metric、Entity、Event）。这些可观测数据都会被阿里云云监控 2.0 系统记录下来。这个系统的数据开放给所有人访问，无需登录认证。
- 故障注入系统会在各层次注入故障。

赛题输入

参赛选手从天池下载 JSONL 输入文件，每行一题，包含时间窗、候选根因与告警规则。

代码块

```
1 {"problem_id": "string", "time_range": "YYYY-MM-DD HH:mm:ss ~ YYYY-MM-DD  
HH:mm:ss", "candidate_root_causes": ["<component>.<fault_type>",  
"..."], "alarm_rules": ["overall_error_count", "..."]}
```

赛题输出

需要参赛选手基于系统的可观测数据，能准确定位故障原因。选手需在候选集合中选择可能的根因，提交格式为 JSONL 文件。每道题的标准答案 root_causes 是一个没有多余项的待修复注入点列表。

代码块

```
1 {"problem_id": "string", "root_causes": ["<component>.<fault_type>", "..."],  
"evidences": ["string", "..."]}
```

示例代码思路

1. 数据输入处理：

从 dataset/input.jsonl 文件读取问题数据

每个问题包含：问题ID、时间范围、候选根因列表和告警规则

2. 分析类型判断：

通过 determine_analysis_type 函数分析告警规则

将问题分为两类：错误类型(error)和延迟类型(latency)

错误类型关键词：error、failure、exception、status

延迟类型关键词：rt、latency、response、duration、time

3. 根因分析执行：

错误类型问题调用 analyze_error_root_cause 函数

延迟类型问题调用 analyze_latency_root_cause 函数

这些函数分别从 STS_Root_Cause_Analysis_Error.py 和 STS_Root_Cause_Analysis_Latency.py 导入

4. 错误根因分析：

使用 find_root_cause_spans_error.py 中的算法

查询状态码>1的spans，识别错误

按traceID分组，构建span父子关系

找出"根因span"：状态码>1且没有子span出错的span

5. 延迟根因分析：

使用 find_root_cause_spans_rt.py 中的算法

分析响应时间异常的spans

类似错误分析，但关注的是响应时间而非错误状态

6. 结果处理与输出：

将分析结果格式化为标准JSON格式

输出到 dataset/output.jsonl 文件

每个问题的结果包含问题ID和根因列表