AIOps Challenge 智能运维挑战赛

2021国际AIOps挑战赛决赛
暨AIOps创新高峰论坛

微服务系统异常检测与根因定位

队伍名称：一行bug

答辩队员：张世泽

科创会 NIICON
第一届国际互联网产业科技创新大会暨互联网创新产品展览会
The First International Internet Industry Science And Technology Innovation Conference & Internet Innovation Product Exhibition

目录

团队介绍

单位

- 清华大学网络科学与网络空间研究院

成员

- 张世泽　博士研究生
- 赵鋆峰　硕士研究生
- 王凯　博士研究生
- 指导老师：杨家海、王之梁老师

实验室主页地址：　http://nmgroup.tsinghua.edu.cn

正赛成绩

AIOps Challenge 智能运维挑战赛 | NIICON 科创会

初赛总成绩第 3 名

ranks	cname	score
1	Apollo 11	3235
2	pa_tech	2734
3	一行bug	2684
4	LR-AIOps	2424
5	浦智运维战队	1708
6	一轮游	1669
7	伊莉丝	1526
8	铃动时序智能	1393
9	AIOps先锋战队	1157
10	JustDo	1065

复赛总成绩第 2 名

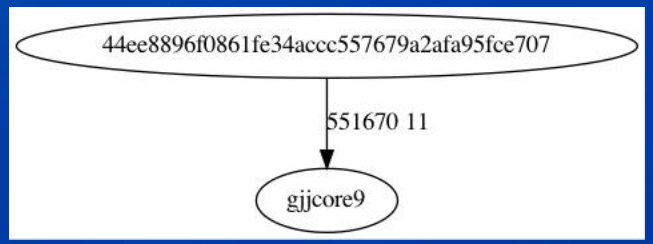
ranks	cname	score
1	LR-AIOps	2799
2	一行bug	2441
3	铃动时序智能	2070
4	一轮游	1470
5	pa_tech	1336
6	宝兰德	1241
7	伊莉丝	1160
8	bocoiops	1085
9	mad-brAIned	975
10	翼起飞	972

赛题数据分析

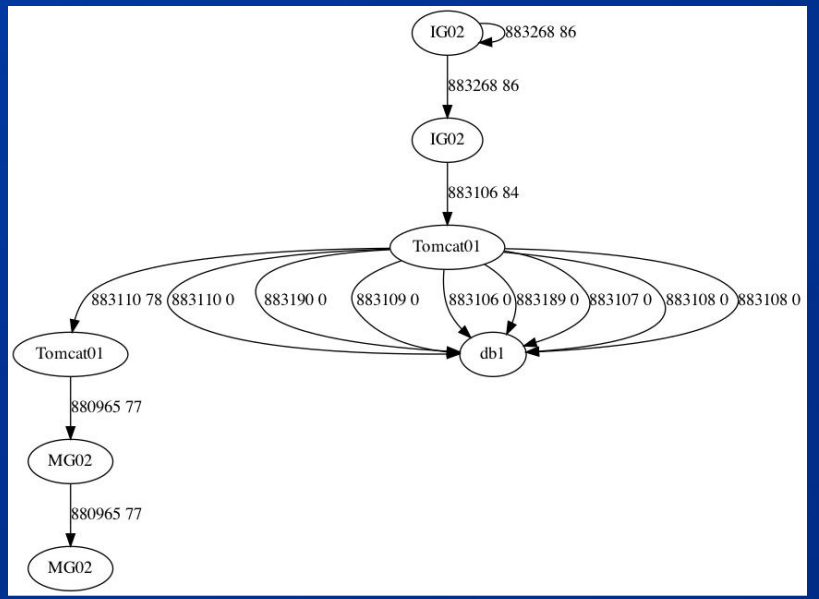
本次比赛采用了两个不同微服务系统的数据

- 指标型数据
- 调用链数据
- 日志数据

- 为了体现算法的**通用性**，我们只使用了**指标数据**和**日志数据**，**去除了调用链数据**
- **核心采用指标数据，日志数据作为补充**



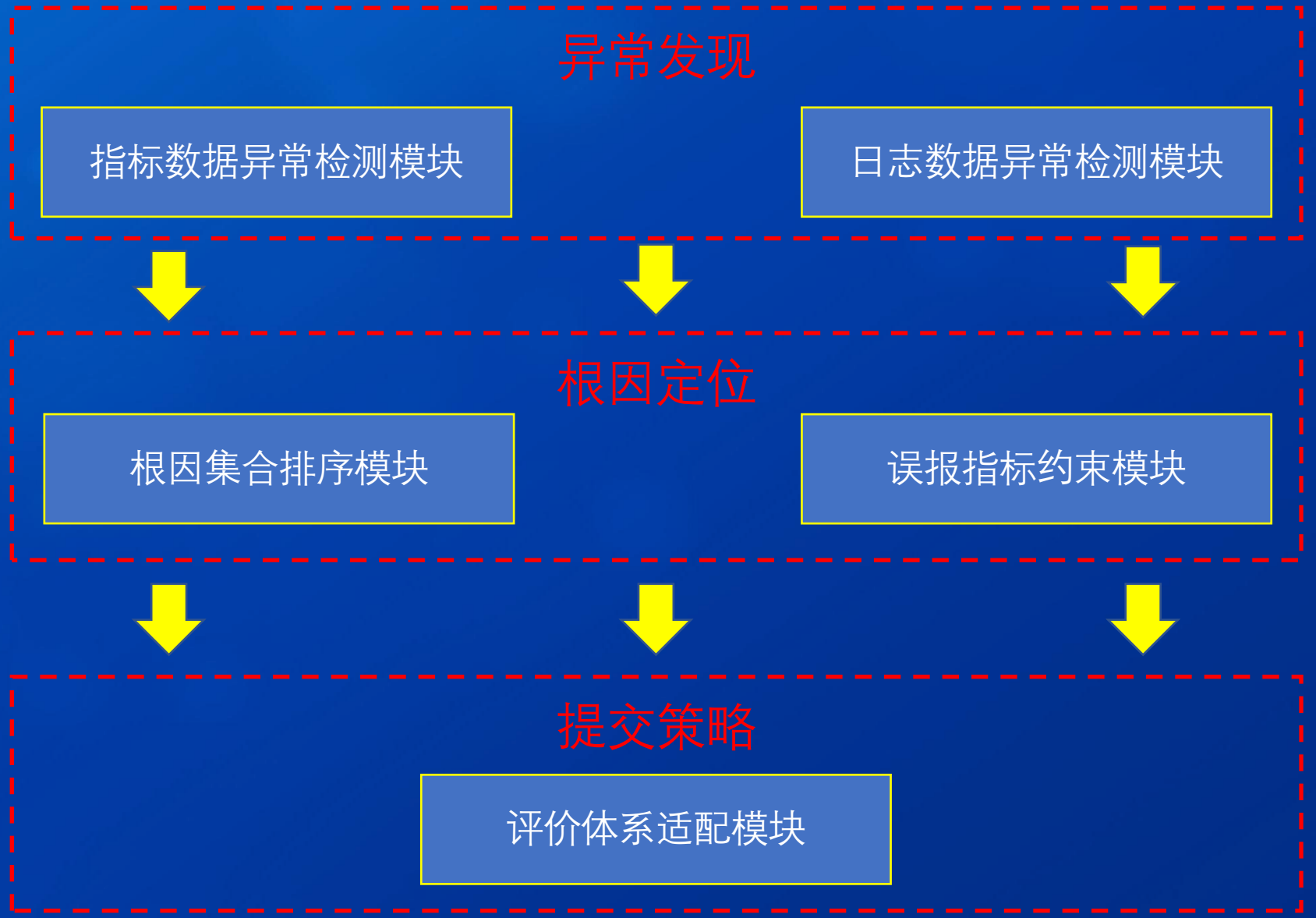
系统A



系统B

系统A与系统B的调用链数据存在明显的差异性

算法框架

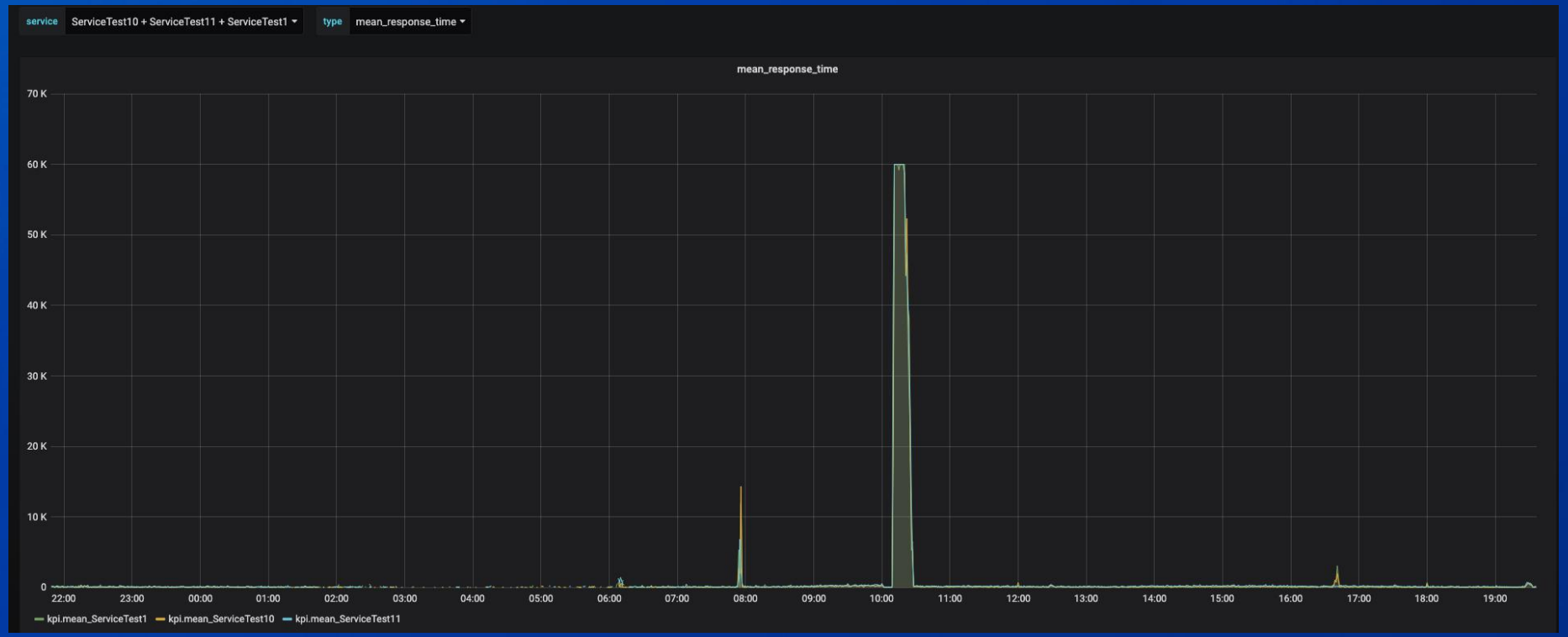

异常发现
指标数据异常检测模块
日志数据异常检测模块
根因定位
根因集合排序模块
误报指标约束模块
提交策略
评价体系适配模块

异常发现→根因定位→提交策略

AIOps Challenge 智能运维挑战赛 | 科创会 NIICON

指标数据

- 常用的方法：以业务指标为入口
  - 系统响应率
  - 业务成功率
  - 交易量
  - 平均响应时间

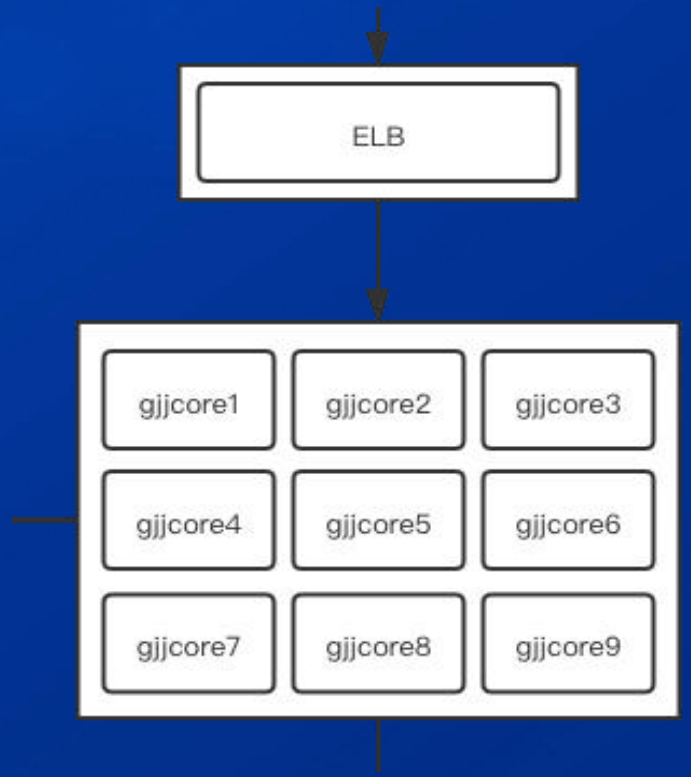


系统B的平均响应时间

异常发现→根因定位→提交策略

指标数据

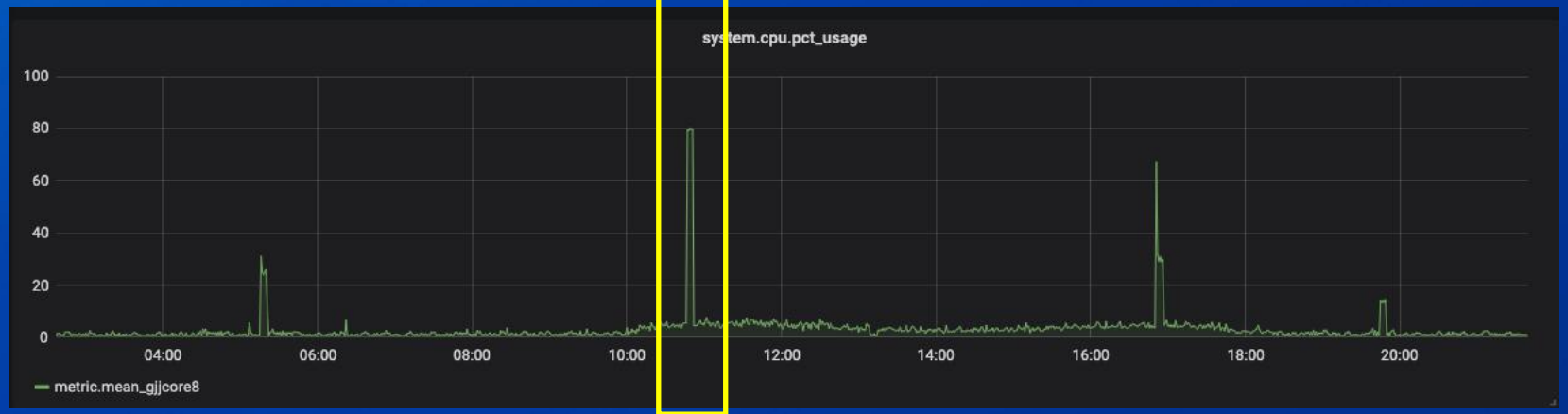
- 常用的方法：以业务指标为入口
  - 问题：系统A由于存在较多的负载均衡组件，单机的故障无法影响业务指标



系统A负载均衡组件

异常发现→根因定位→提交策略

业务指标:平均响应时间无明显变化



gjjcore8的CPU利用率指标有明显异常

2021/2/26 10:47:03AM

gjjcore8 CPU使用率高

异常发现→根因定位→提交策略

指标数据

- 常用的方法：以业务指标为入口
  - 问题：系统A由于存在较多的负载均衡组件，单机的故障无法影响业务指标
- 实际采用的方法：对全部的监控指标都进行异常检测
  - 问题：指标数量较多，导致异常检测算法的速度与准确率要进行均衡

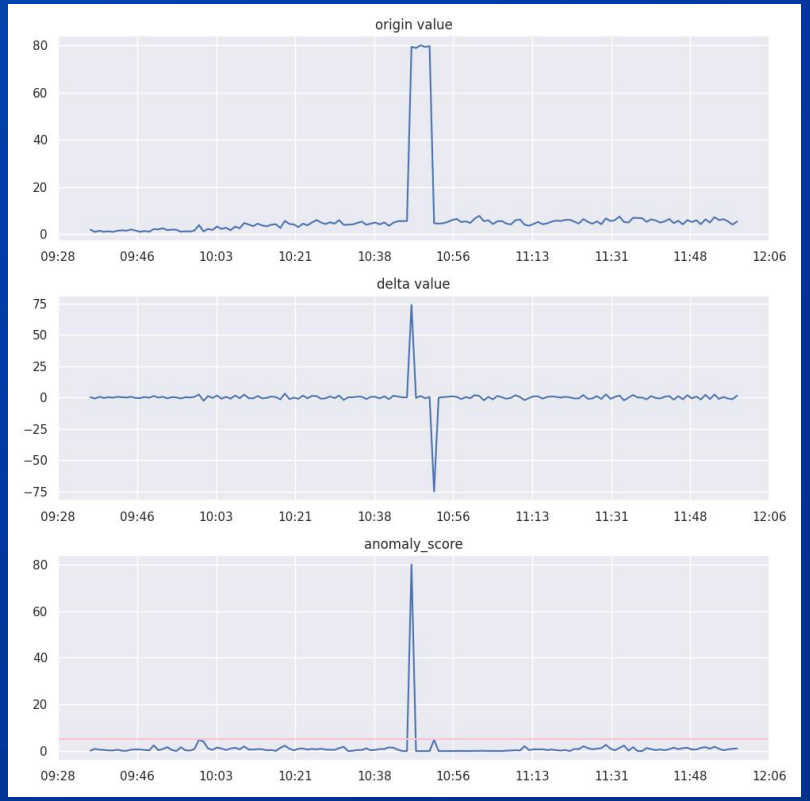
	A	B
总指标数量	2006	604

只考虑可能出现在根因列表里的指标

异常发现→根因定位→提交策略

指标数据

- 常用的方法：以业务指标为入口
  - 问题：系统A由于存在较多的负载均衡组件，单机的故障无法影响业务指标
- 实际采用的方法：对全部的监控指标都进行异常检测
  - 问题：指标数量较多，导致异常检测算法的速度与准确率要进行均衡
  - 解决思路：
    - 异常检测要保证速度与覆盖率（容忍高误报）
    - 算法选择：一阶差分+3σ



	A	B
运行速度	0.05s/轮	0.01s/轮
指标覆盖率	0.75	0.57
故障覆盖率	1.0	0.83
正确指标数量	50(/66)	26(/45)
总提交指标数量	7142	3790

异常发现→根因定位→提交策略

AIOps Challenge 智能运维挑战赛 | NIICON 科创会

日志数据

- 可供参考的标签只有3个
- 保守策略，从给的标签中提取特征

```
4699398.668：[Full GC (Allocation Failure) 2021-03-04T03:14:14.656+0800：4699398.668：[CMS：3145387K->3145305K(3145728K)，8.2829892 secs] 4089009K->3998566K(4089472K)，[Metaspace：815524K->815524K(1794048K)]，8.2832236 secs] [Times：user=8.25 sys=0.00，real=8.28 secs]
```

Tomcat 03的一个gc异常

log_weblogic_mSrv1;

log_weblogic_hfms7001

gc

关键词匹配
+
信息抽取

```
#### <Info> <Health> <gjjcore2> <mSrv1> <weblogic.GCMonitor> <<anonymous>> <> <> <phone_numec670ffd36a8a8570bf758405f93dddd24f4a33776> <BEA-310002> <1% of the total memory in the server is free>
```

gjjcore2的log_weblogic_mSrv1异常

日志无法单独报出，线上存在持续十小时的连续日志异常…

异常发现→根因定位→提交策略

根因定位

- 核心要解决的问题：
  - 多节点、多指标类型告警信息排序
  - 约束异常检测产生的大量误报

2021-02-26 10:49:05 gjjcore8 CPU使用率高

1. gjjuaap1: system.tcp.time_wait, system.cpu.iowait
2. gjjha1: system.tcp.last_ack
3. gjjuaap2: system.tcp.established
4. gjjweb002: system.load.5, system.load.15, system.cpu.iowait
5. gjjcoreap03: system.mem.cached, system.net.tcp.retrans_segs
6. gjjap03: system.tcp.close_wait, system.tcp.fin_wait1
7. gjjweb003: system.load.norm.5, system.load.5
8. gjjcore5: system.cpu.iowait
9. gjjvincio2: system.load.15, system.load.1.pct, system.load.1, system.load.norm.1, system.load.5
10. gjjweb001: system.load.1, system.load.5, system.load.norm.1, system.load.norm.5, system.load.norm.15, system.load.15
11. gjjbps1: system.tcp.fin_wait1, system.load.norm.15
12. gjjcore2: system.tcp.established
13. gjjcoreap04: system.load.norm.5, system.load.15, system.load.5, system.load.norm.1, system.load.norm.15, system.net.tcp.retrans_segs, system.load.1
14. gjjap1: system.tcp.close_wait
15. gjjvincio1: system.tcp.fin_wait1
16. gjjcore8: system.load.15, system.load.1, system.load.1.pct, system.cpu.i_dle, system.load.5, system.cpu.user, system.load.norm.5, system.cpu.pct_usage, system.load.norm.1

异常发现→根因定位→提交策略

根因定位

- 核心要解决的问题：
  - 多节点、多指标类型告警信息排序
  - 约束异常检测产生的大量误报
- 解决思路：
  - 多节点
    - 按告警指标数量多少进行排序
  - 多指标类型
    - 按指标优先级
      - CPU<网络<内存<硬盘

	A	B
CPU	39	17
内存	38	15
网络	28	2
硬盘	8	27

相关指标数量

	A	B
节点精确率	1	0.83
指标类别精确率	0.57	0.58
提交根因正确数	4(/7)	7(/12)
总提交次数	349	488

用了该策略后的效果

异常发现→根因定位→提交策略

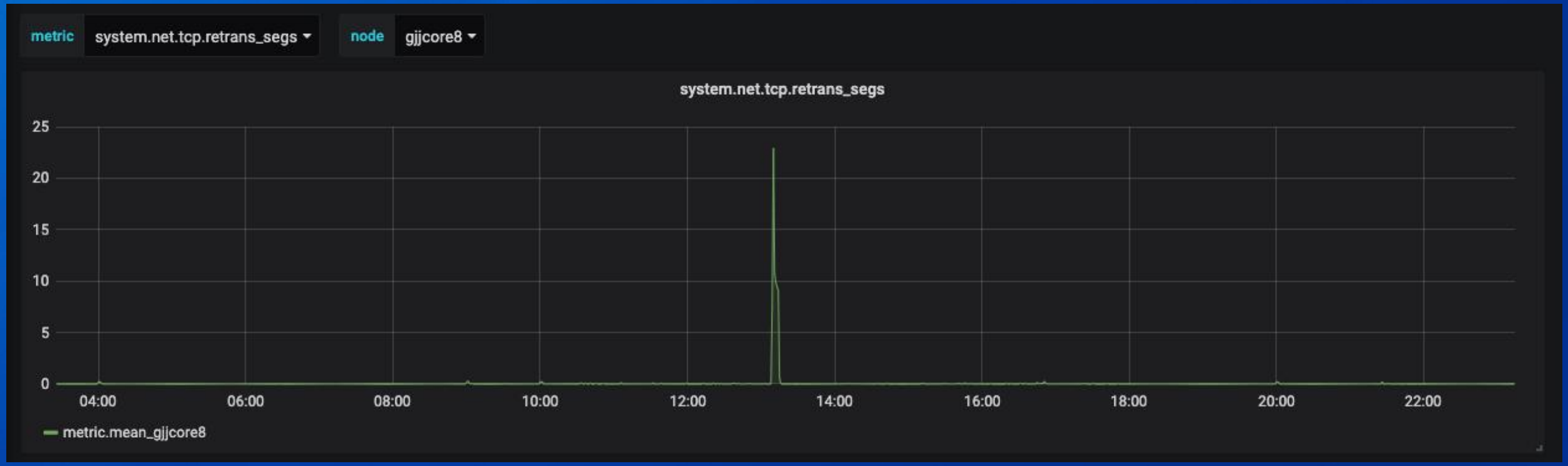
根因定位

- 核心要解决的问题：
  - 多节点、多指标类型告警信息排序
  - 约束异常检测产生的大量误报
- 解决思路：
  - 寻找每类故障的稳定性指标
    - 指标误报率低
    - 专家经验
      - （故障与指标关系能合理解释）
  - 稳定性指标出现故障是每类故障认定的前提

	A	B
CPU	cpu.pct_usage	CPUCpuUtil
内存	mem.pct_usage	TCP-FIN-WAIT, TCP-CLOSE-WAIT
网络	net.tcp.retrans_segs	ReadWrite
硬盘	mem.pct_usage	MEMUsedMemPerc

稳定性指标列表

异常发现→根因定位→提交策略



2021/2/26 1:09:00 PM

gjjcore8的网络丢包故障

异常发现→根因定位→提交策略

AIOps Challenge 智能运维挑战赛 | 科创会 NIICON

	A	B
节点精确率	1	0.83→0.91
指标类别精确率	0.57→1	0.58→0.83
提交根因正确数	4→7(/7)	7→10(/12)
总提交次数	349→82	488→94

使用该策略后的效果提升

异常发现→根因定位→提交策略

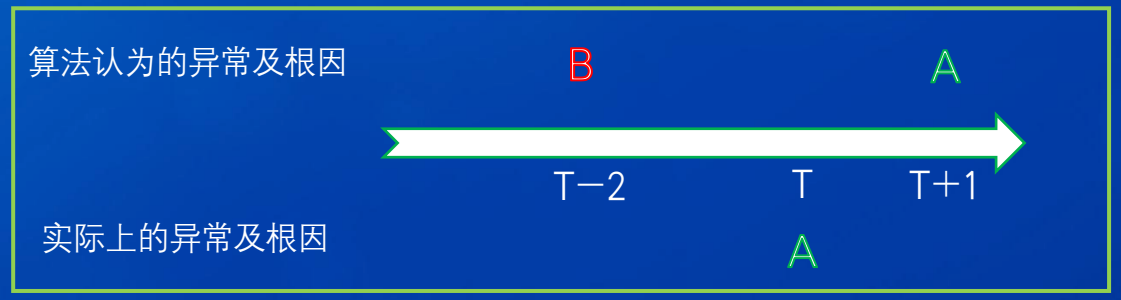
现有规则：

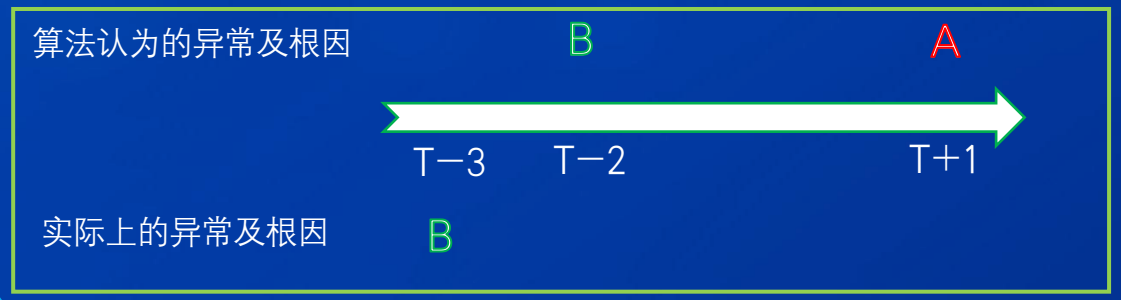
- 同时考虑precision、recall、检测延时，在precision大于一定阈值时才算分
- 故障发生10分钟内取最后一次提交

出现的问题：

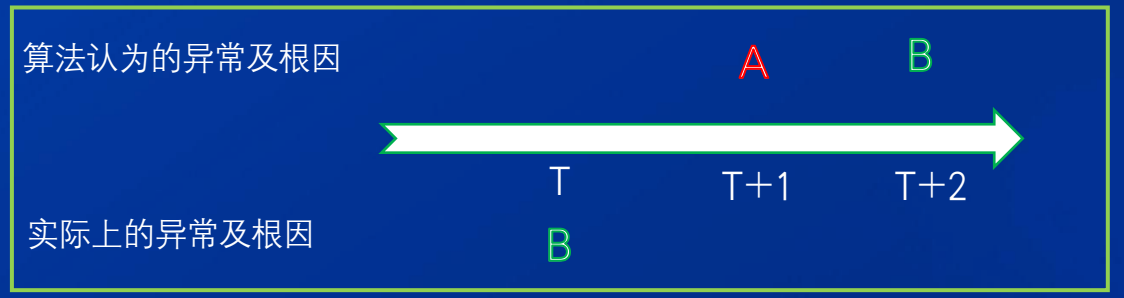
- 当算法在10分钟内给出两个不同的答案时，如何确定提交策略

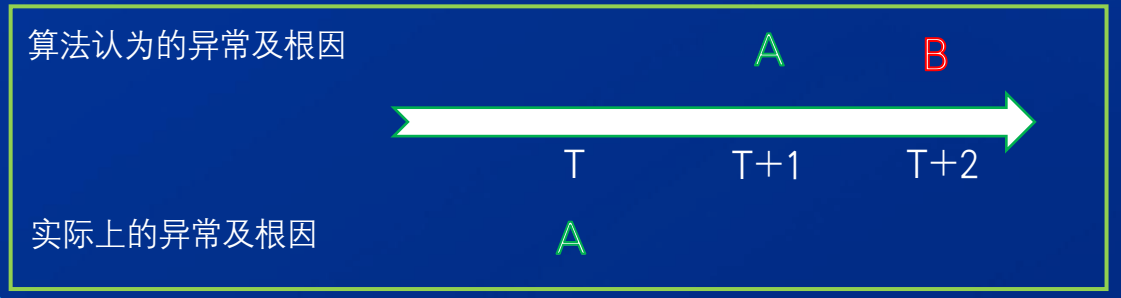
不同异常的根因是否应该覆盖？





针对同一个异常得到的根因不同要不要更新？

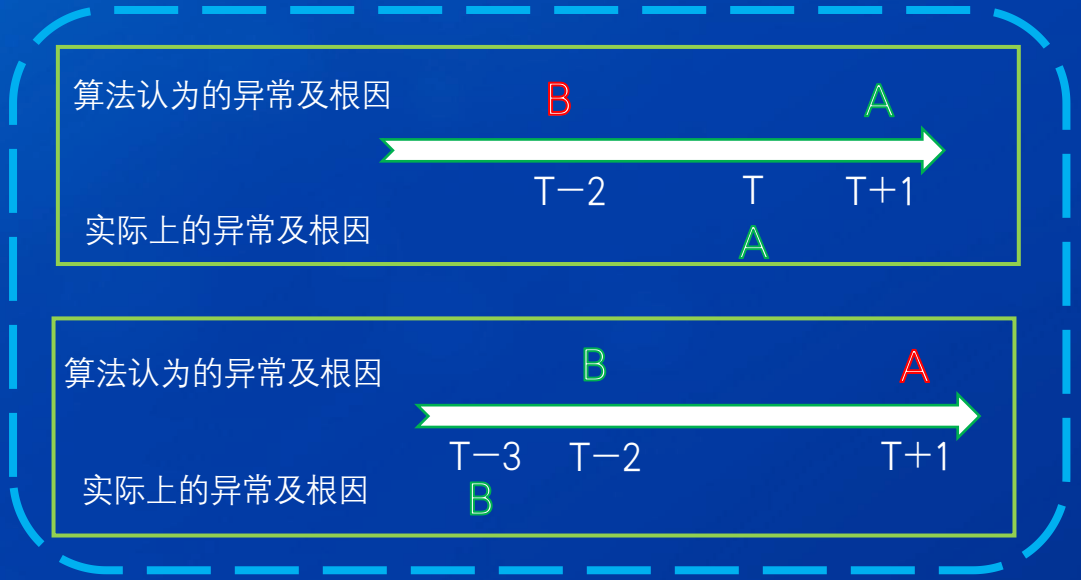




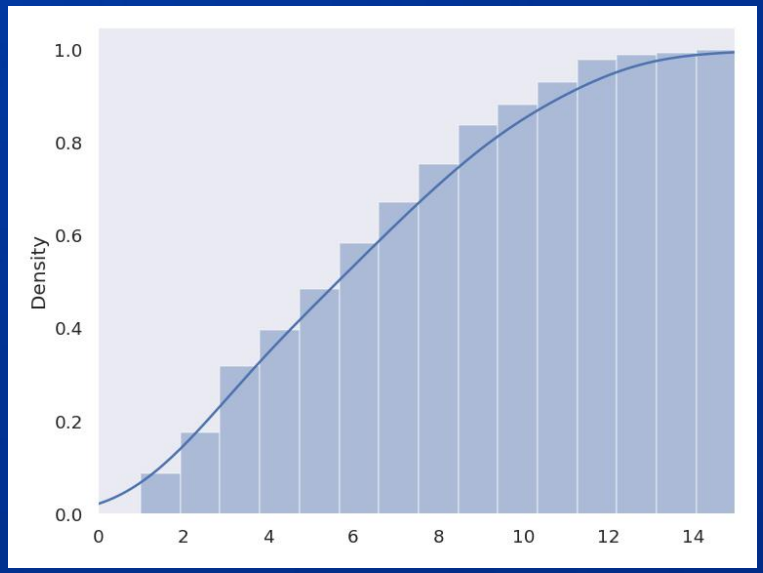
异常发现→根因定位→提交策略

应对策略：

- 在任何一类根因作为真正的根因都不超过precision_limit的条件下，将上次的结果（如果距离很近的话）与这次的合并一起报出
  - 牺牲每次的分数大小
  - 保全每次有分数保底
  - 与保底分大小有关



(20，0) → (15，15)



复赛每道题有分队伍的数量分布

复赛总共20支队伍

评价指标建议

- 10分钟内多次提交取最好
- 采用Top K precision作为评价结果正确性的指标

2021国际AIOps挑战赛决赛暨AIOps创新高峰论坛

THANKS

谢 谢 观 看

第一届国际互联网产业科技创新大会暨互联网创新产品展览会

The First International Internet Industry Science And Technology Innovation Conference & Internet Innovation Product Exhibition