

# 基于多源异构数据的异常智能分析

马宇驰

华为云 PaaS技术创新LAB (DevAI Lab负责人)





## 1. 背景与挑战

☆ 1.1 背景

☆ 1.2 现状与挑战

## 2. 方案介绍

☆ 2.1 嵌入式场景下异常分析方法

☆ 2.2 云化场景下异常分析方法

## 3. 总结

☆ 3.1 在研工作

☆ 3.2 总结



# 1. 背景与挑战



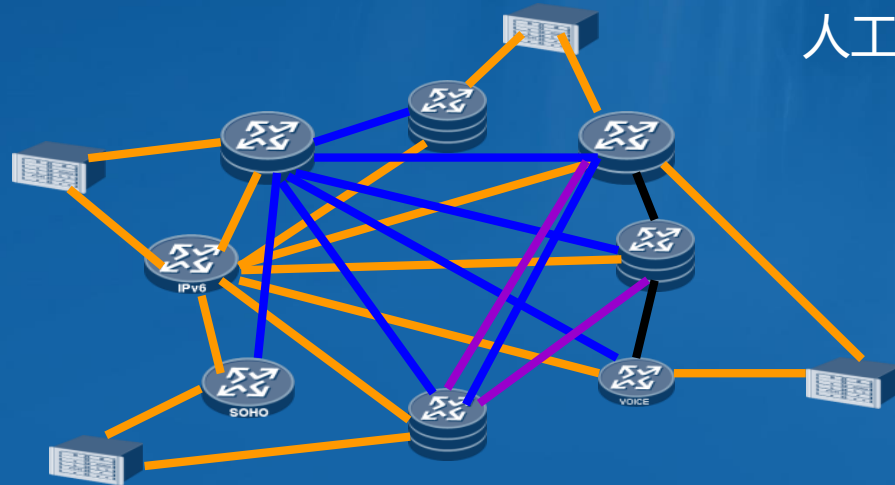


# CNCC 1.1 背景

※ 嵌入式场景：快节奏大规模的版本上线、测试资产不断积累、软硬件多态、多元组合场景下带来了大量失败和不稳定的测试用例，但分析失败用例工作量大、耗时长、重复度高，严重制约测试效率，难以满足快速质量反馈的诉求

☆ 以某嵌入式产品为例：

- 总逻辑用例数量：\*\*W+
- 自动化用例数量：\*\*W+
- 物理测试床：\*\*\*+ 个
- 被测设备：\* K+ 台
- 每天执行：\* W+ 例次
- 非Pass用例次：\* K+例次



人工标注量少、准确率低

质量要求高

用例多





# CNCC 1.1 背景

✧ 云化场景下：微服务架构与DevOps开发模式下微服务上线节奏快，快节奏大规模的版本上线与测试带来了大量失败和不稳定的测试用例，但分析单个失败用例耗时长达4人时（单微服务测试）或30人时（E2E集成测试），难以满足快速迭代的质量要求

- ☆ E2E集成测试（微服务集成复杂场景）：失败用例规模：150个，分析耗时：平均约30人时
- ☆ 单微服务测试（微服务API测试）：失败用例规模：150个，分析耗时：平均约4人时

用例分析存在瓶颈

快速质量反馈

服务快速发布



### 6.11.2 Release Report V1.0

测试结论

**262/266**    **161.8/851.2**    **4390/4436**

已交付需求/规划交付需求    新增DUI/累计缺陷DUI    执行成功测试点/总测试点

产品: CSB Solution    产品版本: 6.11.2    发布人: cWX394341    版本精度: versionL

1. CBC Solution 6.11.2特性继承CBC Solution 6.11.1特性。
2. 本文主要针对CBC Solution 6.11.2版本的特性或解决方案测试进行总结，文档内容包括测试数据统计、评估等。
3. 当前测试活动已结束，可验证的测试点TP已完成覆盖，需求已执行测试前4370个，已执行TP通过率99%，失败测试点、阻塞测试点及当前版本不可测试部分，均已提单跟踪，问题具体见遗留问题和需求变更发布场景，具体测试详情请参见：<http://hoshone.gkeetest.huawei.com/webstatic/csbtest/>
4. BI E2E基础特性自动以执行完成，接口自动化100%，类库自动化100%，无业务问题。
5. 测试活动发现的主要问题得到修正，遗留问题详见测试报告对应章节。
6. FE2021081700156国际站辅助需求——客户接口表、FE2021081700130国际站辅助需求——应收接口表、FE2021081700131国际站辅助需求——收入接口表等3个需求，实际随辅助cloud6.12.2版本发布，6.11.2版本仅为验证和辅助构造数据，且存在因产品不存在导致的场景构造异常情况。

功能度量

序号	需求单号	标题	责任服务	是否转IAV	状态	风险分析	分析人
1	FE2021081700130	国际站辅助需求——应收接口表	商业智能	是	不可发布	详情	蔡权祥 cWX394341
2	FE2021081700131	国际站辅助需求——收入接口表	商业智能	是	不可发布	详情	蔡权祥 cWX394341
3	FE2021081700135	国际站辅助需求——合同接口表	商业智能	否	不可发布	详情	余文彬 sKF76256
4	FE2021081700136	国际站辅助需求——客户接口表	商业智能	是	不可发布	详情	蔡权祥 cWX394341



# CNCC 1.2 现状与挑战

测试TSE

期望快速定界责任田

高质量交付

研发SE



版本经理

过点成本高

按时交付

研发PL



测试工程师

失败日志量大，  
多源日志种类多，分析难度大

产品日志与问题单  
协同处理难

研发工程师

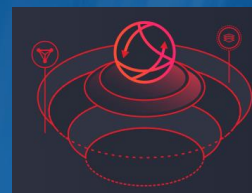






# CNCC 1.2 现状与挑战

- 小作坊时代，手工测试为主
- 部分商用工具进行专项测试
- 用例在Excel中管理
- 7x24测试自动化工厂模式建立，进入全面自动化时代
- 实现了管理工具、执行工具的集成拉通，测试桌面问世
- 执行效率大幅提升，产品验收周期从周到天



- 大数据及机器学习驱动，人工智能支撑辅助测试
- 精准测试，安全测试



- 功能、性能测试进入自动化时代
- 脚本开发，AW封装
- 测试用例管理工具、自动化执行器问世



- 大规模自动化并发执行，环境并发调度
- 测试与编译构建拉通，构建CD能力
- 验收周期从天到小时
- 使用先进技术，如人工智能 (AI)、机器学习 (ML) 和机器人过程自动化 (RPA)，来自自动化曾经由人类完成的任务





# CNCC 1.2 现状与挑战

质量做加法

VS

测试成本做减法

- ※ 挑战1：嵌入式场景测试量大，过点时间紧、任务重
- ※ 挑战2：云化场景下，微服务及中间件调用链复杂，定位难度大
- ※ 挑战3：云化场景下，版本迭代快，未知问题占主导，定位时间长





## 2. 方案介绍





# CNCC 2.1 嵌入式场景下异常分析方法

✧ 基于CLAIM的测试失败分析

✧ Composite Q-learning based Log Analysis fraMework



- 从日志管理工具数据库中导出历史日志和用户分析结论

- 数据清洗
- 基于用户词典和默认词典生成的字典树和隐马尔科夫模型进行分词

- 基于词频统计和多种文本嵌入方法，得到日志的特征向量

- Lazy Learning的方法解决样本稀疏及快速变化的类标问题

- 特征比对：基于日志的特征向量对比
- 滑动窗口：丢弃过期日志，避免干扰
- 强化学习：引入奖惩机制，趋利避害，持续学习
- 模型失效机制：弱化时间久远的经验



## Diff 模块

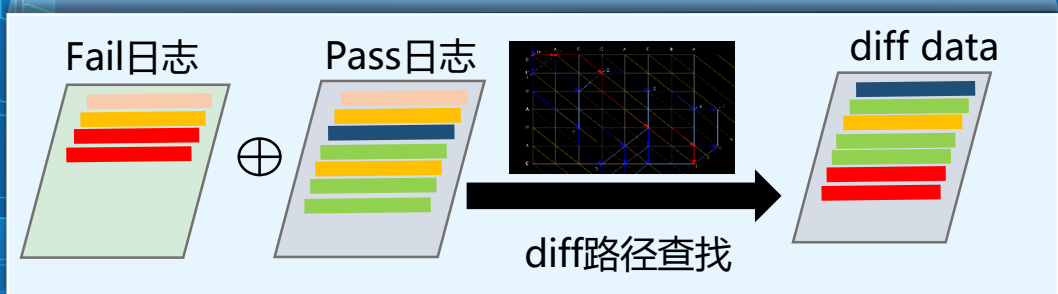
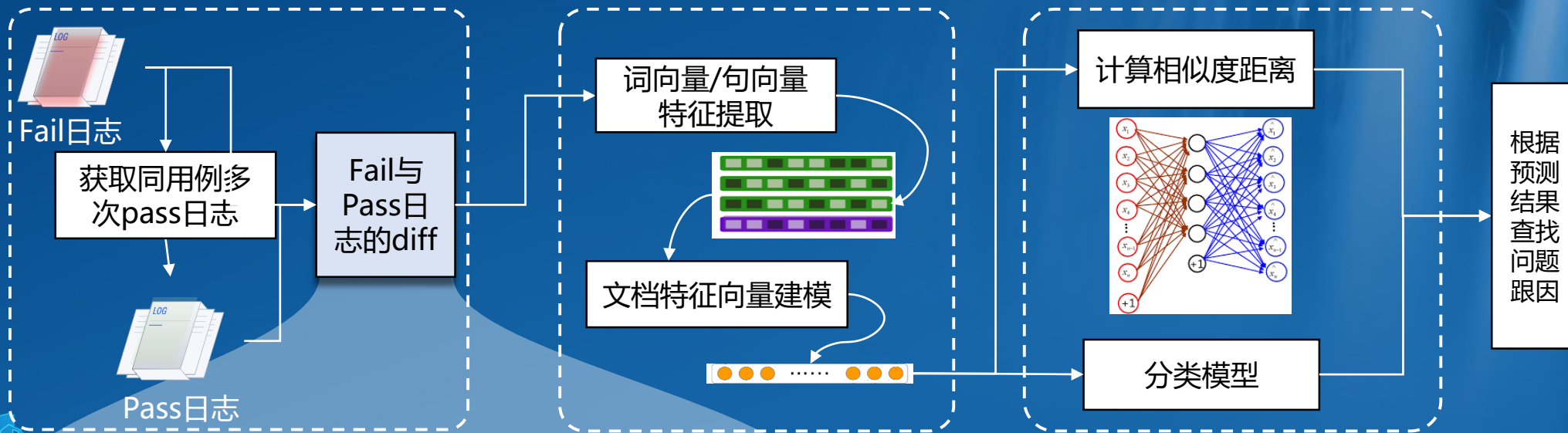
- ✓ 日志动态基准与当前失败日志差异对比
- ✓ 免用户规则

## 特征提取模块

- ✓ 差异特征提取
- ✓ 文档特征自动建模

## 模型训练

- ✓ 用例级、产品级分类模型训练
- ✓ 强分类器

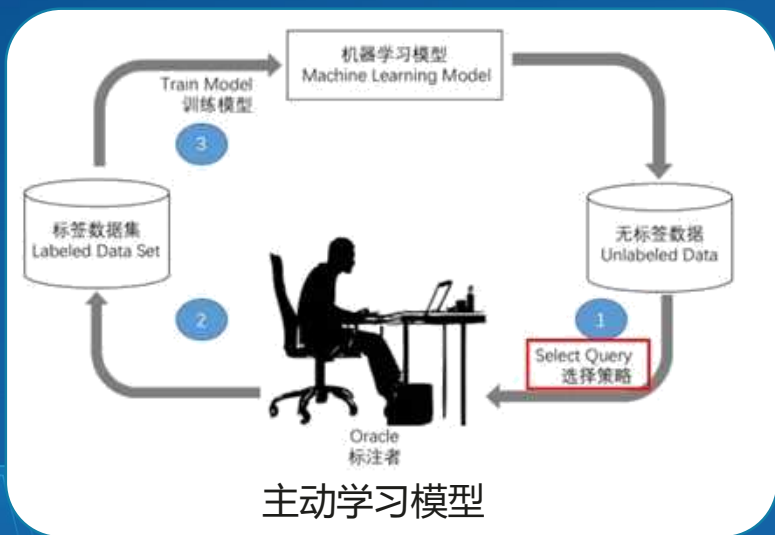


✧ 支持较少标注数据获取更高准确率：

✧ 基于 **置信度最低**、**边缘采样**、**信息熵** 最大三种选择策略，采用主动学习算法从训练数据集中主动地选择一些不带标签的数据进行标注，然后模型增量在带标签的数据上训练，从而通过少量的人工提升机器学习样本的质量。

与现有流程相比

- 更少的人工分析样本，相近的准确率
- 相近的人工样本数量，更高的准确率



## 置信度最低

在多分类问题场景下，每条日志智能分析的输出除了类别信息外还有置信度信息。同等情况下，优先选择置信度低的样本进行人工标注和分析。

## 边缘采样

边缘采样是指极容易被判定成两类的日志样本，或者这些数据样本被判定成不同类的概率相差不大。边缘采样就是选择模型预测最大和次大的概率差值最小的日志样本

## 信息熵最大

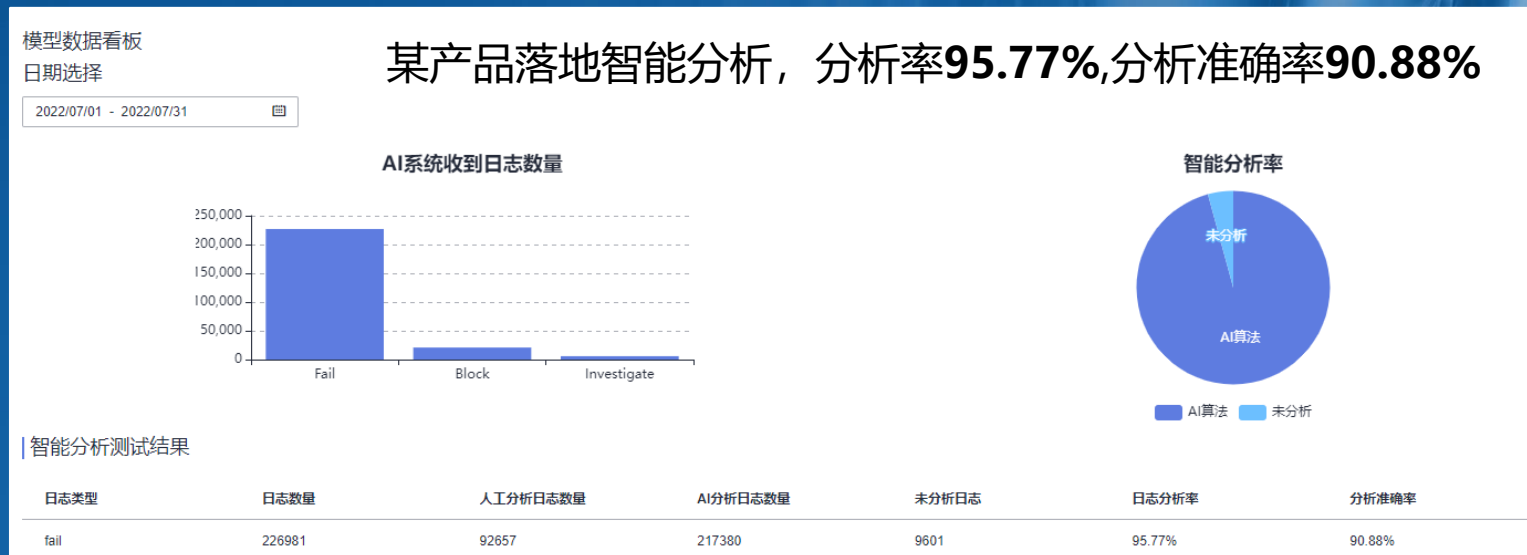
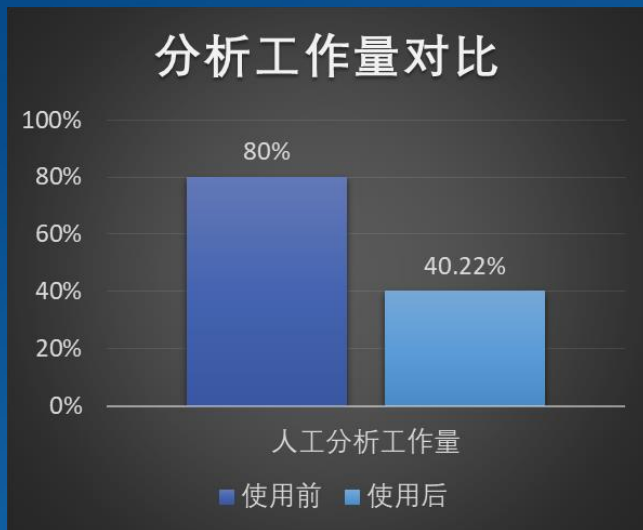
信息熵越大表示系统的不确定性越大，信息熵越小表示系统的不确定性越小。在失败测试日志分析的多分类场景下，可以选择熵比较大的日志样本作为优先标记数据

TestCase Name	TestCase Number	Task Name	Test Suite	BVersion Name	Test Result	Major Cause	Minor Cause	Ticket Number	Manual Analyzer
588x芯片转发至包身识别...	RT_CASE_Intelligent_Moni...	1TA_none_1204_2021-12...	/Router CarrierEth-AutoFac...	NE5000E V800R021C10 A...	Fail	支撑库问题	支撑库错误	RouterEnv2021121415155761	100630935
默认情况下不支持远程镜像...	RT_CASE_MIRROR A88309	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	版本缺陷	版本缺陷	DTS2021081324670	100630935
脚本策略模板下命令行增删...	RT_CASE_AAA A80457	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	环境问题	环境准备问题		100630935
脚本策略模板下命令行增删...	RT_CASE_AAA A80457	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	环境问题	环境准备问题		100630935
城下增值业务计费为radius...	RT_CASE_BRAS_VALUE...	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	脚本问题	脚本写作错误	RouterEnv2021120416140794	wWX975520
城下增值业务计费为radius...	RT_CASE_BRAS_VALUE...	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	脚本问题	脚本写作错误	RouterEnv2021120416140794	wWX975520
城下增值业务计费为radius...	RT_CASE_BRAS_VALUE...	236_1711_1203_1fq_none...	/Router CarrierEth-AutoFac...	NE40E V800R021C10 AUT...	Fail	脚本问题	脚本写作错误	RouterEnv2021120416140794	wWX975520



# CNCC 2.1 嵌入式场景下异常分析方法：落地效果

\* 单个产品落地，机器分析分析\*\* .7万份日志/月，智能分析率达到95.77%，免人工复核占比57.37%



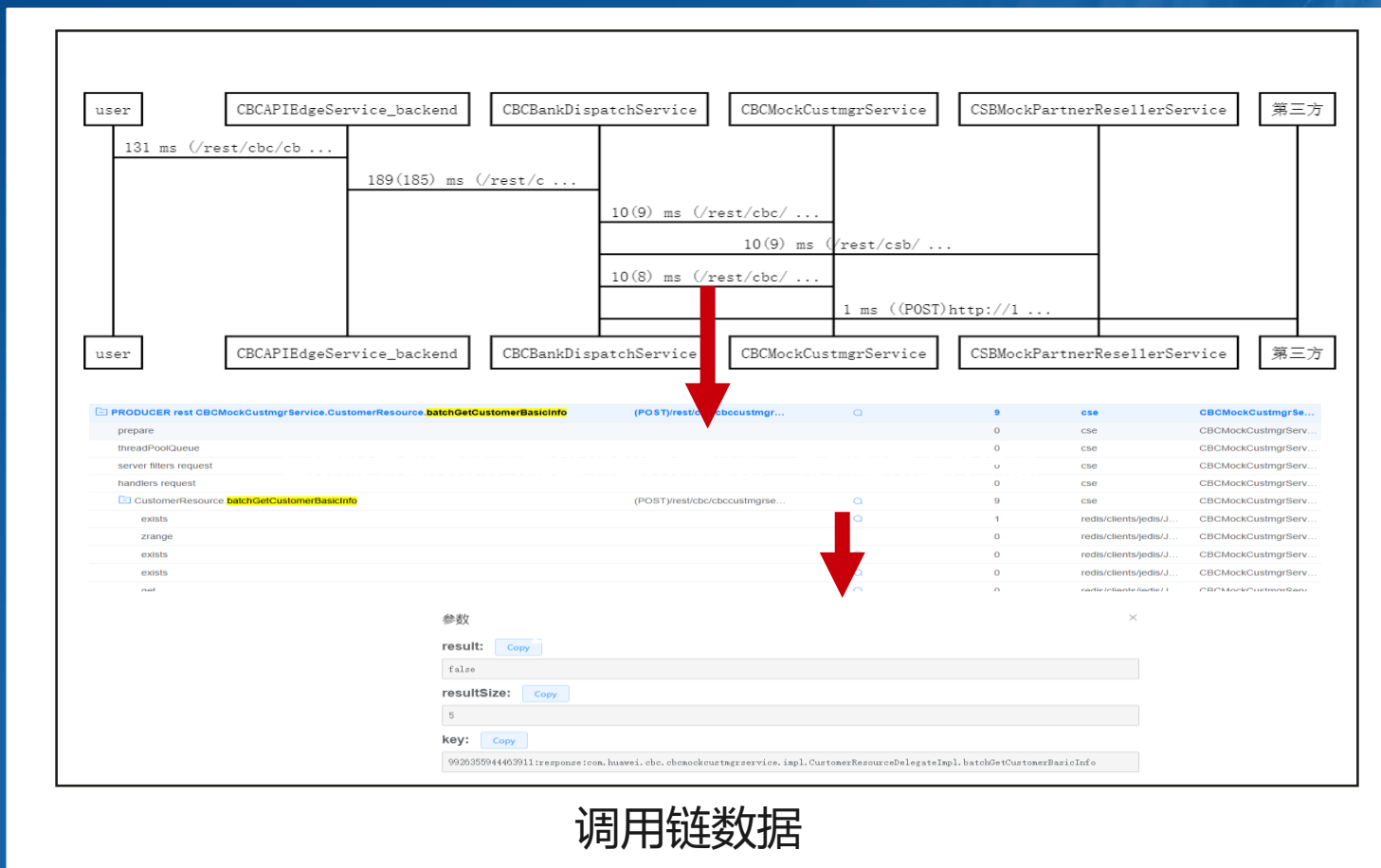




CNCC

## 2.2 云化场景下异常分析方法： 基于调用链日志的无监督分析

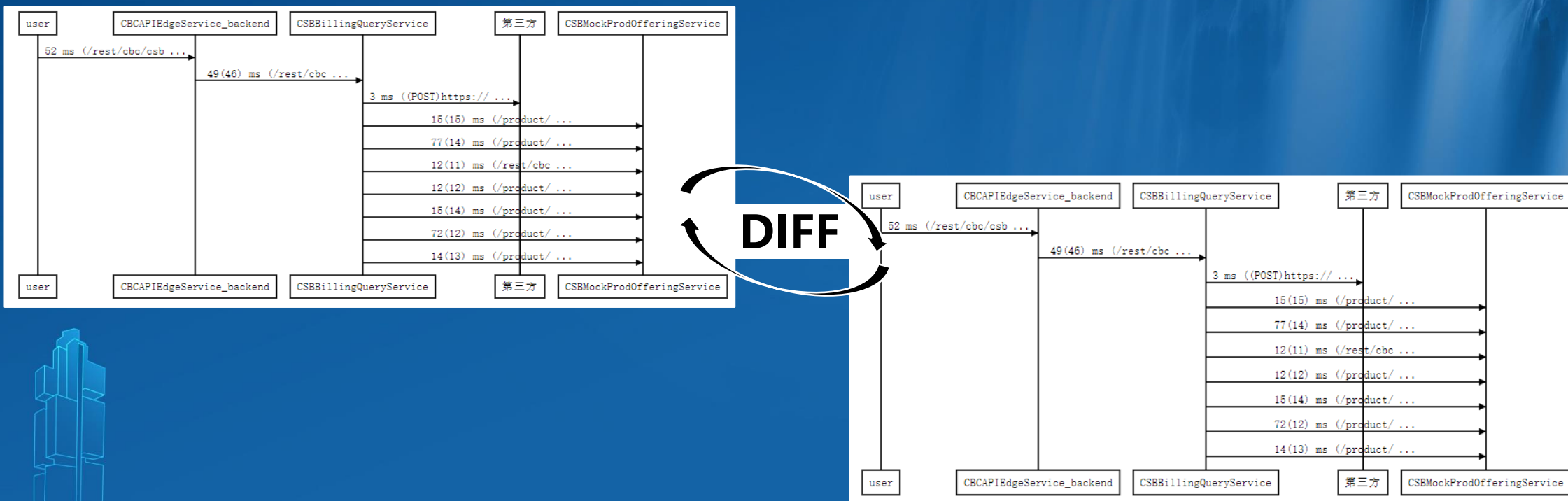
- 一个完整调用链由一系列服务调用组成，服务调用由一系列调用事件组成
- 导致测试失败的根因会反映在调用链数据变化，调用链自动化分析关键在于提取其正常模式





# 2.2 云化场景下异常分析方法： 基于调用链日志的无监督分析

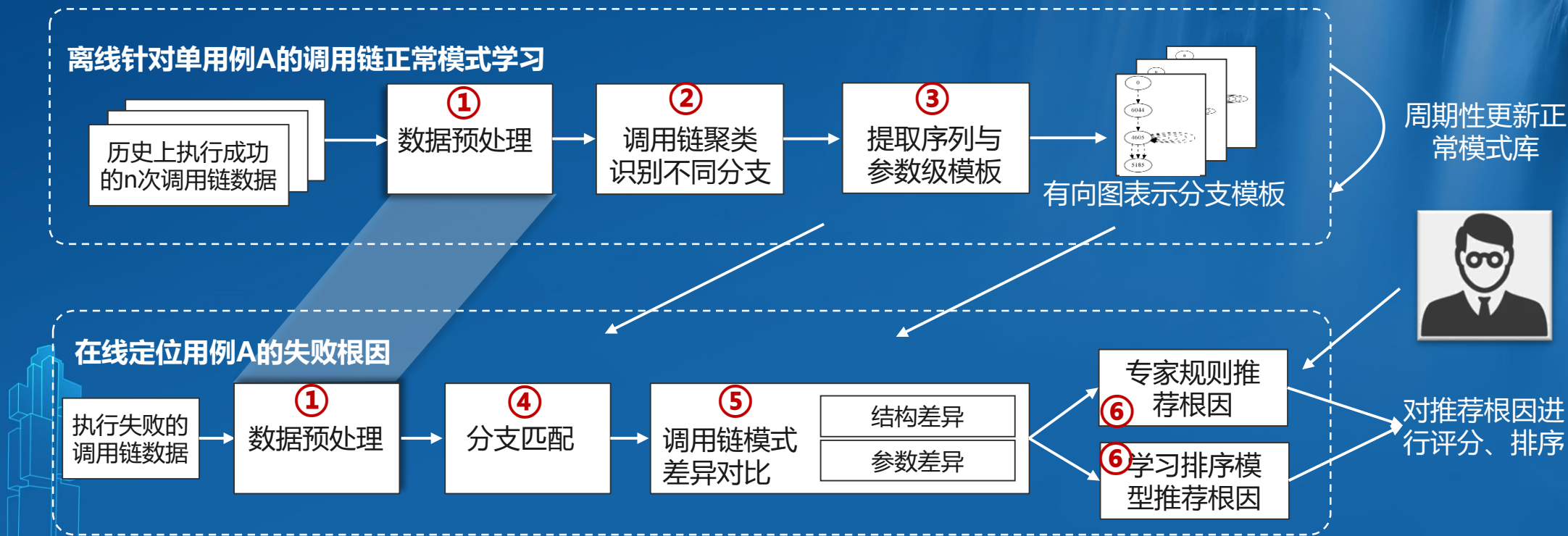
- 代码执行逻辑导致的调用链模式不唯一（重试、并行、缓存机制、采集缺失、依赖参数）
- 版本迭代快 -> 用例变更频繁 -> 调用链正常模式不稳定 -> 同一模式的样本量稀疏





## \* 基于有向属性图的失败用例细粒度根因定位算法

- 数据驱动下自动化提取调用链参数级正常模式，识别失败用例的调用链异常点推荐潜在根因





# 2.2 云化场景下异常分析方法： 基于调用链日志的无监督分析

## \*Step1: 离线学习模块

### ① 数据预处理:

对用例的历史n条成功执行例次  
相关数据记录, 进行预处理

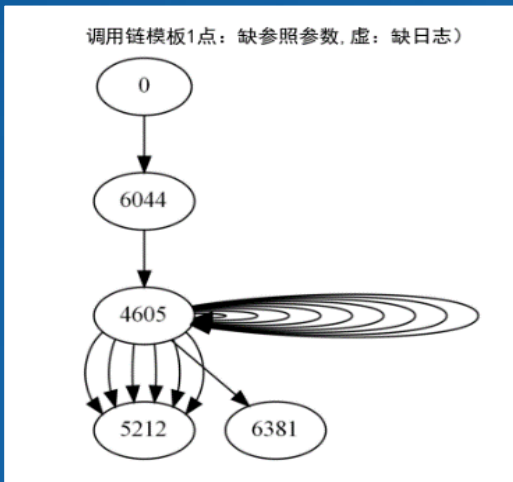
### ② 调用链聚类识别不同分支:

根据调用链特征将调用链聚类,  
分成不同正常分支

### ③ 提取序列与参数级模板:

根据参数类型进行解析, 记录时延统计量和  
通过文本模式匹配技术提取参数正常模式

Details	Script	Result	History Info
Name	Result	Reason of Fail	Author CreateTime
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-12 20:57:54
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-12 19:33:21
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-12 19:17:30
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-12 09:51:07
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-12 03:56:21
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-11 17:07:50
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-10 15:10:04
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-10 15:01:19
master Result2021-...	Failed	FAILURE	PipeLine 2021-10-10 14:50:48
master Result2021-...	Passed		PipeLine 2021-10-10 12:12:51
master Result2021-...	Passed		PipeLine 2021-10-10 12:05:40
master Result2021-...	Passed		PipeLine 2021-10-10 11:56:48
master Result2021-...	Passed		PipeLine 2021-10-10 11:45:11
master Result2021-...	Passed		PipeLine 2021-10-10 03:52:46



```

'parameters': {'result': 'true',
               'resultSize': '4',
               'key': '@0rsjqvs8j7,FA6C1F2B2C39412FAB9746A12EBED2CA:response:com.huawei.csb.csbmock
'parameters_invariant_parts': {'result': '*',
                                'key': '@0rsjqvs8j7,*:response:com.huawei.csb.csbmocksrmervice.impl
  
```

```

'parameters': {'value': '/rest/cbc/csbsrmervice/v1/partner/trade_mode',
               'key': '@0rsjqvs8j7,FA6C1F2B2C39412FAB9746A12EBED2CA:request:path:com.huawei.csb.csbmo
'parameters_invariant_parts': {'value': '/rest/cbc/csbsrmervice/v1/partner/trade_mode',
                                'key': '@0rsjqvs8j7,*:request:path:com.huawei.csb.csbmocksrmervice.im
  
```

.....



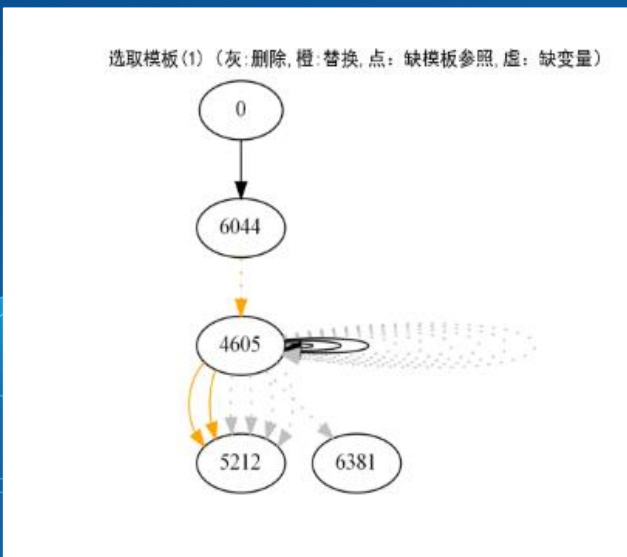


## \*Step2: 在线定位模块

**④ 分支匹配：**计算相似度，匹配经过服务节点最相似以及服务调用顺序最一致的分支模板

**⑤ 调用链模式差异对比：**检测调用链模式差异，包括结构差异和（新增和缺失服务调用）和参数差异

**⑥ 根因排序：**基于专家规则和机器学习模型相结合的方式，进行跟因排序，推荐根因结果



参数	推荐优先级分数
mysqlResult	5
result	5
argument	5
responseBody	4
sql	4
originSql	4
key	4
value	4
resultCode	3
code	3
bizCode	3
requestBody	2
responseHeader	1
requestHeader	1
DBName	1
其他非忽略参数	1

1. Redis exists的key发生变化;
2. queryOOperatorByPartyId接口的responseBody变成空值
3. requestHeader的host发生改变

```

失效日志调用链智能分析结果
#####
#
caseUri: 00rsjqvrf9q
成功提取不同分支的调用模板: 1/2
是否提取模板成功: [False, True]
用于提取模板的成功调用链信息:
模板与检测调用链的相似度分数: [0.939, 0.934]
检测选取模板: 1
    
```

00rsjqvrf9q\_rca.txt



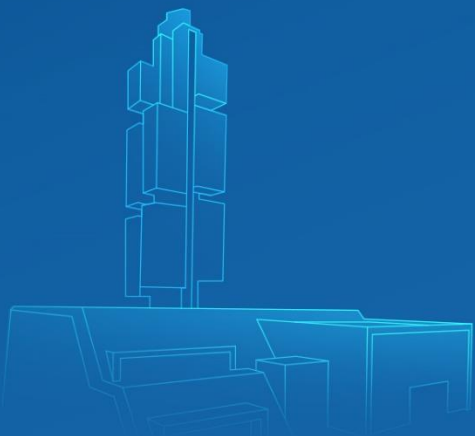
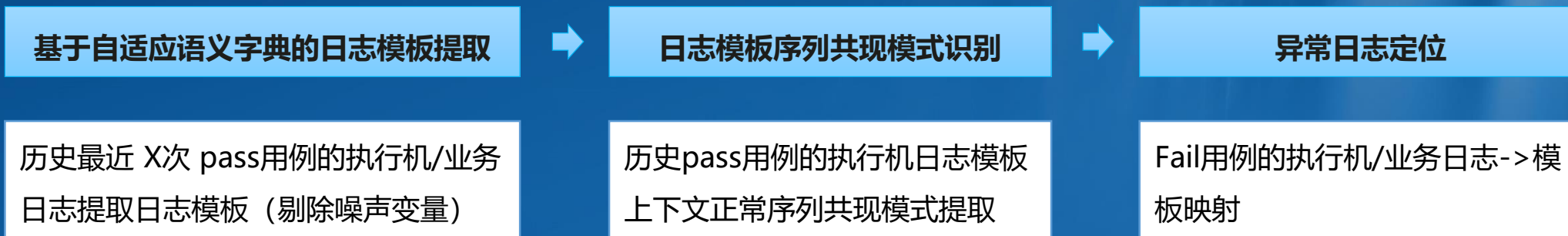


CNCC

## 2.2 云化场景下异常分析方法：基于业务日志的无监督分析方法

### \* 基于模板提取的高鲁棒异常定位算法

- ☆ 通过预处理模板化日志，学习单行日志的固定模式，降低数据分析维度，进而学习正常日志间的序列共现模式用于定位异常





# 2.2 云化场景下异常分析方法： 基于业务日志的无监督分析方法

## \* Step1: 基于自适应语义字典的日志模板提取

### 基于自适应语义字典的日志模板提取

历史最近 X次 pass用例的执行机/业务日志提取日志模板 (剔除噪声)

海量原始日志 (时间戳、等级、模板/常量、其余变量) :

```
17/06/09 20:11:11 INFO storage.BlockManager: Found block rdd_42 locally
17/06/09 20:15:11 INFO storage.BlockManager: Found block rdd_21 locally
17/06/09 20:20:11 INFO storage.BlockManager: Dropping block rdd_42 from memory
```

1. 日志模板中常量与变量存在频率、语义上差别, 通过启发式方法自适应扩充语义字典, 尽量将包含语义信息的以及不变的部分视作日志语句中的常量

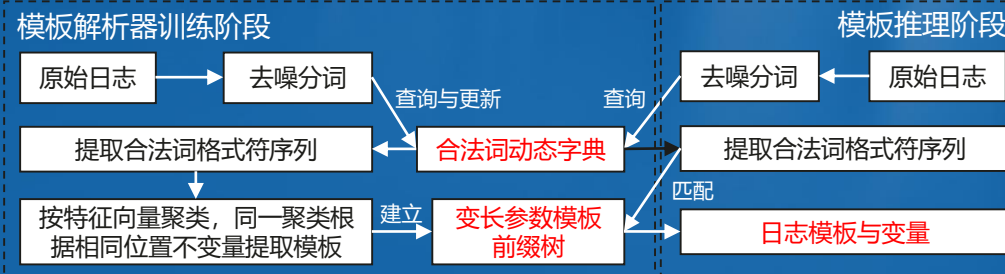
2. 并通过前缀树实现变长变量的模板推理与提升推理效率

通过有限的日志模板表示

```
storage.BlockManager: Found block <*> locally
storage.BlockManager: Found block <*> locally
storage.BlockManager: Dropping block <*> from memory
```

变量

```
rdd_42
rdd_21
rdd_42
```



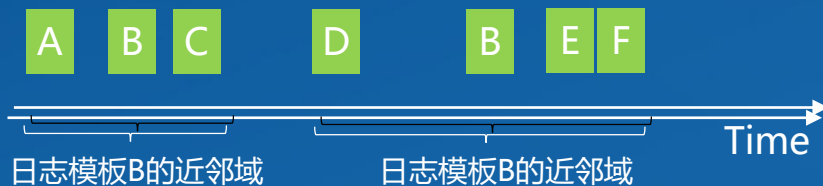
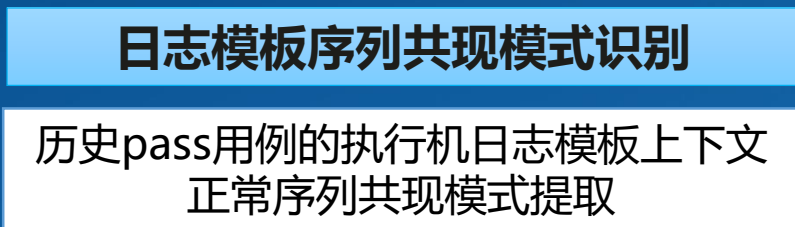
日志解析算法: 线性时间复杂度流式更新与推理, 超参数不敏感



## 2.2 云化场景下异常分析方法：基于业务日志的无监督分析方法

### \* Step2: 日志模板序列共现模式识别

- ☆ 日志反映了程序执行逻辑，日志之间顺序满足上下文约束。
- ☆ 基于时间步长的自适应滑动窗口，从历史正常例次的日志中统计每个日志模板邻近域内与其他日志模板的共现频率
- ☆ 学习日志间上下文转移关系，用于检测异常



	A	B	C	D	E	F
A	0	1	1	0	0	0
B	0.5	0	0.5	1	0.5	0.5
...						
F	0	0	...		1	0

高鲁棒：已有的（深度学习）序列建模方法容易受日志偶现型随机乱序、缺失噪声影响，基于共现频率模式的方法能避免此类误报，提升日志异常检测准确率。



# 2.2 云化场景下异常分析方法： 基于业务日志的无监督分析方法

## \* Step3: 异常日志定位

☆ 通过异常降噪、排序与正常日志上下文对比，更有利于理解异常产生原因与表现，进而进行异常日志定位

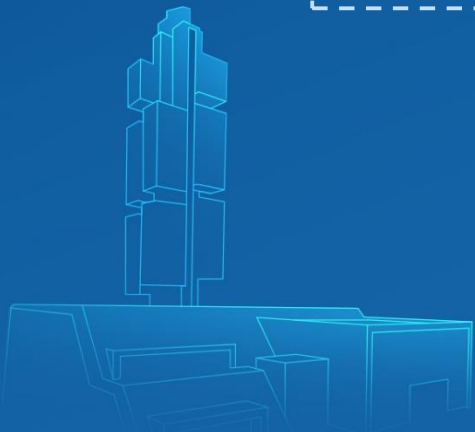


异常日志模板序列根据共现频率矩阵判断DIFF

- ① 全新异常：出现了测试成功时没有的日志模板；  
A B C 变成 A G
- ② 冗余异常：出现了测试成功时没有的两个日志共现情况；  
A B C 变成 A E B C
- ③ 缺失异常：缺少了测试成功时上下文中必须出现的日志；  
A B C 变成 A C



异常降噪、排序与正常日志上下文对比，更有利于理解异常产生原因与表现







CNCC

# 2.2 云化场景下异常分析方法： 落地效果

✧ 以某产品测试执行为例：失败用例91个，聚类后待分析的数量为13类

HC_导出任务	1	已完成	13	91	0%	0%	276
53.8%	AW名称: newDBQuery	涉及用例数: 49	已分析: 0	关键词: t_query_export_task \$filestatus			
27.5%	AW名称: newDBQuery	涉及用例数: 25	已分析: 0	关键词: t_query_export_task AssertionError			
3.3%	AW名称: yamIAPI	涉及用例数: 3	已分析: 0	关键词: get /rest/.../bill/500 framework.remote SystemError \$code			
3.3%	AW名称: newDBQuery	涉及用例数: 3	已分析: 0	关键词: t_query_export_file \$bucketname			
2.2%	AW名称: checkOBSFile	涉及用例数: 2	已分析: 0	关键词: financ...eductinfo csv Get property fail			
2.2%	AW名称: checkOBSFile	涉及用例数: 2	已分析: 0	关键词: financ...rce csv Get property fail			
1.1%	AW名称: GetReqByKey	涉及用例数: 1	已分析: 0	关键词: com.huawei.csb.csbmockcreditmgrservice.impl.Custom			
1.1%	AW名称: checkOBSFile	涉及用例数: 1	已分析: 0	关键词: financ...57776C7 csv			
1.1%	AW名称: GetRequest	涉及用例数: 1	已分析: 0	失败推测: 外部Mock预制的接口未被调用, 请检查			
1.1%	AW名称: newDBQuery	涉及用例数: 1	已分析: 0	关键词: t_query_export_task \$extendparam5 cons			
1.1%	AW名称: checkOBSFile	涉及用例数: 1	已分析: 0	关键词: financ...spendings csv			
1.1%	AW名称: checkOBSFile	涉及用例数: 1	已分析: 0	关键词: financ...gbao05 csv			
1.1%	AW名称: checkOBSFile	涉及用例数: 1	已分析: 0	关键词: financ...endings csv			

## 用例画像&差异显示差异

**用例画像**

- 有差异
- 有差异
- 有差异

业务配置

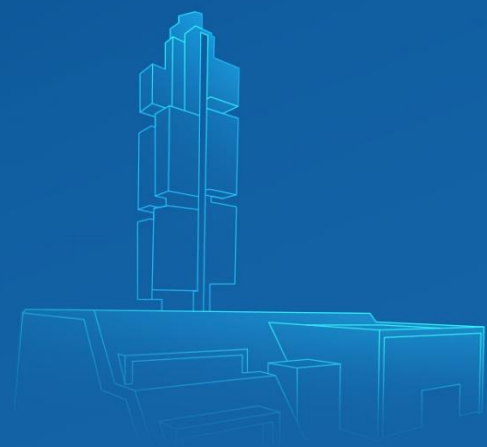
用例脚本

---

**智能推荐**

失败原因摘要	失败原因详情	失败原因打分
CSE Service服务未创建异步执行线程	线程类: java.util.concurrent.FutureTask	100
CSB Service服务少调了一个sql	sql:UPDATE t_..._task SET ...	66
CSB Service服务少调了一个sql	sql:SELECT id, cust..., kid...	33

## 推荐出失败原因





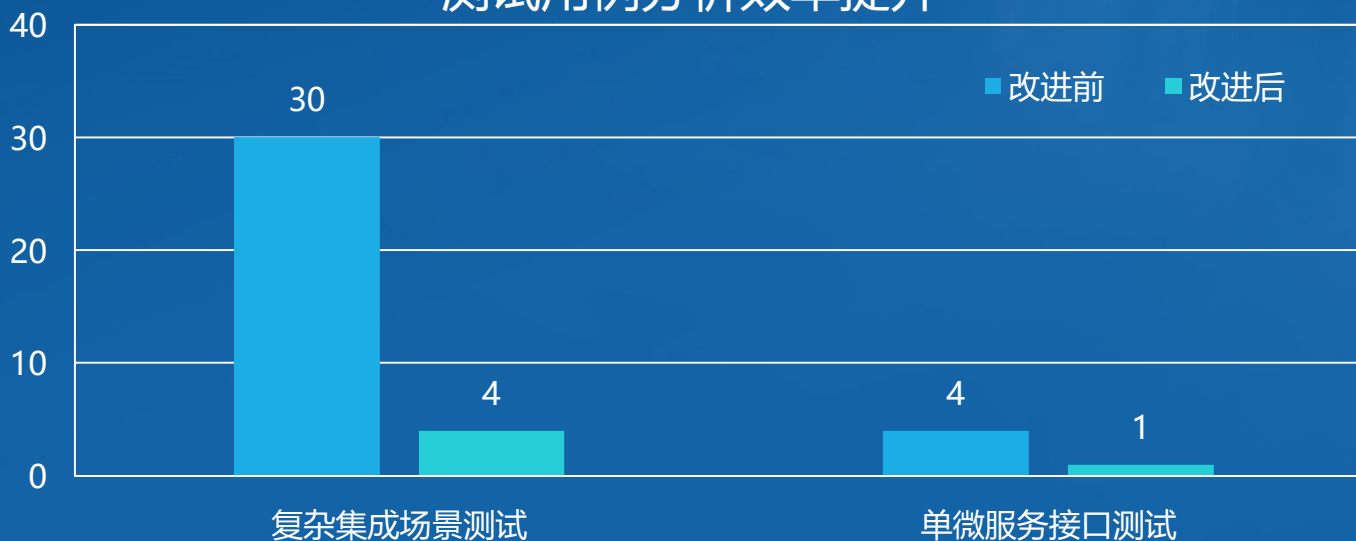


CINCC

## 2.2 云化场景下异常分析方法： 落地效果

- ※ 复杂集成场景失败用例分析耗时从30小时降低至4小时（以150个失败用例规模计算），**分析效率提升86%**
- ※ 单微服务接口测试失败用例耗时分析分析从4小时降低至1小时（以150个失败用例规模计算），**分析效率提升75%**

测试用例分析效率提升



# 3. 在研工作与总结





# CNCC 3.1 在研工作：多态日志关联分析方法

## ✧ 跨用例 谱分析计算根因的异常分数+失败用例聚类

### 失败用例聚类

1. 同一测试任务多用例**多态日志DIFF列表**，进一步模板化聚类为根因事件，对DIFF日志行列表再次进行模板化处理
2. 计算TFIDF相似度，将相似度大于阈值的DIFF日志行模板聚类为根因事件



### 构建根因事件异质图

1. 构建 ‘用例-根因事件’ 的异构图, 通过High-order Personalized PageRank (HPPR) 得到每个根因事件的异常权重
2. 通过加权的谱分析计算每个根因事件类的异常分数



### 加权谱分析计算根因事件排序

1. 加权的谱分析计算根因事件异常分数排序
2. 相同TOP根因的用例聚类，相同TOP根因事件的失败用例会被聚合为同一类失败原因导致的用例，将根因结果推荐给用户



## \*Step1: 失败用例聚类

同一测试任务多用例多态日志DIFF列表  
进一步模板化聚类为根因事件

日志来源: 测试日志, 产品日志, 调用日志, 代码覆盖 ...

(成功) 用例T1的DIFF日志行: L3

(失败) 用例T2的DIFF日志行: L2, L4

(失败) 用例T3的DIFF日志行: L1, L5, L6

- L2: bpcustomerapplyfailmanager.addapplyfailinfo  
start,bpid=2bb945231,customerid=686ca,cooperatio  
ntype=3,errorcode=1
- L5: bpcustomerapplyfailmanager.addapplyfailinfo  
start,bpid=2bd332131,customerid=126cx,cooperatio  
ntype=2,errorcode=1



模板化并根据TF-IDF向量进行层次聚类

(成功) 用例T1的DIFF日志行: E3

(失败) 用例T2的DIFF日志行: E2, E3

(失败) 用例T3的DIFF日志行: E1, E2, E4



度量用例相似性进行失败用例聚类

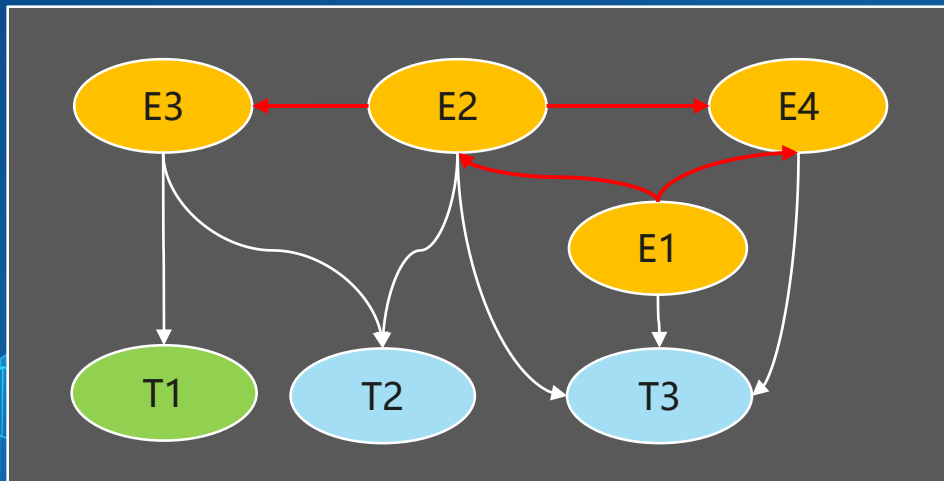


## \*Step2: 构建根因事件异质图

构建‘用例-根因事件’的异质图  
计算异常权重



求解根因事件的异常权重，  
再进行谱分析异常分数计算



1. 用例权重：如果一个失败用例只包含很少根因事件，那么这个用例需要排查的异常范围更小，在计算异常分数时权重应该更高；
2. 根因事件权重：如果一个根因事件被多个权重高的失败用例覆盖，那么这个根因事件更可能是根因，权重也应该高；
3. 发生时间早或是含有异常关键字的根因事件权重更高；





# CNCC 3.1 在研工作：多态日志关联分析方法

## \*Step3: 加权谱分析计算根因事件排序

加权谱分析计算根因事件异常分数排序与相同TOP根因的用例聚类

谱分析异常分数公式，反映根因事件导致测试失效的比例

根因事件谱

根因事件\用例	T1	T2	T3
E1			
E2			
E3			
E4			
测试结果	Pass	Fail	Fail

Formula	Definition	Formula	Definition
Dstar2	$\frac{O_{ef}^2}{O_{ep}+O_{nf}}$	Ochiai	$\frac{O_{ef}}{\sqrt{(O_{ef}+O_{ep})(O_{ef}+O_{nf})}}$
Goodman	$\frac{2O_{ef}^2-O_{nf}-O_{ep}}{2O_{ef}+O_{nf}+O_{ep}}$	Sørensen	$\frac{2O_{ef}}{2O_{ef}+O_{nf}+O_{ep}}$
Jaccard	$\frac{O_{ef}}{O_{ef}+O_{nf}+O_{ep}}$	RussellRao	$\frac{O_{ef}}{O_{ef}+O_{nf}+O_{ep}+O_{np}}$
M2	$\frac{O_{ef}}{O_{ef}+O_{np}+2O_{ep}+2O_{nf}}$	Dice	$\frac{2O_{ef}}{O_{ef}+O_{ep}+O_{nf}}$

根因事件	异常分数
E1	0.4
E2	0.9
E3	0.2
E4	0.6

每个用例的DIFF日志行依据根因事件的异常分数排序，相同TOP根因的用例聚为一类



# CNCC 3.2 总结

※ 挑战1：嵌入式场景测试量大，过点时间紧、任务重

☆ 嵌入式场景下异常分析方法

- 基于CLAIM的测试失败分析
- 基于DIFF的特征学习
- 主动学习

※ 挑战2：云化场景下，微服务及中间件调用链复杂，定位难度大

☆ 云化场景下异常分析方法：基于调用链日志的无监督分析

- 基于有向属性图的失败用例细粒度根因定位

※ 挑战3：云化场景下，版本迭代快，未知问题占主导，定位时间长

☆ 云化场景下异常分析方法：基于业务日志的无监督分析方法

- 基于模板提取的高鲁棒异常定位算法



基于同源数据的DIFF并特征化，形成统一的多源异构分析方法：**多源异构日志关联分析方法**

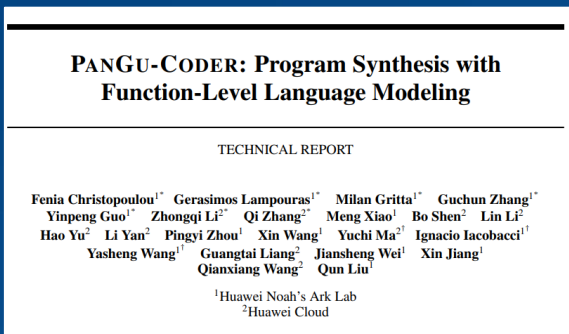




# CNCC 相关高校合作

合作老师	合作高校	合作课题
贺品嘉	香港中文大学 (深圳)	在线异常检测及定位算法研究 <b>二期</b>
董震	复旦大学	历史数据驱动的Flaky测试检测与根因定位技术研究 <b>New</b>
张圣林	南开大学	基于知识图谱的多态失败日志根因定位机制研究 <b>New</b>





华为云PaaS技术创新Lab×诺亚语音语义Lab发表文章：**多项重要指标国际领先**



PANGU-CODER受到国内外媒体广泛关注与报道



【CNCC 2022专属白名单通道】  
欢迎大家扫码注册华为云账号  
申请成为CodeArts Snap内测用户！

NEW

智能编程助手

## CodeArts Snap

@软件开发生产线CodeArts

- CodeArts Snap能根据中英文描述生成完整的函数级代码；可以替代重复繁琐的人工编码，高效生成测试代码；还能提供代码的自动检查和修复，其检查准确率达到业界领先。



自然语言

智能编程助手  
CodeArts Snap

编程语言

函数级代码生成 | 单元测试用例生成 | 代码检查与修复



CNCC

谢谢!

