北京交通大學 | 电子信息工程学院
School of Electronic and Information Engineering

机器学习和人工智能的应用

知识驱动的铁路智能运维

网络智能实验室

网络智能与信息安全研究所

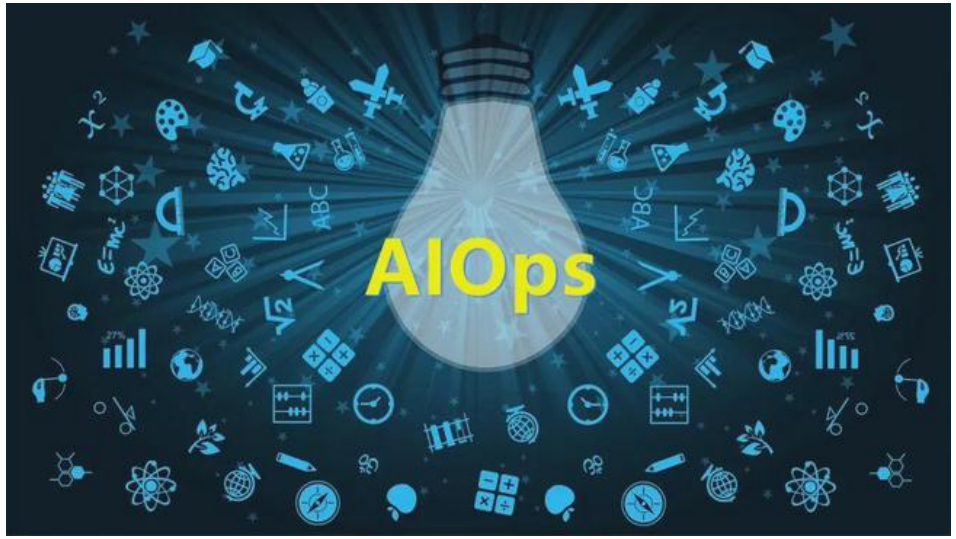
大纲

- 背景
- 监测
  - 日志
  - 指标
- 诊断
  - 异常检测
- 推理
  - 根因分析
  - 知识图谱

智能运维时代已经到来

手工运维→自动化运维→运维开发一体化→智能运维

互联网系统数据规模急剧膨胀
服务类型复杂多样
人工智能的高速发展
} 智能运维(AIOps)



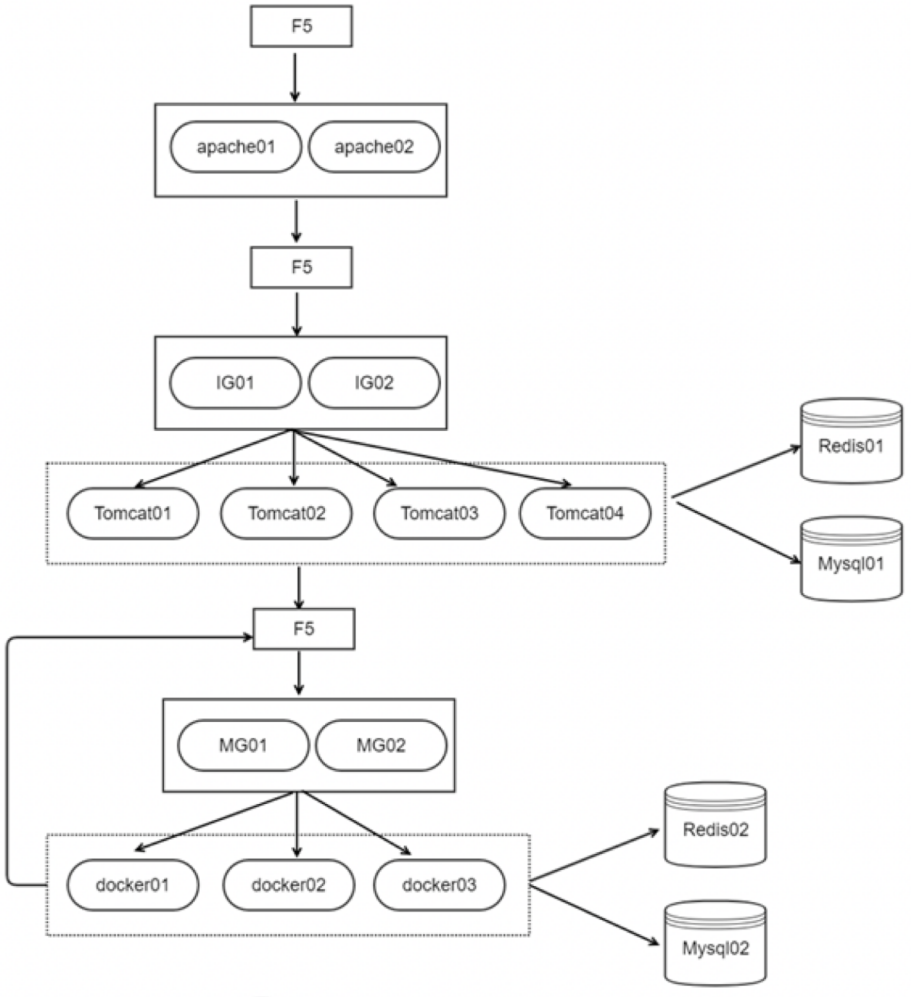
伴随着通信系统升级转型、人工智能日趋成熟，智能运维时代已经到来。

背景

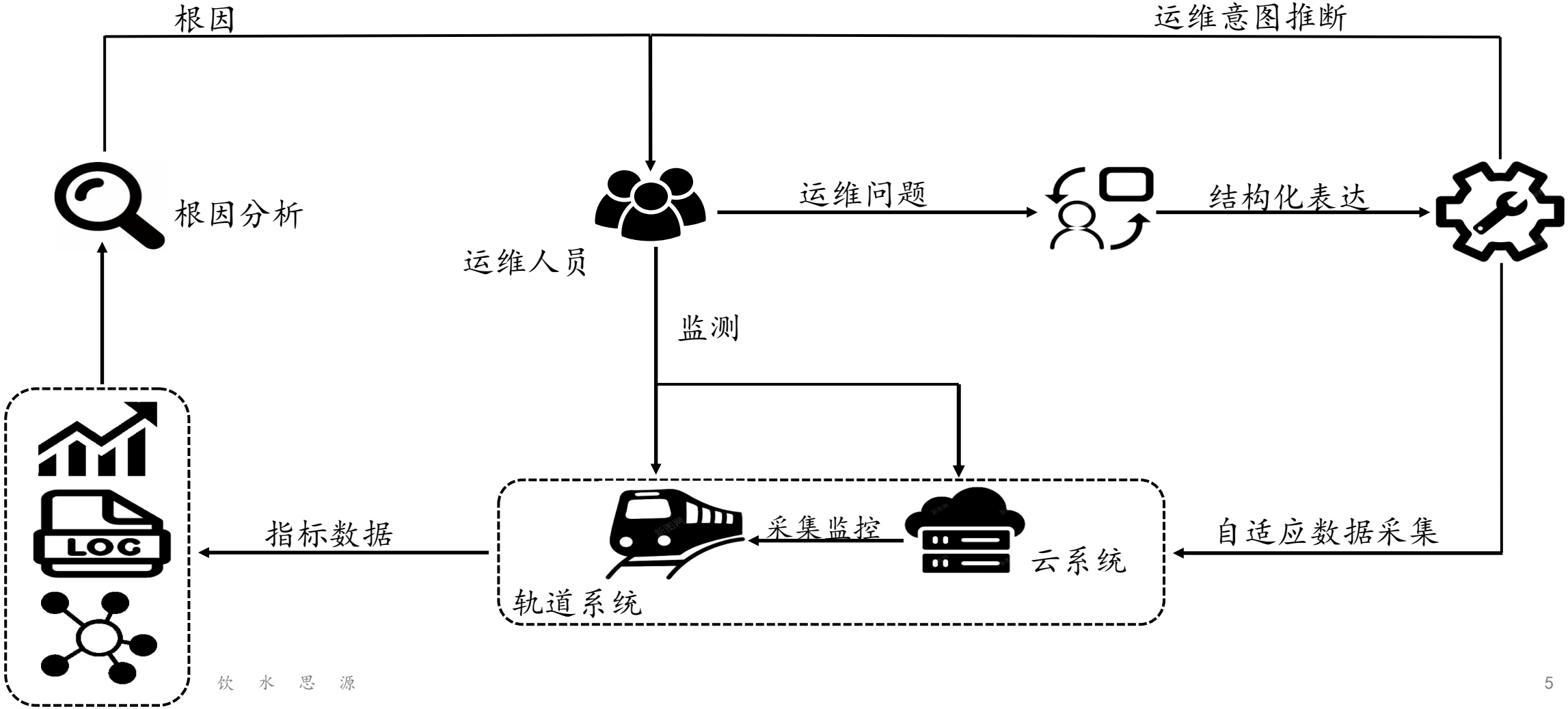
- 复杂网络系统由成千上万的设备构成，它们之间有复杂的调用关系
- 一个故障可能会导致多个机器运行不正常
- 一个故障伴随数个告警信号，运维人员需要从众多的报警信号找到故障发生的真正原因

近年兴起的智能运维致力于解决上述问题：

1. 实时检测机器指标（日志，时间序列异常检测）
2. 快速定位故障（知识推理、根原因分析）



知识驱动的铁路智能运维



大纲

- 背景
- 监测
  - 日志
  - 指标
- 诊断
  - 异常检测
- 推理
  - 根因分析
  - 知识图谱

- 铁路系统中部署的大规模设备在运行过程中产生海量日志，随着系统规模的变大、复杂度的提高，日志数据量越来越大，运维人员很难从海量的日志数据中发现有用的信息。
- 例如，某地铁线路的自动列车监控系统每天产生的日志平均约在 25 万条，一年的数据量高达9000万条。
- 智能日志分析通过AI算法，自动、实时、准确的从日志数据中发现异常，为后续故障的定位、诊断、自愈提供基础。

- 日志是一种半结构化文本格式的时序数据，它在系统运行过程中记录着设备状态和关键事件。
- 如图所示，日志文件的每一行表示不同的事件，并且可能包含不同类型的信息，例如日志类型、时间戳、进程ID、线程ID和日志消息等。
- 通过日志分析，运维人员可以发现或预知网络中已发生或潜在的故障。

日志示例:
Log1：2009-11-08 20:36:15 Packet Responder 1 for block blk123 terminating
Log2：2009-11-08 20:38:07 Packet Responder 0 for block blk456 terminating
Log3：2009-11-08 20:46:55 Received block blk_789 of size 67108864 from /10.251.42.84

- 经典方法
  - 根据日志级别（如 Info、Warning、Error）进行报警
  - 设置规则，匹配日志中特定字符串进行报警
- 不足
  - 需要手工设定，依赖人工经验，实现起来较难
  - 报警依旧很多
  - 只能检测已知和确定模式的异常

智能报警

- 机器学习（Machine learning）方法
  - 自动化日志分析
    - 主成分分析（PCA）
    - 不变量挖掘
    - 聚类
- 不足
  - 解释性不足
  - 适应性差
  - 手工特征工程成本高昂

- 深度学习方法
  - 强大的复杂关系建模能力
- 两种
  - DeepLog 为代表的日志模式序列异常检测
  - LogRobust 为代表的、针对日志语义进行异常检测的分析方式

- **挑战：日志格式不统一**
  - 不同类型设备的日志格式不同，如：时间格式、日志级别不统一、不同厂家自定义的专业词汇或缩略语不统一
  - 这些问题增加了日志分析的难度
- **机器学习方法**
  - 智能模板识别
  - 实体提取
  - 关系提取
  - 语义解析

应用

- 在路网系统的运行过程中，会产生海量日志数据
  - 维护人员的操作信息
  - 机车运行各种指标
  - 机车状态
- 这些日志信息对运维人员进行故障检测以及根因定位尤为重要

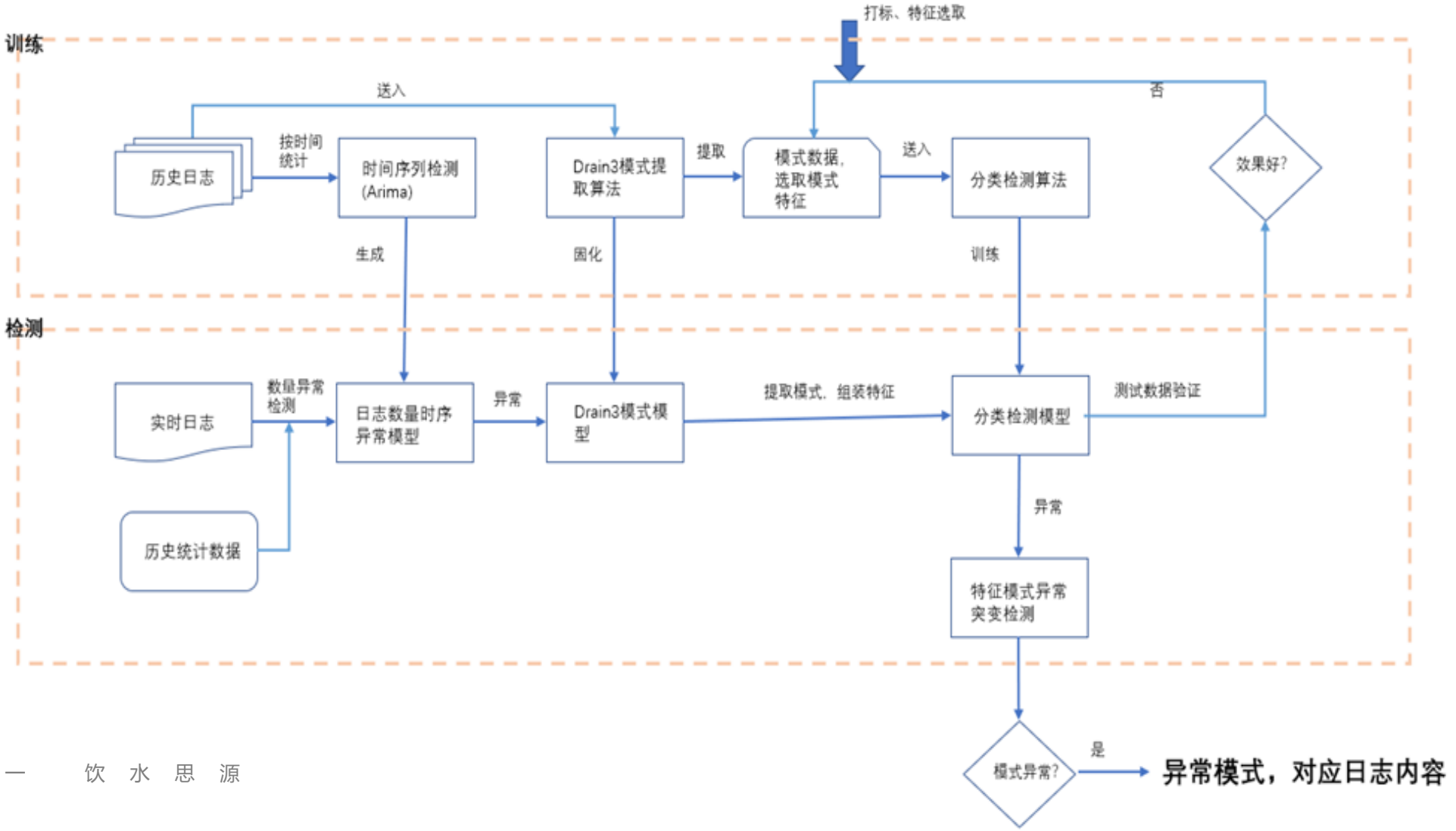
应用场景

- 对日志数据进行采集、清洗以及分析。
- 通过对日志数据的分析，回溯机车告警信息
- 对维护人员的操作能成功排除故障的案例，获取其成功维护经验，自动生成专家知识库，用于指导同类故障处理；
- 预先制定告警项、触发器、故障动作等，做到实时故障预警、故障恢复，实现机车运营的智能化、自动化；
- 实现日志的可视化多维分析。

智能故障处理和恢复流程

- 日志模式的自动提取
  - Drain 是目前最常用的日志提取算法，可以以流方式实时解析日志。它使用固定深度解析树，经过特殊设计的解析规则进行日志编码，归类
- 故障识别模型的训练
  - 通过自然语言处理和时间序列异常检测方法，使用历史日志信息对模型进行训练，对日志内容进行建模
- 在线故障识别和处理
  - 基于训练好的模型，在预测阶段，结合日志多维度信息发现问题
  - 查找知识库，搜索解决方案，自动进行故障恢复

自动日志异常检测流程


训练
检测
打标、特征选取
送入
否
历史日志
按时间统计
时间序列检测(Arima)
Drain3模式提取算法
提取
模式数据，选取模式特征
送入
分类检测算法
效果好?
生成
固化
训练
实时日志
数量异常检测
日志数量时序异常模型
异常
Drain3模式模型
提取模式，组装特征
分类检测模型
测试数据验证
历史统计数据
异常
特征模式异常突变检测
模式异常?
是
异常模式，对应日志内容

大纲

- 背景
- 监测
  - 日志
  - 指标
- 诊断
  - 异常检测
- 推理
  - 根因分析
  - 知识图谱

作用：通过相关的算法自动地发现运维数据中的异常波动或异常行为，为后续的异常告警、根因分析等任务提供相应的决策依据

突增
突降
抖动
……

→

服务调用延迟
服务器故障
配置错误
缺陷版本上线
外部攻击
……

作为AIOps的关键场景和技术之一，异常检测对AIOps的推进和落地同样具有重要意义。

传统方法：

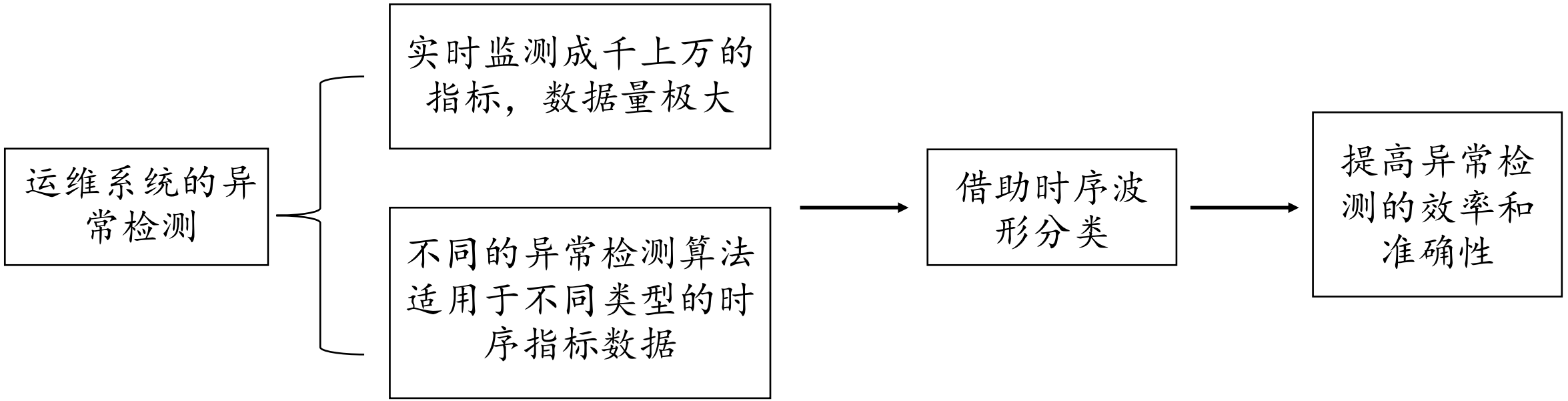
1.基于统计的方法

2.基于规则的方法

智能运维场景

运维平台需要对上万的指标进行监控，时间序列在维度和长度这两个方面急剧增长，传统方法无法适应数据大规模增长的需求，异常检测效率低下。

波形分类


运维系统的异常检测
实时监测成千上万的指标，数据量极大
不同的异常检测算法适用于不同类型的时序指标数据
借助时序波形分类
提高异常检测的效率和准确性

机器学习方法

- 基于统计的方法
- 基于相似度量的方法
- 基于深度学习的方法

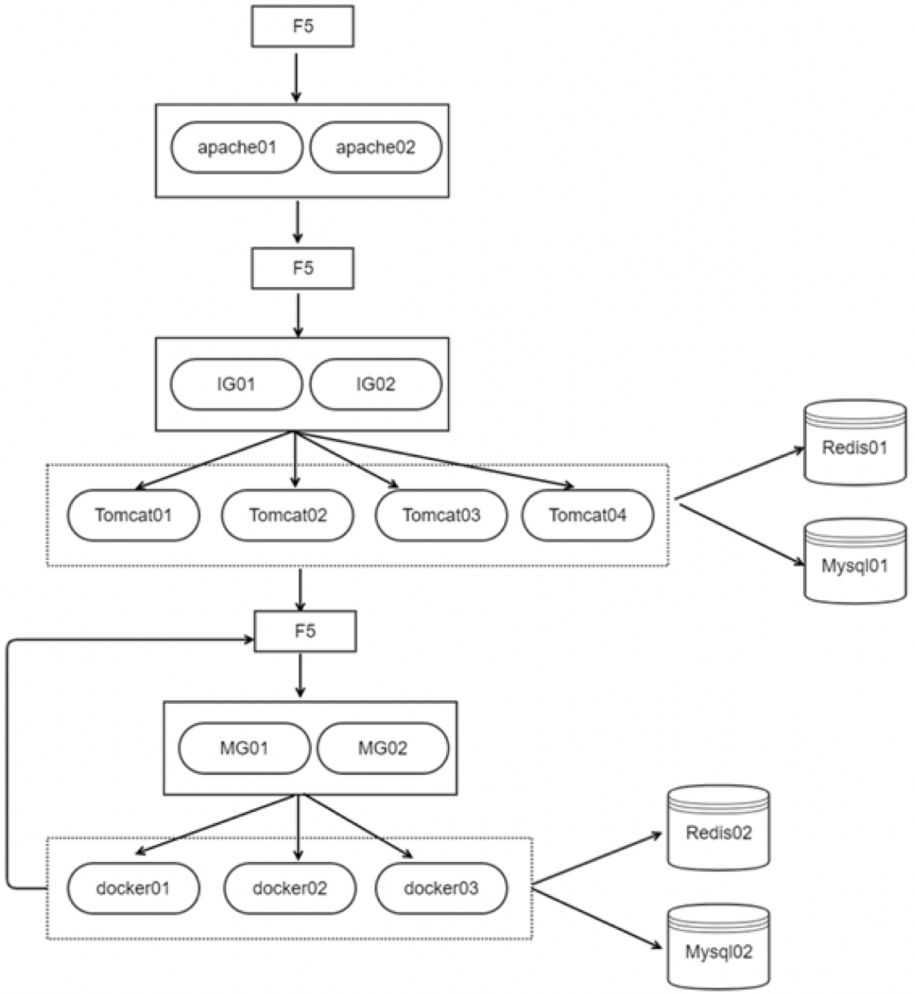
基于距离的方法:LOF(Local Outlier Factor)

基于聚类的方法：K-means

基于集成学习的方法：孤立森林

某金融机构业务系统

- 简化的业务拓扑图如右图所示
- 实际上机器有上百台
- 每台机器有上百个指标，例如：
  system.cpu.pct_usage（CPU使用率）
  system.io.w_await（磁盘响应时间）
- 每个指标每秒或者每隔几秒产生一个数据点
- 每分钟监控上万个指标



大纲

- 背景
- 监测
  - 日志
  - 指标
- 诊断
  - 异常检测
- 推理
  - 根因分析
  - 知识图谱

根因分析

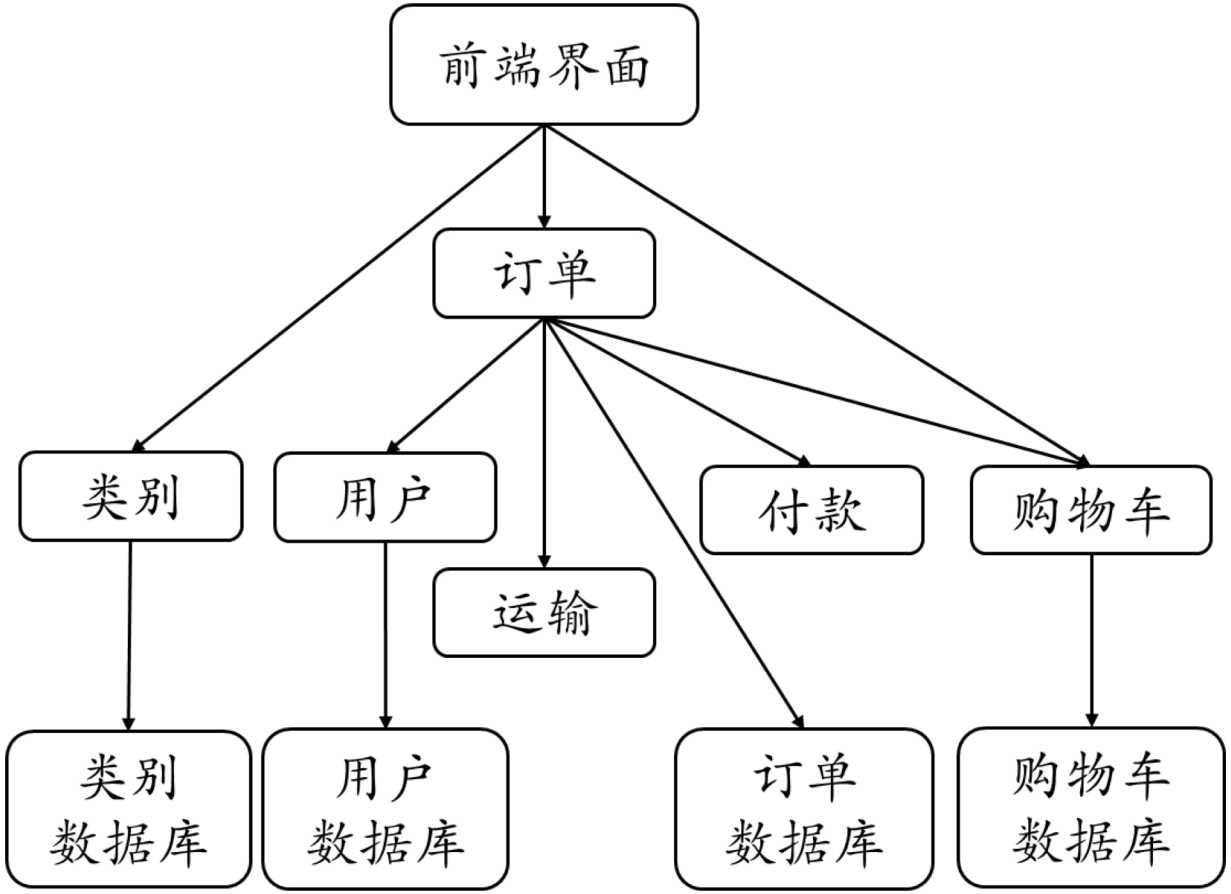


图1 复杂网络系统的关联网络

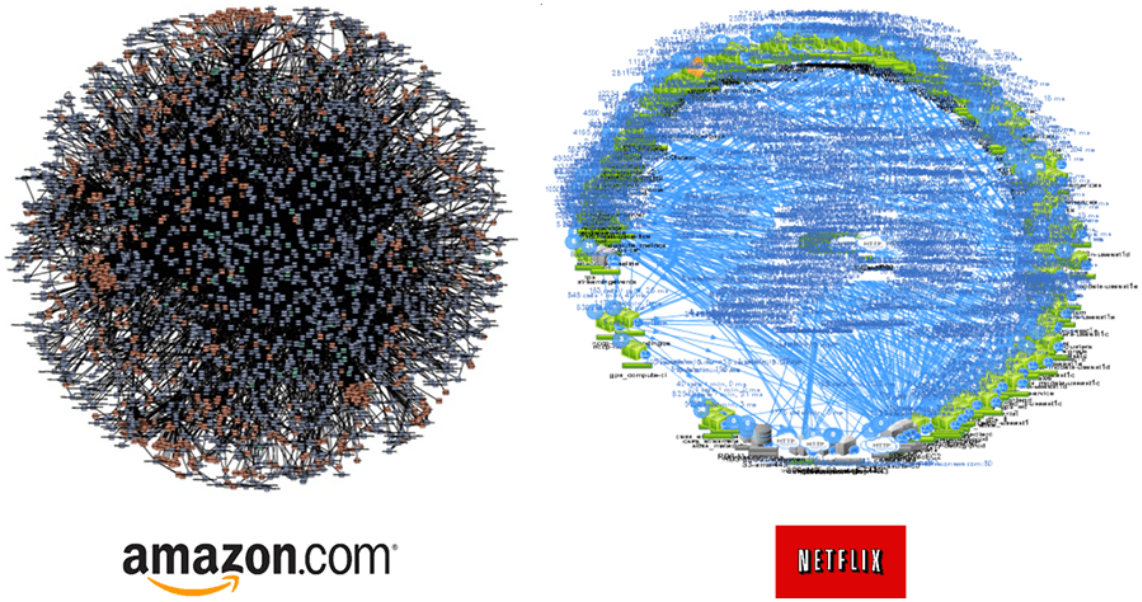
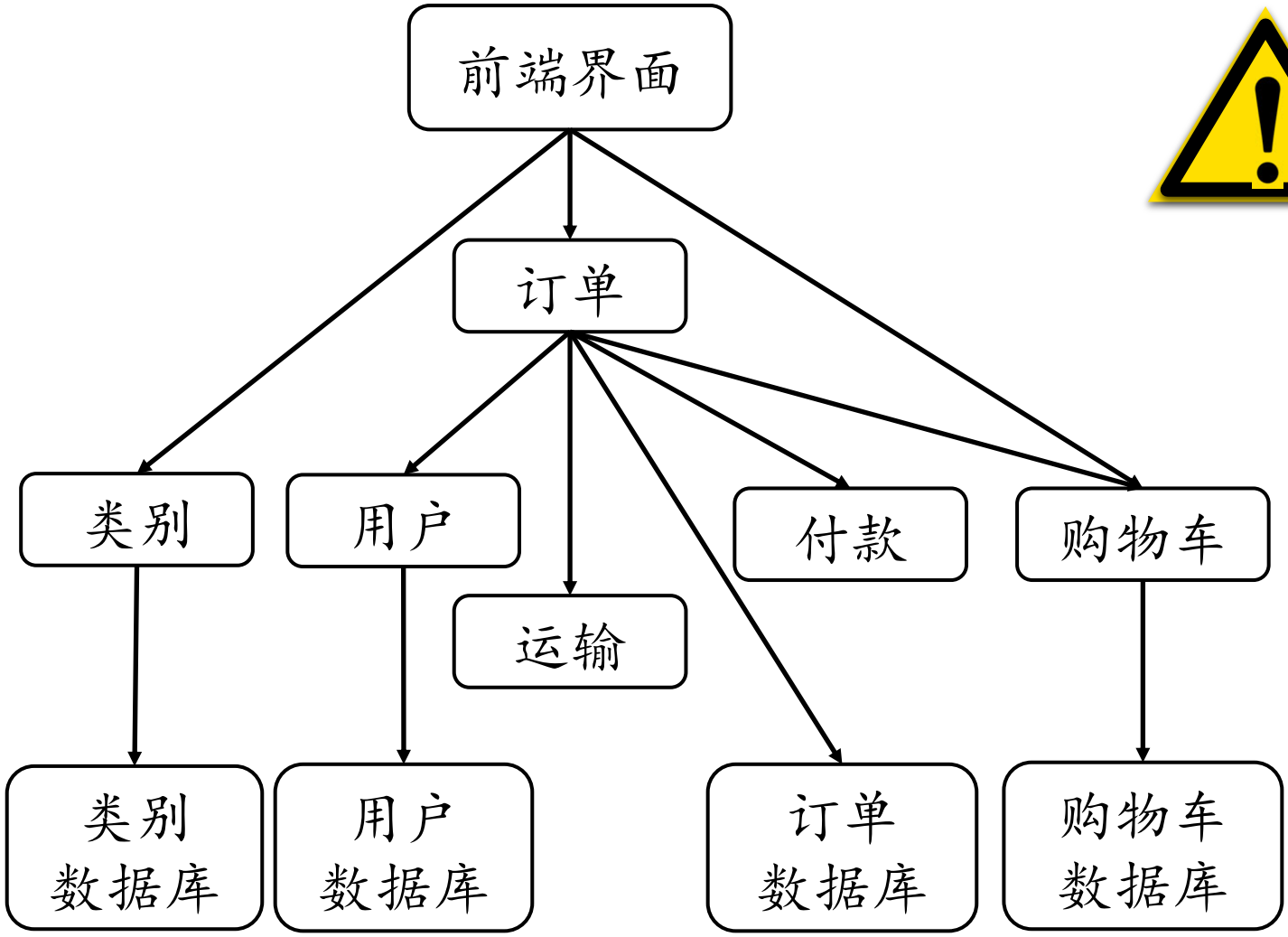


图2 微服务系统示例

故障传播

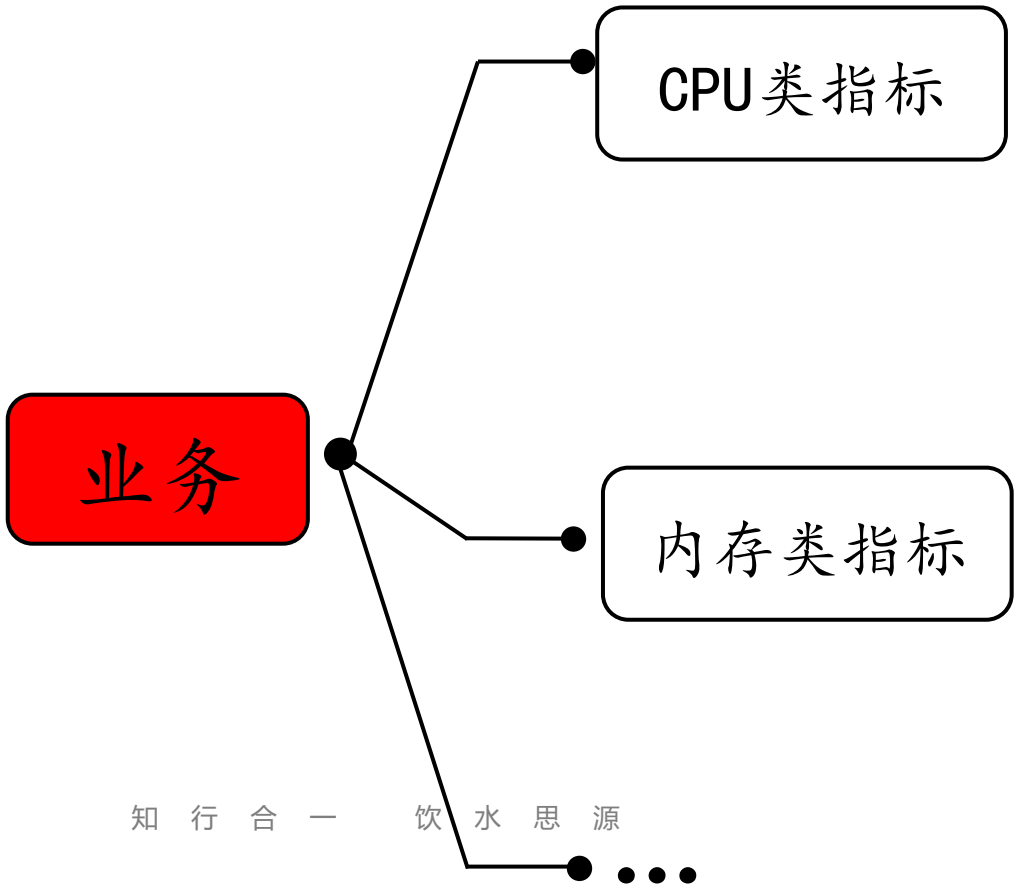


代码漏洞
配置错误
外部攻击

1. 单个微服务的故障会传播至多个相关的微服务

挑战

2. 一个异常的微服务会有大量的（上百个）指标异常



Processor_load_1_min[系统CPU总负载/1min]
Processor_load_5_min[系统CPU总负载/5min]
CPU_util_pct[CPU使用率]
CPU_user_time[CPU时间用户百分比]
CPU_system_time[CPU时间系统百分比]
…

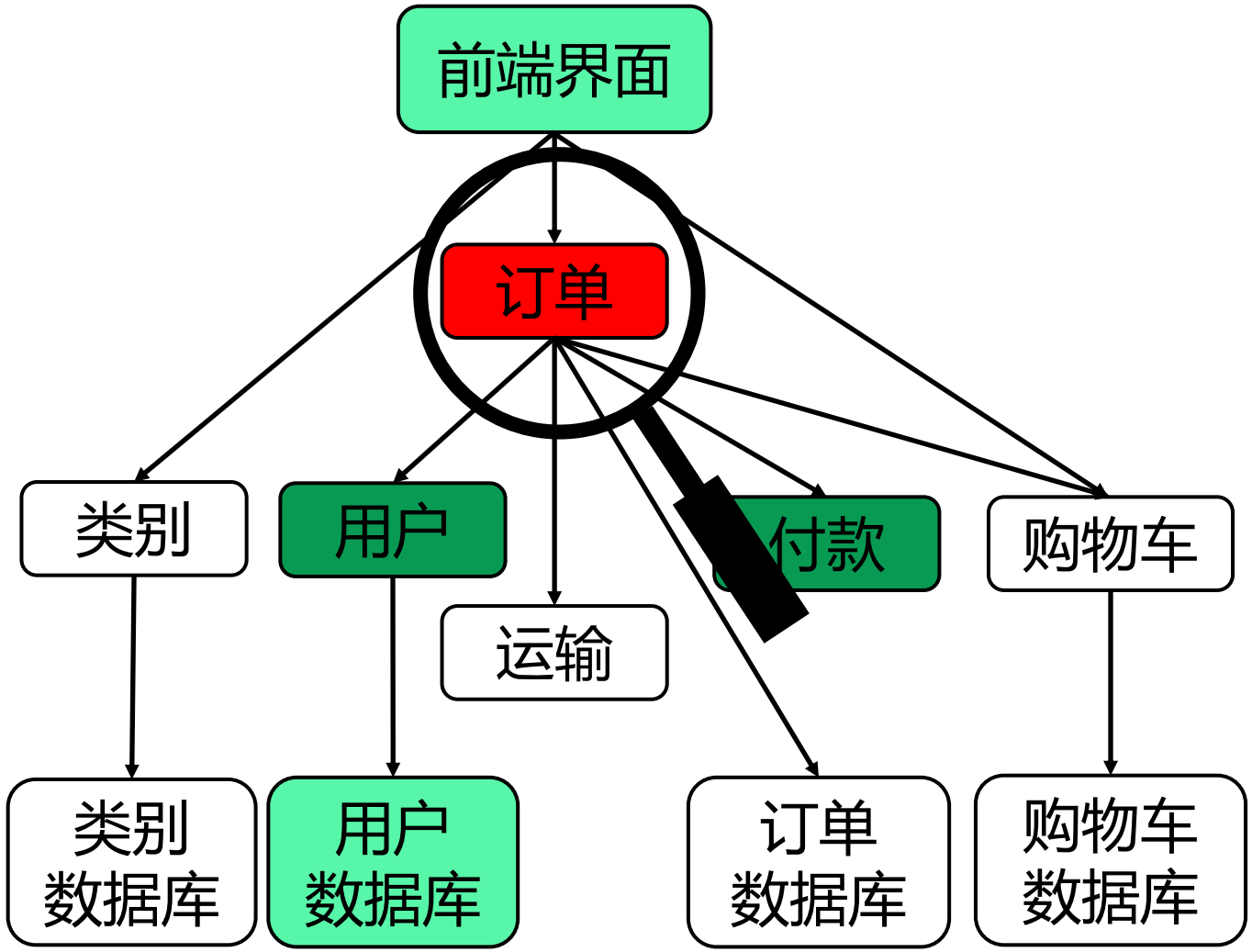
Memory_total[物理内存总量]
Memory_used[物理内存使用量]
Shared_memory[内存共享区域大小]
Memory_available_pct[物理内存可用率]
…

问题

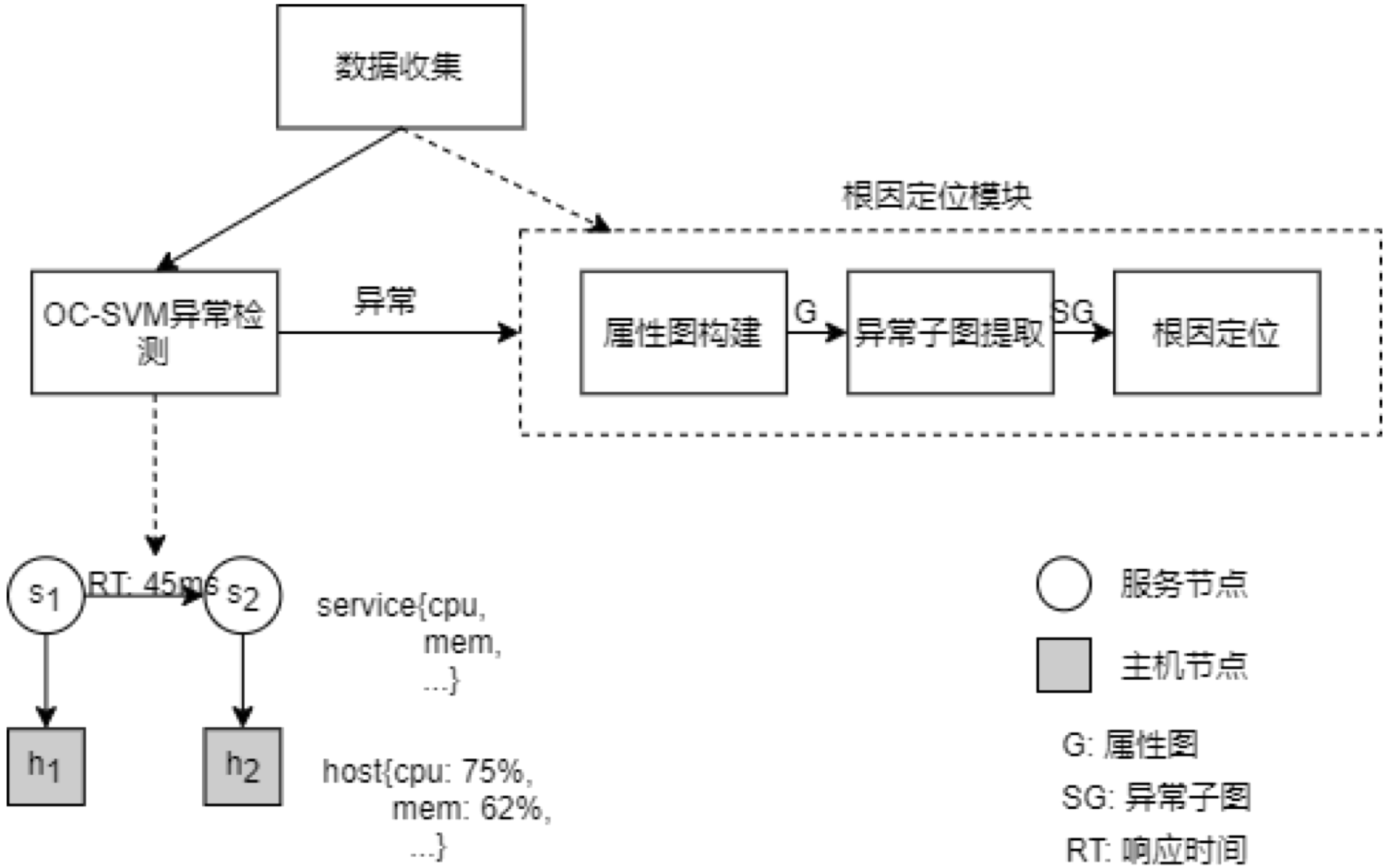
1. 从多个异常的微服务中定位出根因微服务

2. 从根因微服务的众多异常指标中定位出根因指标

Processor_load_1_min[系统CPU总负载/1min]
Processor_load_5_min[系统CPU总负载/5min]
CPU_util_pct[CPU使用率]
Memory_total[物理内存总量]
Memory_used[物理内存使用量]
…

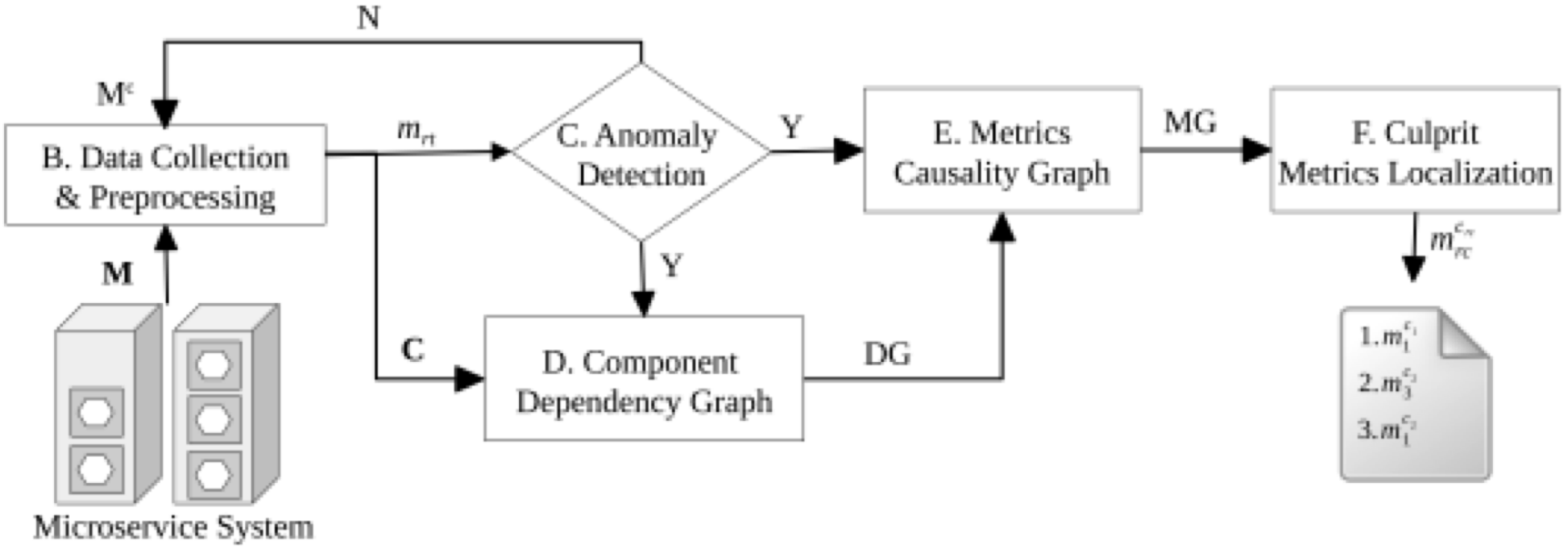


方案



以检测服务间的异常响应时间为入口，检测到异常后，将相关的微服务作为节点，一定时间窗口的调用关系作为连边，构成一张属性图，根据相关性赋予边的权重，最后通过随机游走的方式找到根因服务。

方案



定位到根因微服务以后，将经过筛选的指标作为节点，采用因果推断方式为它们建立连边，在指标图上采用PageRank算法，找到根因指标。

MicroDiag (ICSE 2021)

挑战及难点

大量的异常指标

- 指标数量众多，异常表现差异大，假警难以识别
- 指标间相关性动态变化，难以刻画

多种故障类型

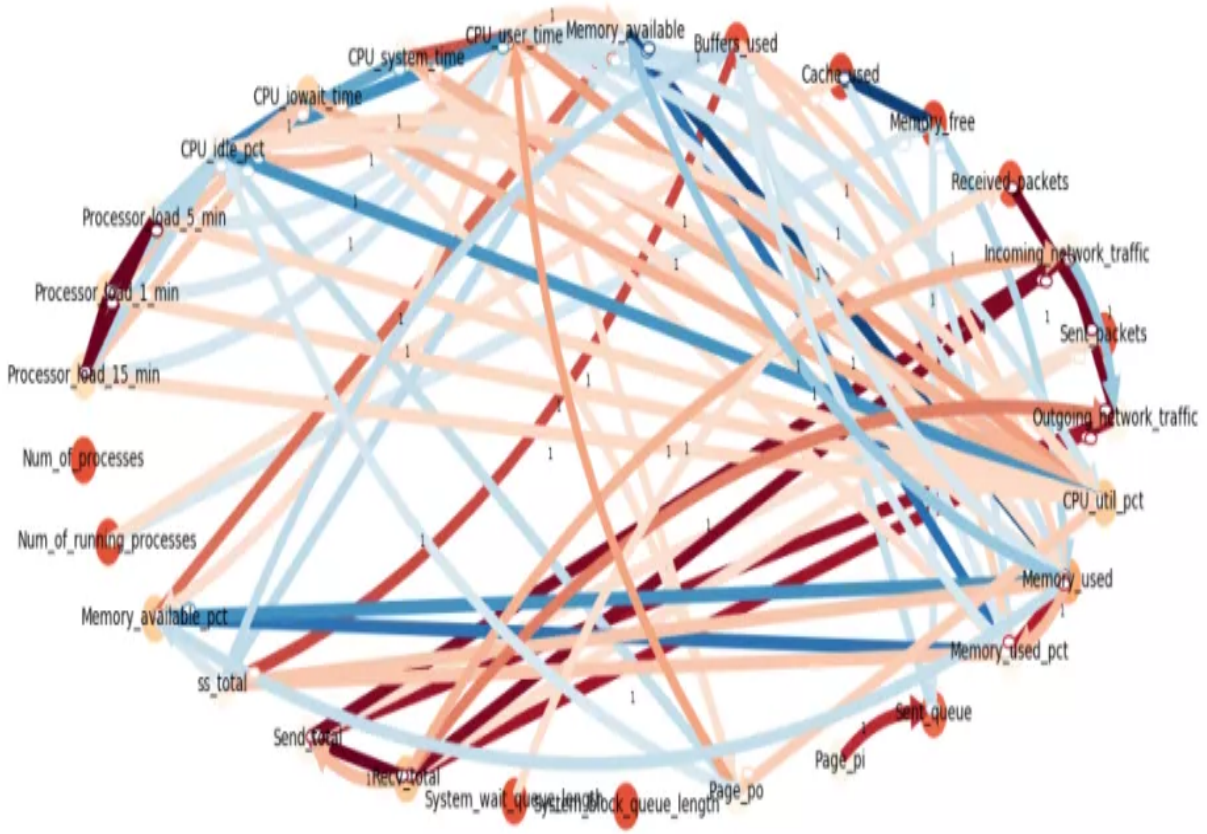
- 故障类型表现形式不同
- 难以刻画不同故障类型的模式

方案

大量的异常指标 ⇨ 结合指标相关性 + 结合调用链数据 ⇨ 过滤假警、提升根因定位的准确性。

多种故障类型 ⇨ 系统注入不同类型的故障 + 刻画不同的故障传播模式 ⇨ 根据故障传播模式，识别故障类型，定位根因。

根因分析


CPU_user_time
Memory_available
Buffers_used
CPU_system_time
Cache_used
CPU_iowait_time
Memory_free
CPU_idle_pct
Received_packets
Processor_load_5_min
Incoming_network_traffic
Processor_load_1_min
Sent_packets
Processor_load_15_min
Outgoing_network_traffic
Num_of_processes
CPU_util_pct
Num_of_running_processes
Memory_used
Memory_available_pct
Memory_used_pct
ss_total
Sent_queue
Send_total
Page_pi
Recv_total
Page_po
System_wait_queue_length
System_block_queue_length

大纲

- 背景
- 监测
  - 日志
  - 指标
- 诊断
  - 异常检测
- 推理
  - 根因分析
  - 知识图谱

- 知识驱动的智能运维

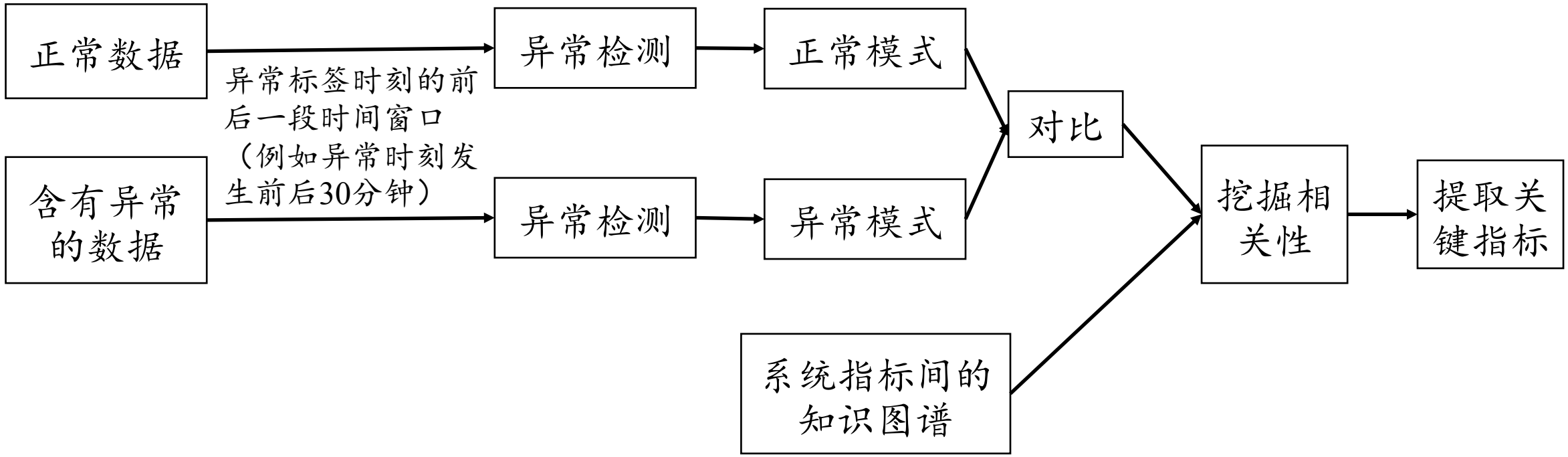
  - ✓ 利用知识图谱引入专家知识
  - ✓ 解决故障诊断和检测的流畅性与可靠性
  - ✓ 提高大规模系统的诊断和故障恢复效率
  - ✓ 推动铁路知识推理、自动化技术的发展

研究现状

作者，年份	知识形态	类型	模型
Ghazvininejad，2018	文本	基于知识生成	Memory Networks
Zhu，2017	知识库	基于知识生成	Seq2Seq, Knowledge Retriver
Zhou，2018	知识图谱	基于知识生成	Seq2Seq, Graph Attention
Moon，2019	知识图谱	知识选择	Bi-LSTM, KG path decoder
Jung，2020	知识图谱	知识选择	ALBERT, Attention flow GNN，
Galetzka ，2020	知识图谱	基于知识生成	Graph Attention，GPT

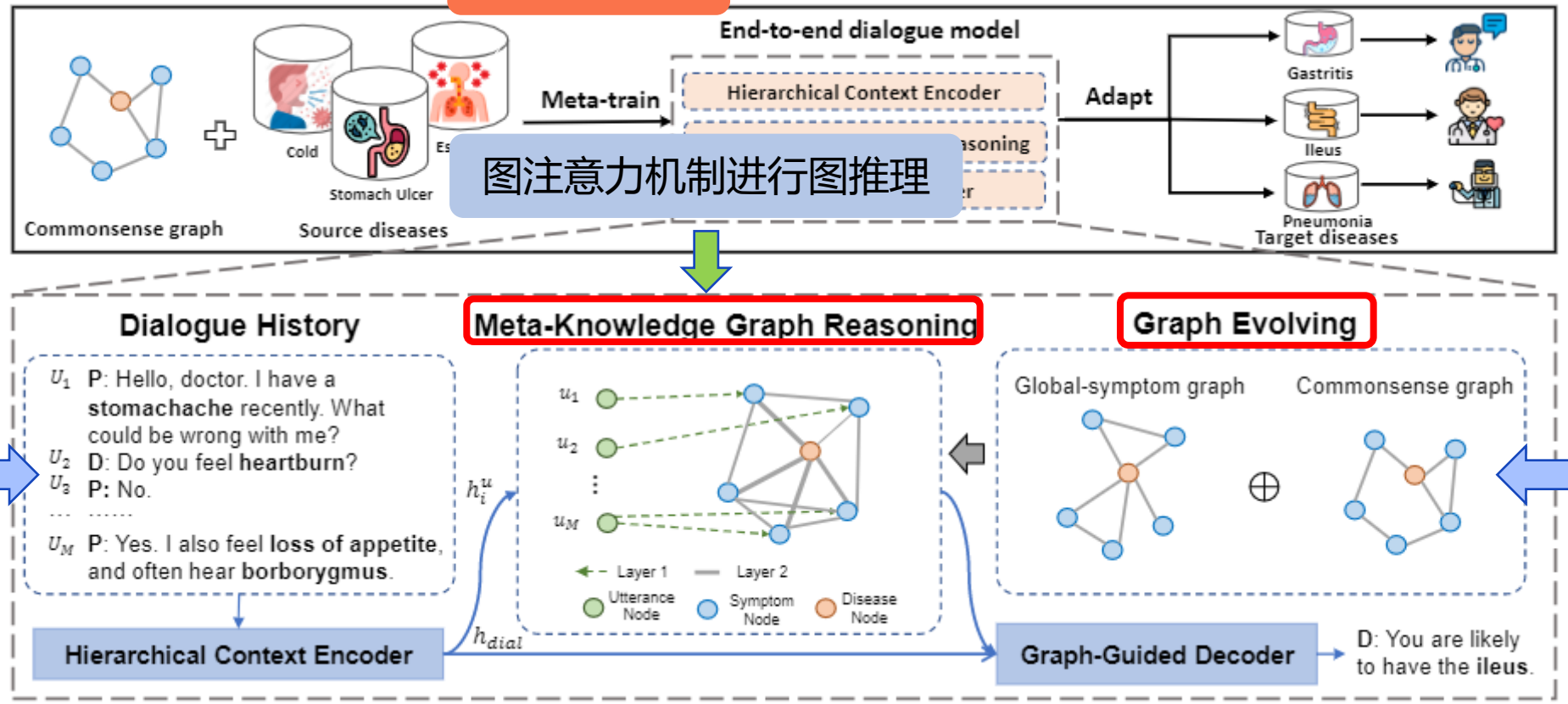
在众多要监测的指标中发现关键指标

方案：



知识驱动的智能诊断

MAML元学习



LSTM分层编码

图进化

[3] Graph-Evolving Meta-Learning for Low-Resource Medical Dialogue Generation

关键研究内容和挑战

➢ 图谱构建

- 实体具有随机性和稀疏性
- 引入知识图谱的过程中易出现知识噪声
- 知识图谱的简化与动态更新

➢ 诊断推理

- 有效结合预测出的实体和历史故障信息
- 故障出现的随机性
- 少量故障历史情况下的冷启动问题

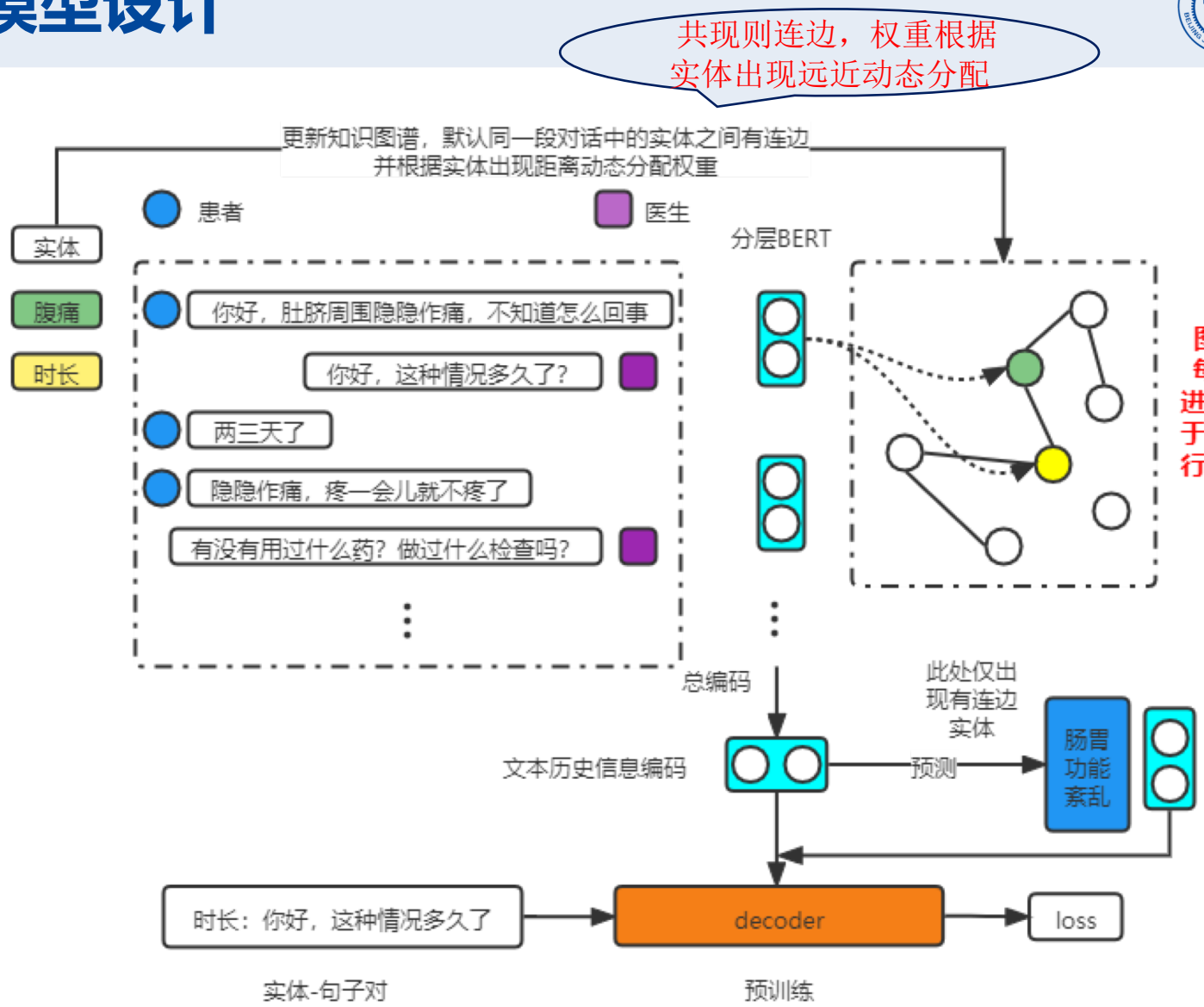
➢ 图谱构建

- 引入知识图谱，通过检索知识图谱获取实体之间的相关性
- 简化知识图谱，减小知识图谱引入带来的噪声
- 智能图动态更新策略，一方面缓解噪声问题，另一方面使知识图谱可以动态更新，适应新故障

➢ 诊断推理

- 实体-句子相互重构的预训练方法，建立实体与句子间的强关联性
- 实体与历史日志融合，增加实体权重
- 采用元学习方式，缓解案例缺少的问题

模型设计



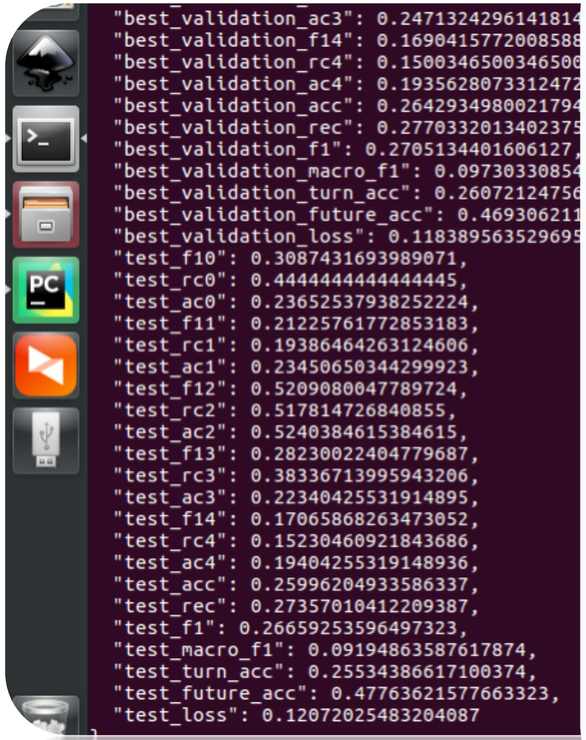
图注意力机制。
每个epoch之后
进行剪枝操作，对
于权重较小的边进
行删除，对无连边
节点进行删除

动态更新
的常识图

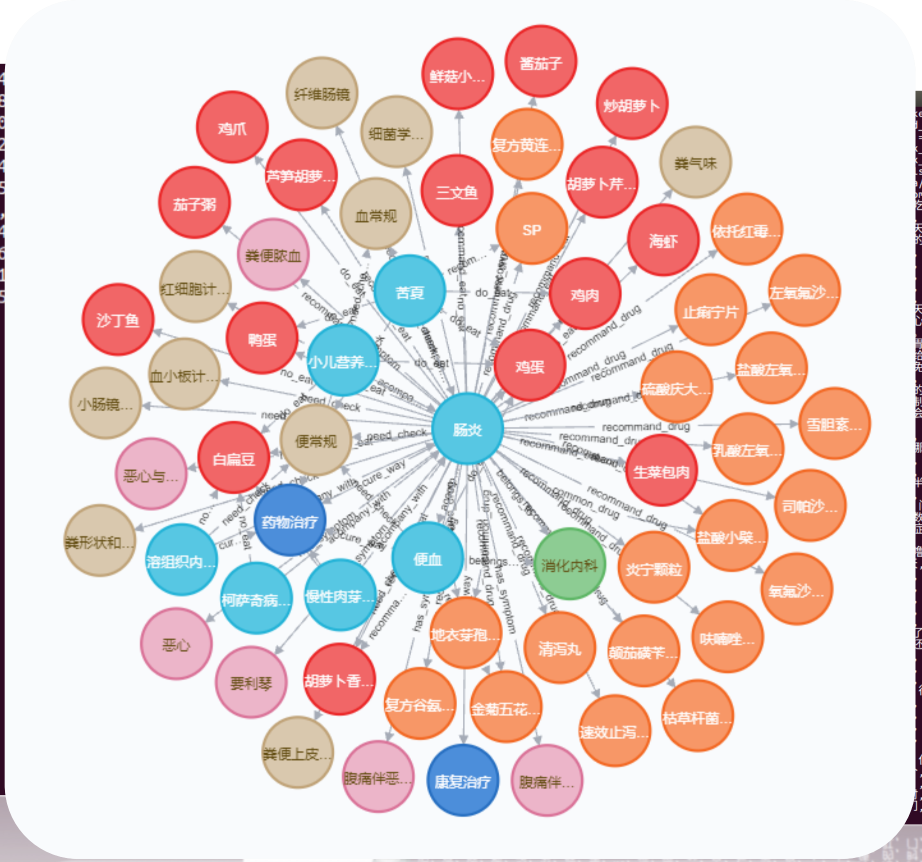
对decoder预训练建立
实体-句子的强关联

核心算法

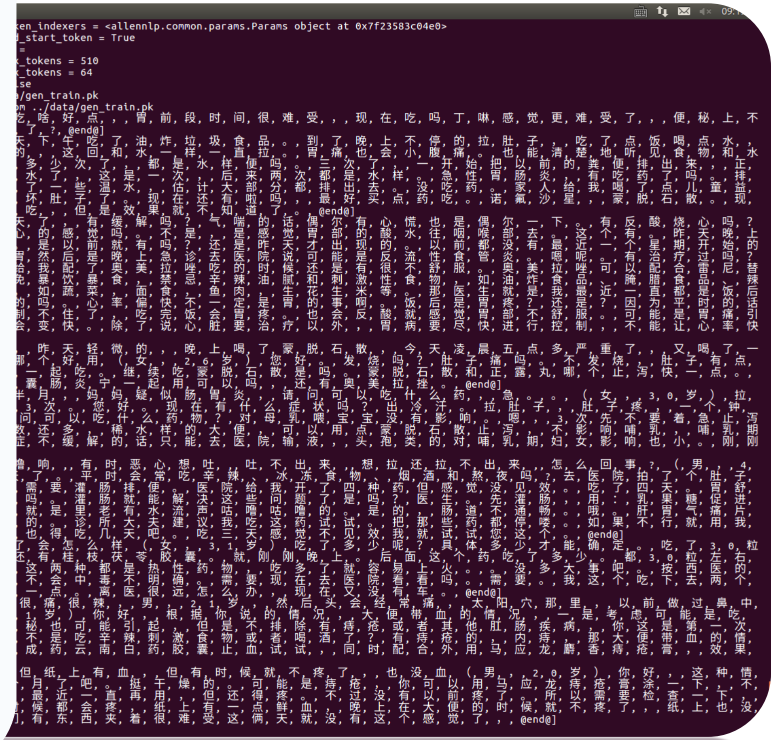
- 知识图谱和文本表征的融合
- 知识图谱动态更新
- 智能诊断和对话
- 实体预测与智能诊断



"best_validation_ac3": 0.2471324296141814
"best_validation_f14": 0.1690415772008588
"best_validation_rc4": 0.1500346500346500
"best_validation_ac4": 0.1935628073312472
"best_validation_acc": 0.2642934980021794
"best_validation_rec": 0.2770332013402375
"best_validation_f1": 0.2705134401606127,
"best_validation_macro_f1": 0.09730330854
"best_validation_turn_acc": 0.26072124756
"best_validation_future_acc": 0.469306211
"best_validation_loss": 0.118389563529695
"test_f10": 0.3087431693989071,
"test_rc0": 0.4444444444444445,
"test_ac0": 0.23652537938252224,
"test_f11": 0.21225761772853183,
"test_rc1": 0.19386464263124606,
"test_ac1": 0.23450650344299923,
"test_f12": 0.5209080047789724,
"test_rc2": 0.517814726840855,
"test_ac2": 0.5240384615384615,
"test_f13": 0.28230022404779687,
"test_rc3": 0.38336713995943206,
"test_ac3": 0.22340425531914895,
"test_f14": 0.17065868263473052,
"test_rc4": 0.15230460921843686,
"test_ac4": 0.19404255319148936,
"test_acc": 0.25996204933586337,
"test_rec": 0.27357010412209387,
"test_f1": 0.26659253596497323,
"test_macro_f1": 0.09194863587617874,
"test_turn_acc": 0.25534386617100374,
"test_future_acc": 0.47763621577663323,
"test_loss": 0.12072025483204087



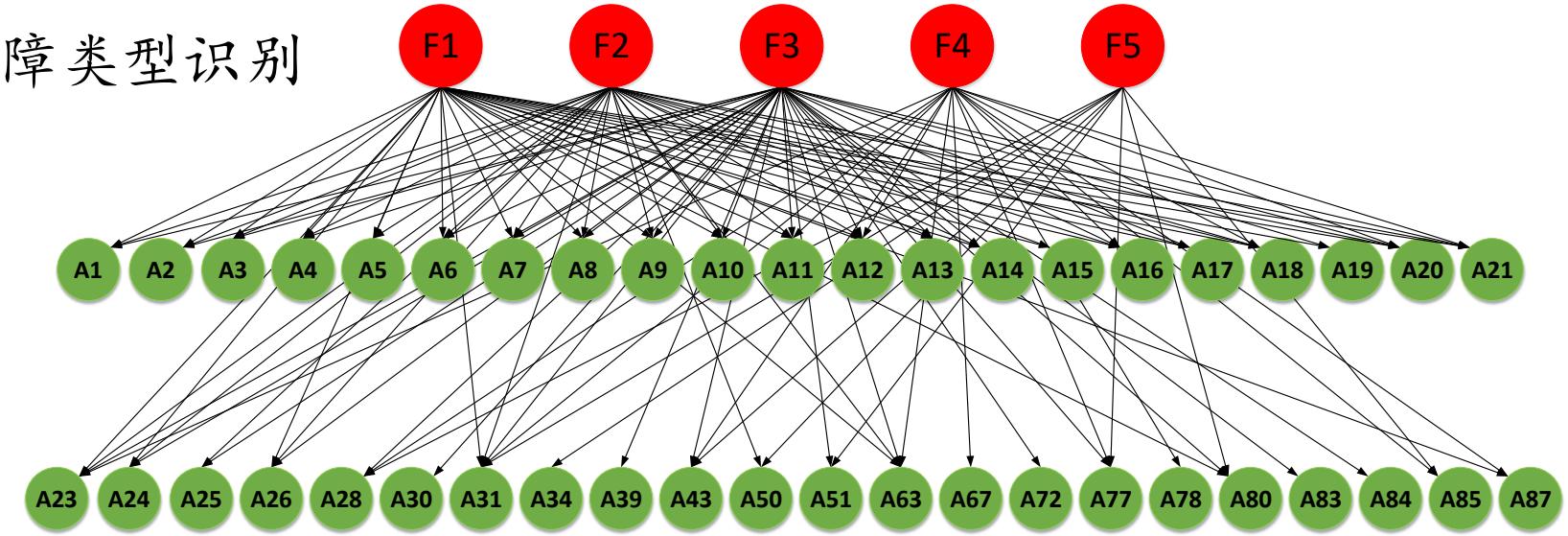
肠炎
苦夏
小儿营养...
便常规
药物治疗
便血
消化内科
鸡蛋
鸡肉
SP
三文鱼
血常规
鸭蛋
白扁豆
沙丁鱼
康复治疗



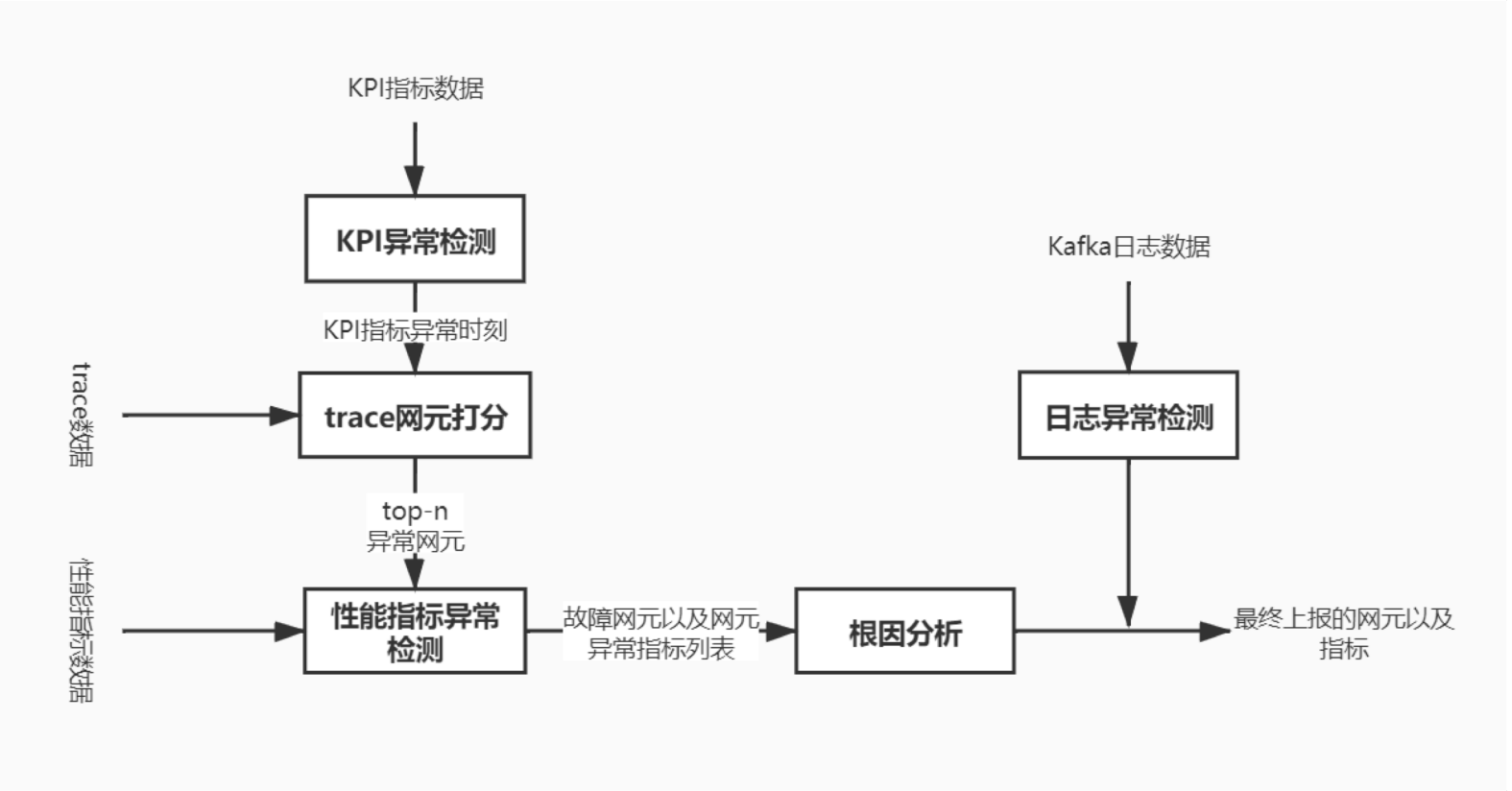
应用

- 中国移动：基于告警日志的大规模基站网络设备故障识别与预测
- 宝兰德： KPI、性能指标、Tracing、日志联合性能监控、故障预测、异常检测和根因定位
- 中国联通：基站无线网络流量时空模型和预测
- 久其软件：贵州高速交通流量模型与预测
- 视频监控、传输优化、内容分析与识别

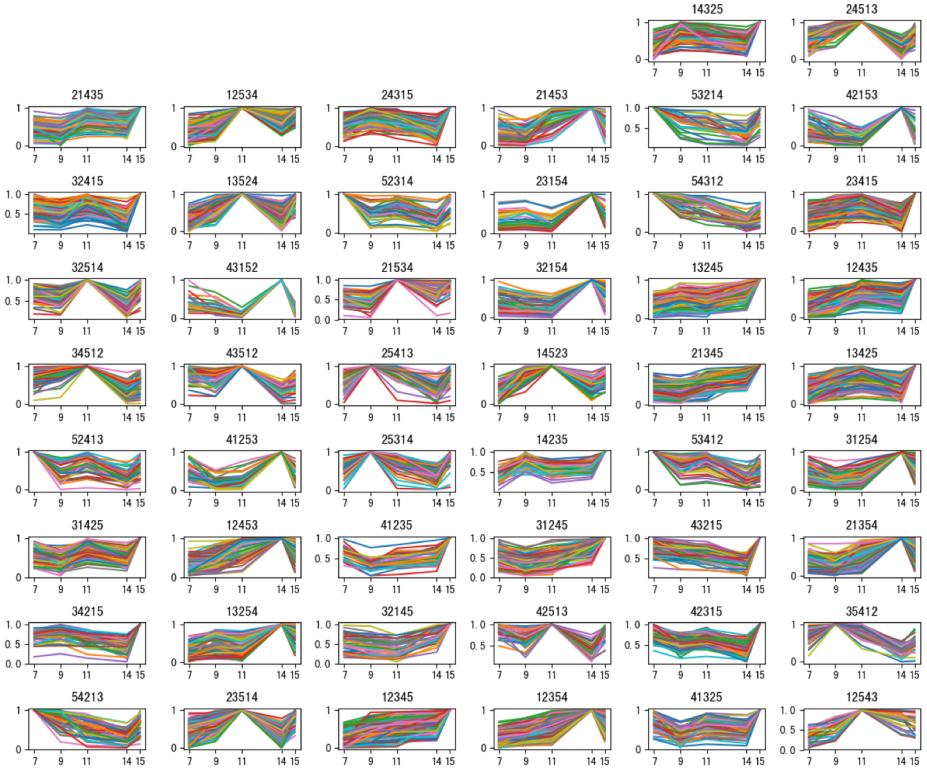
- 基于告警日志的大规模基站网络设备故障识别与预测
  - 自然语言处理技术
  - 移动通信基站故障类型的识别
  - 基于贝叶斯网络的故障类型识别
  - 告警关联关系挖掘
  - 故障关联关系挖掘
  - 中国移动应用



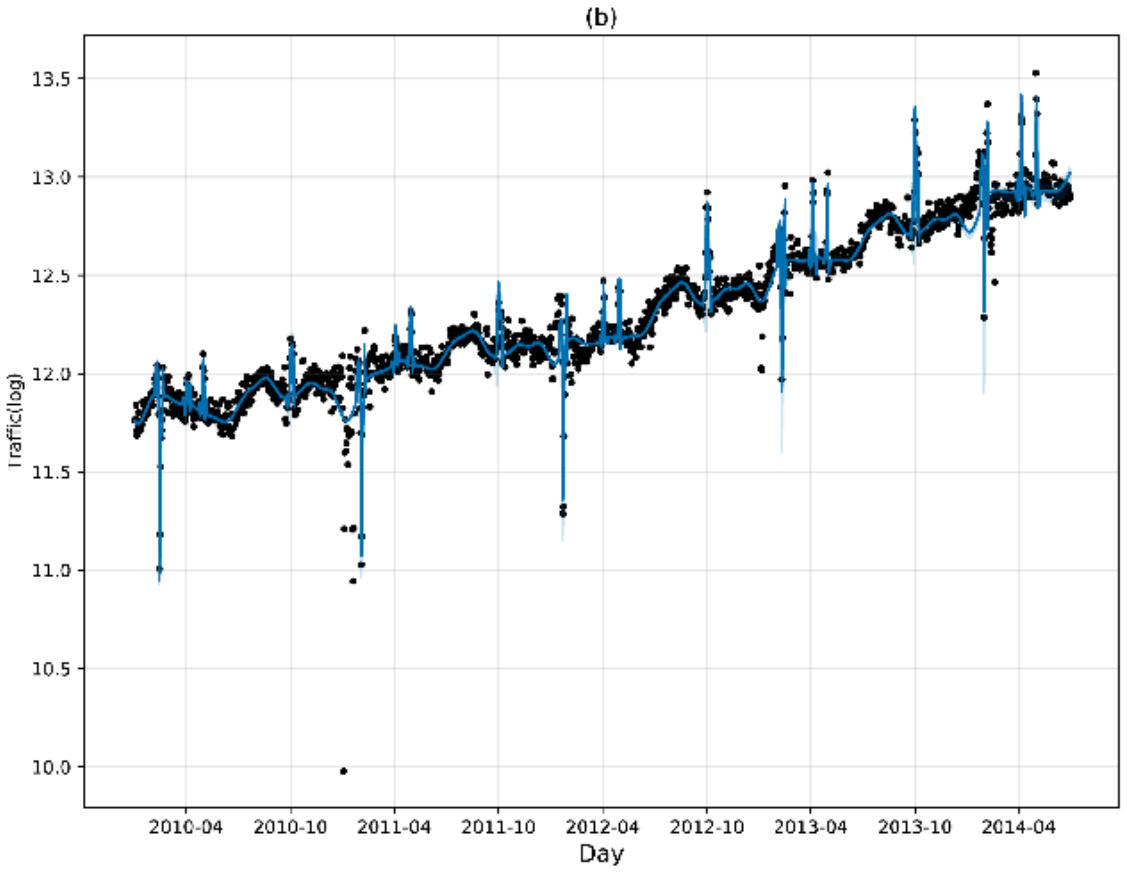
- KPI、性能指标、Tracing、日志联合性能监控、故障预测、异常检测和根因定位
  - 时间序列异常检测
  - 日志异常检测
  - 调用链异常检测
  - 异常节点识别
  - 根因定位
  - 企业应用



- 大规模无线网络流量时空模型与预测
  - 流量增长模型
  - 假日流量增长模型
  - 流量地理位置分布
  - 基于基站语义的流量模式预测
  - 中国联通应用



- 大规模交通网络流量时空模型与预测
  - 超长时间（5年）内
  - 细粒度（天）交通流量预测
  - 贵州省高速公路网应用



团队

- 教授：郭宇春、孙强
- 副教授：赵永祥、李纯喜、张立军、郑宏云、陈一帅
- 讲师：李磊、张梅

郭宇春　孙 强　郑宏云　张立军　赵永祥　李纯喜　李磊　陈一帅　张梅

感谢观看

联系方式

赵永祥
博士

地址 / 北京市海淀区上园村3号北京交通大学
电话 / 13621054870
邮箱 / yxzhao@bjtu.edu.cn