实时智能异常检测平台

——算法及工程实现

演讲人：陈剑明

1 运维的眼睛

一个监控系统

一个实时监控系统

一个大而全的实时监控系统

一个能发现业务异常的大而全的实时监控系统

接下来将面临什么？

2 问题

如何去设置告警？

- 忽略一部分
- 平稳指标 —— 单一阈值
- 周期指标 —— 量化幅度

怎样量化异常？

- 和过去同期比较
- 和预测值比较

规则维护成本高

没有统一的标准

3 报警系统是神经

既不要敏感

又不能大条

还得自适应

引入算法来解决这个问题

4 算法黑洞

明确的评估标准

- 召回率、准确率可衡量么？
- 异常本身有明确的定义么？

足够的样本数量

- 标注成本和标准？
- 每个指标都标注？

我们怎么面对？

5 控制变量，寻找最接近的情况

没有歧义的异常全集

- 锁定订单类监控指标（广泛认可/不被错过）
- 明确检验标准

算法选择

- 去规则化
- 经典统计分析方法的困扰 —— 误报和漏报

不接受稳妥，鱼和熊掌要兼得，可否？

6 呼唤新的算法

降低报警总量，到可以人工处理的程度

不以增加漏报为代价

不影响实时性

算法即服务，可重用

7 神经网络够神么

RNN适合处理序列变化的数据

LSTM在更长的序列上有良好的表现

适用于语音处理、输入法、时间序列

8 离线模型训练

2 weeks

X X X X X … … X X X X X

插补
剔除节假日
标注值替换

- 多尺度滑动窗口时序特征
- 频域特征

特征计算

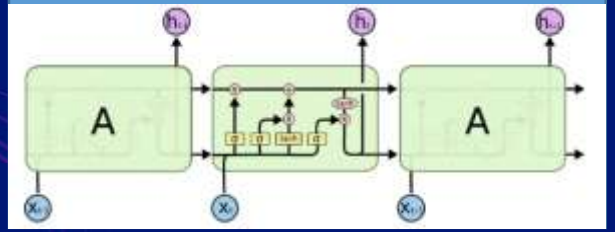
分类模型

平稳

周期

非周期

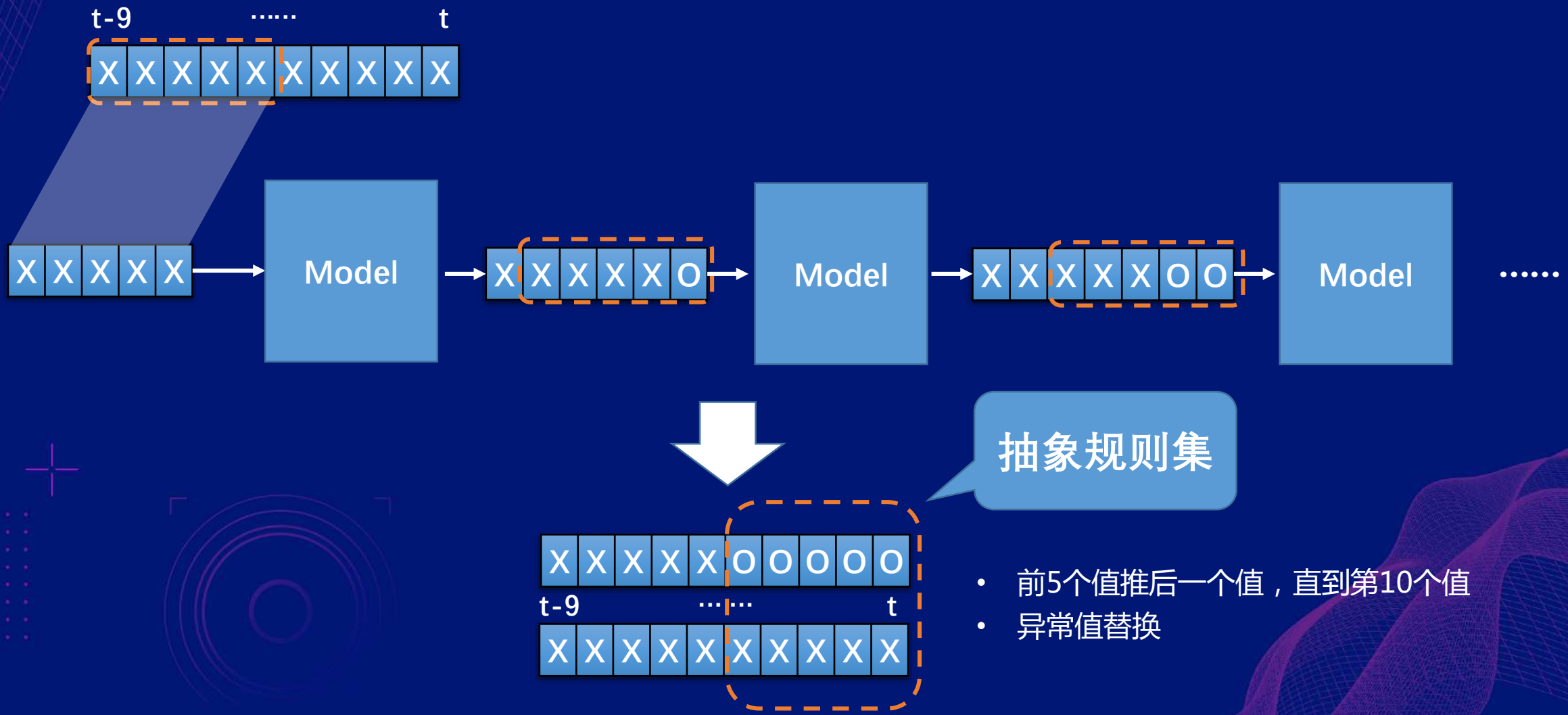
LSTM Model Training



Model

验证、调参

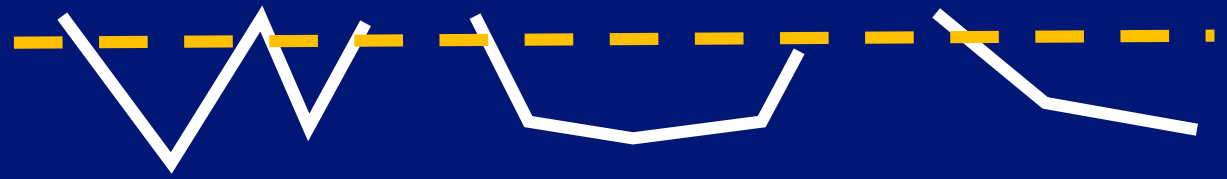
9 在线计算检测



- 前5个值推后一个值，直到第10个值
- 异常值替换

10 抽象规则集

- 1、2、3min
- 幅度、变化范围
- 数据点形态
- 相邻周期前后相似度



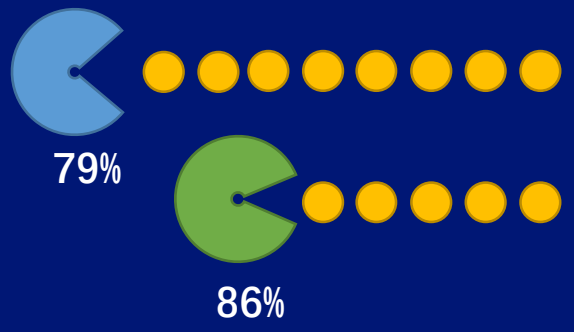
11 算法评估和检验

和当前规则系统相比的总告警量

以被认定的故障为全集，检出异常的数量

- 日均告警量：

- 故障召回率：



12 漏报的情况

肉眼无法识别的

虽有一定周期性，但绝对量小，波动大

波动剧烈，异常下跌没有明显表征

13 实时的痛点

Python在实时处理上的缺陷

数据处理的每一步都在等时间

拿到异常点到发出告警邮件需要3-4min

14 Why Flink ?

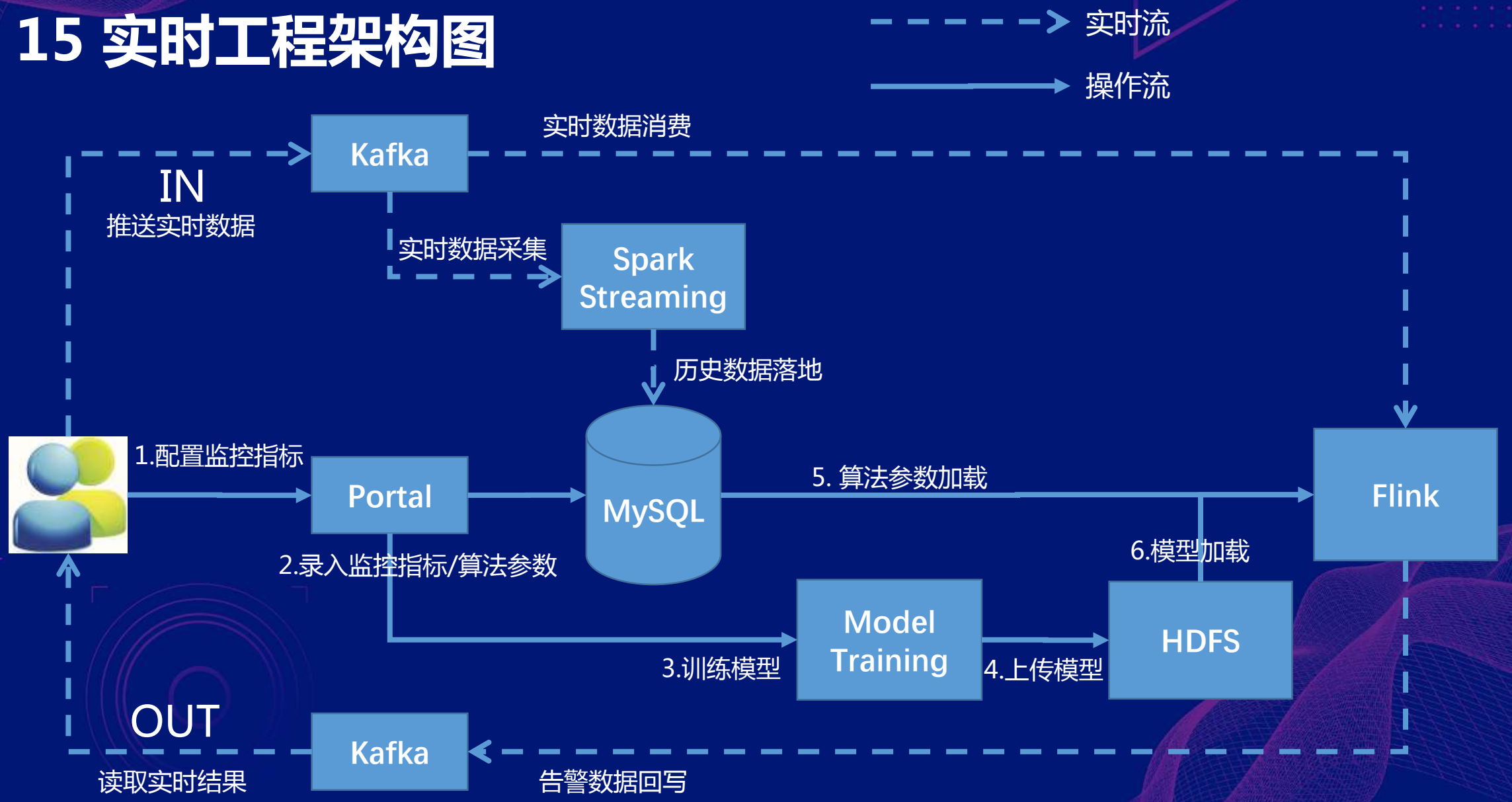
滑动窗口灵活

可基于event time统计计算

容错性佳

支持秒级数据采样和计算的能力

15 实时工程架构图


实时流
操作流
Kafka
实时数据消费
IN
推送实时数据
实时数据采集
Spark Streaming
历史数据落地
1.配置监控指标
Portal
MySQL
5. 算法参数加载
Flink
6.模型加载
2.录入监控指标/算法参数
Model Training
HDFS
3.训练模型
4.上传模型
OUT
Kafka
读取实时结果
告警数据回写

16 效果

实时流数据不落地

历史数据积累用于循环训练

识别新加入指标后触发实时训练

并行的模型训练和实时加载

较小的接入成本

17 局限

单指标单模型，算力消耗大

- 模型大小 10M+
- 训练时间 10min

绝对值量小的指标没有好的解决方案

- 震荡幅度大
- 通用告警难

波动剧烈的非周期型指标hold不住

- 随机性强
- 无明显特征

18 展望

通用模型节省算力

共享、回馈

本PPT来自2018携程技术峰会

更多技术干货，请关注“携程技术中心”微信公众号