# 封神-运维大脑 | 日志检测工具

原创

阿里云云栖号 于 2021-04-01 16:29:14 发布 1938 收藏
分类专栏： 云栖号技术分享 文章标签： 前端 日志

本文链接：https://blog.csdn.net/yunqiinsight/article/details/115377398



简介： 封神-运维大脑 | 日志检测工具

高德臣 阿里云智能 GTS-平台技术部-SRE 团队运维开发工程师

曾任职于北京新水源景大数据开发工程师，负责 ELK 日志管理系统、智慧农业平台以及图像识别等项目的开发工作。现就职于 SRE 混合云 TAM 团队，负责封神运维监控系统开发以及中国邮政混合云运维工作。

## 1. 背景目标

阿里云应用业务有问题，云平台监控可以发现问题，但并不能定位到问题根本原因，运维大脑监控底层日志，可快速定位问题原因，帮助现场运维同学解决问题。
运维大脑融合SRE方法，专注于深度运维的技术服务领域，帮助客户与现场，增强租户视角运维监控能力、提升平台视角问题定位效率、加强双维度容量性能运营能力。浓缩TAM现场运维经验，多样化地、标准化地、智能化地向客户输出运维能力与技术服务。

## 2. 开发设计

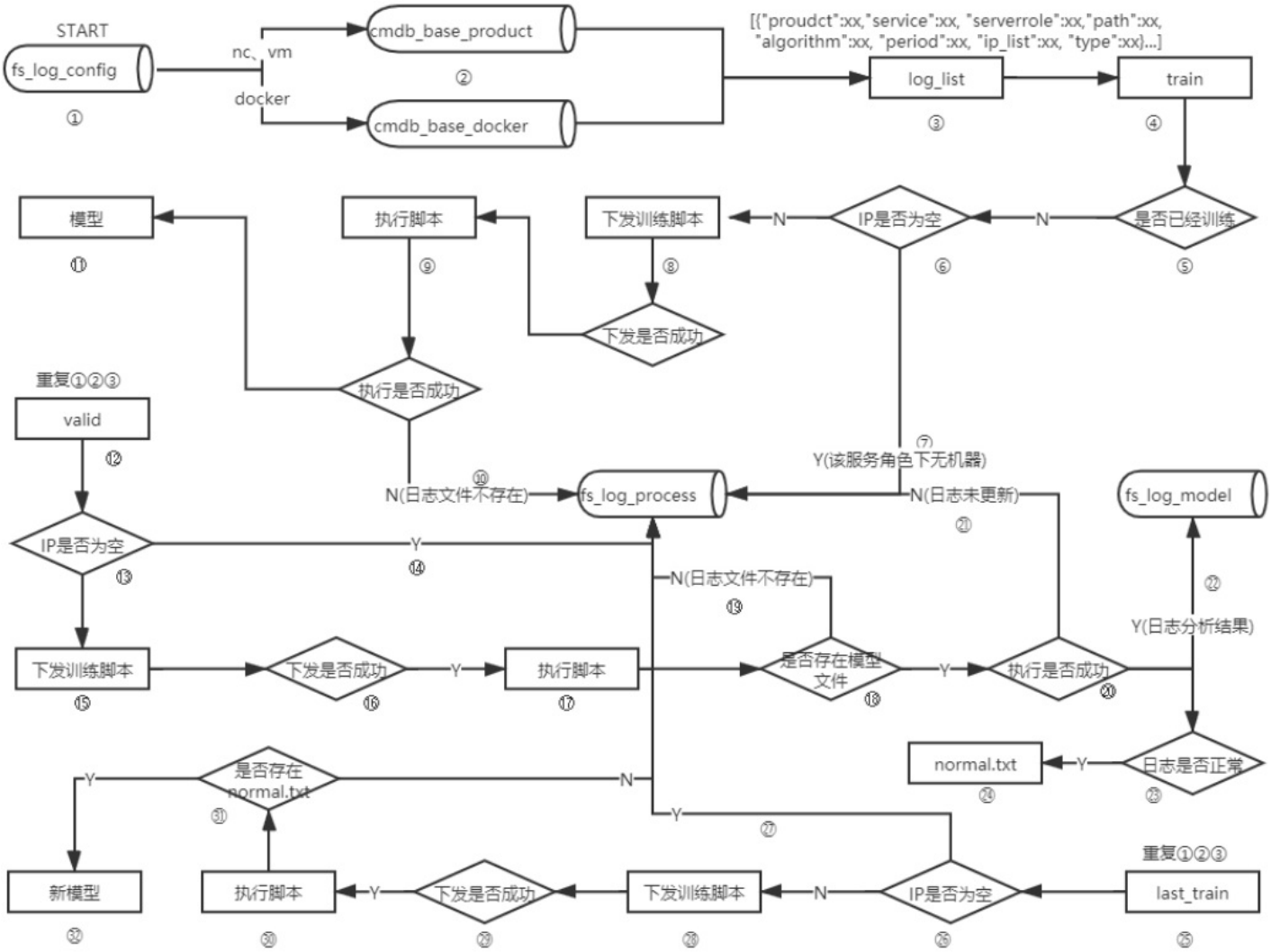


图1：流程图

## 2.1 日志配置

如图2所示，可通过运维大脑前端页面配置需要监控的日志，可单独新增也可批量导入。配置信息包括产品、服务、服务角色、日志类型（DOCKER 物理机 VM）、日志路径、监控周期、算法（ML-TOP ML-CP ML-KEY）、状态（开启\关闭）。


图2：日志配置

## 2.2 日志训练

前端配置日志信息存储到后台数据库，后台程序通过产品、服务、服务角色等条件查询相应的主机名。

```
sql_host = "select hostname from cmdb_base_product where product='{product}' and service='{service}' and serverrole='{serverrole}';"
```

图3：数据库

定时任务启动，根据获取到的主机名通过PSSH命令下发训练脚本到各个机器上。下发前判断各台机器是否已存在训练脚本，如果脚本已存在，则停止下发命令。

```
pscp.pssh -H '{}' /apsara/fs_monitor/fs_log_model/train.py /tmp/train.py'
```

图4：pssh

训练脚本开始工作：首先读取日志，通过正则进行英文分词（英文文本可通过NLTK库分词，中文文本可通过JIEBA分词进行切分，在这里选择最简单的PYTHON自带的RE模块根据特殊符号进行切分），统计总词数，并计算每个单词的词频。按词频排序将单词以二进制形式写入TOP模型文件，词频写入CP模型文件，如图5所示。
警告：文件命名最好以服务角色+文件路径的方式命令，否则在后续读取的时候可能会冲突。

```
-rw-r--r-- 1 root root 7322 Jan 12 11:34 EagleeyeConsole#_home_admin_logs_eagleeye-console_log_cp.model
-rw-r--r-- 1 root root 3571 Jan 12 11:34 EagleeyeConsole#_home_admin_logs_eagleeye-console_log_top.model
```

图5：文件命名

## 2.3 日志分析

定时任务启动，同训练过程初始化一样，首先判断各台机器是否存在分析脚本，如若不存在，进行下发命令。分析脚本开始工作：首先读取日志，区别于日志训练，分析脚本会根据前端配置的监控周期进行选取（比如监控周期为30分钟，则分析脚本会选取当前时间至30分钟之前的日志进行分析）。同训练脚本一样，读取日志后，进行文本分词，计算词数，统计词频。读取模型文件，根据不同的算法（算法这块在文章第三部分会单独进行讲述），计算算法权重值。对算法权重值进行阈值判断，超过阈值，会判断日志异常信息并从日志文件获取。分析结束，最后把产品、服务、服务角色、日志文件、日志级别（ERROR\INFO）、算法值、日志错误详情、监控时间等监控数据进行入库，并在前端页面进行展示，如图6所示。

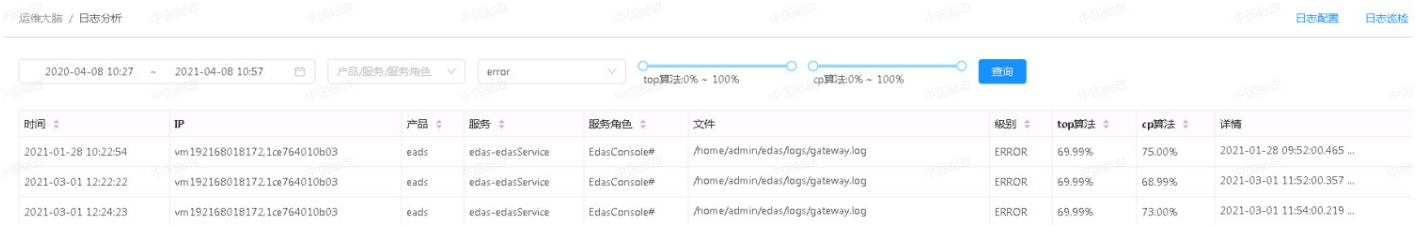


图6：日志分析

## 2.4 模型优化

训练模型初始化的弊端在于无法手动去打标签（正常\异常），所以对于初始化后的模型文件肯定不能是一个完全正常的模型，需要后续不断的去优化。
定时任务启动：还是一样的流程，完成读取文件、分词等工作后，生成的模型文件与源模型文件对比，对比方法与算法相同，阈值比分析阈值更低，低于阈值后，单词词频字典进行合并，按次序排序后分别写入源模型文件。至此，整个日志过程完成闭环操作。

### 2.5 日志巡检

日志巡检是对自身系统运行状况的监控，环绕整个闭环操作。日志训练、分析、模型优化通过定时任务去驱动，日志巡检对每一步操作过程进行成功判断，并对异常的操作进行原因分析，相关数据存储入库，并在前端进行展示，如图7所示。

图7：日志巡检

## 3. 算法逻辑

运维大脑所开发的算法借鉴了贝叶斯和文本相似度两大算法，以传统的自然语言处理方式对文本进行分析。

### 3.1 分词方式两种常用方式：结巴分词 和 nltk库分词

结巴分词适用于中文分词，分词原理为：
①基于Trie树结构实现高效的词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图（DAG)。
②采用动态规划查找最大概率路径, 找出基于词频的最大切分组合。
③对于未登录词，采用了基于汉字成词能力的HMM模型，使用了Viterbi算法
nltk库只能用于英文分词，除此以外还可用于词性标注和文本分析。
个人认为英文分词以空格或部分特殊符号进行切分即可：re.split()。

### 3.2 TF-IDF

TF-IDF是Term Frequency-Inverse Document Frequency的缩写，即词频-逆文档频率，用来刻画一个词语在某篇文档中重要程度，也就是说是否可以用该词语来代表某篇文档的主要内容。

- TF表示词频。给定几个关键词，在某篇文档中出现的次数最高者，则说明该文档与出现次数最高的词语关系最密切。用词语出现的次数除以文档的总词汇数，就是TF，当然此处在统计文档总词汇时，把类似于“了”、“的”、“地”、“即”等词语排除在外不予考虑。引入词频后，则某个词的词频越高，该文档与其关系也就越大。
  TF计算公式为： TF = 词语在文档中出现的次数 / 文档词语次数
- IDF表示逆文档频率。如果一个词语在某篇文档中出现的TF高，但是在语料库的其它文档中出现的次数少，则说明该词语对于文档分类具有重要作用，因此引入IDF来刻画此项数据，其值越大，说明该词语对于语料库来说具有越好的区分能力。如果某个词语在每篇文档里均出现，且出现的次数很接近，则该词语用来区分文档时效果便不好。
  IDF计算公式为： IDF = log(语料库文档总数/包含某词语的文档数+1)
- TF-IDF 值越大说明某个词语用类识别文档的区分度便越大。
- TF-IDF计算公式为： TF * IDF

### 3.3 文本相似度

Latent Semantic Indexing（LSI）从文本潜在的主题进行分析。LSI是概率主题模型的一种，另一种常见的是LDA，核心思想是：每篇文本中有多个概率分布不同的主题；每个主题中都包含所有已知词，但是这些词在不同主题中的概率分布不同。LSI通过奇异值分解的方法计算出文本中各个主题的概率分布，严格的数学证明需要看相关论文。假设有5个主题，那么通过LSI模型，文本向量就可以降到5维，每个分量表示对应主题的权重。
可参考文后资料[1]了解详情。

总结下文本相似度和贝叶斯算法的处理过程：

1. ML-LSI
   ①使用nltk库分词将日志文本切割。
   ②建立词袋模型。
   ③建立TF-IDF模型。
   ④构建一个query文本，确认主题，利用词袋模型的字典将其映射到向量空间。
   ⑤构建LSI模型，设置主题数为2（ERROR、INFO）。
   ⑥计算文本相似度。
2. ML-BAYES
   ①使用nltk库分词将日志文本切割。
   ②对处理之后的文本开始用TF-IDF算法进行单词权值的计算。
   ③去掉停用词。
   ④贝叶斯预测种类。

运维大脑日志分析算法包括：

1. ML-TOP
   weight = x * w
   x : 验证集top10新出现个数
   w : 单个词权重值 0.1
2. ML-CP
   weight = x / w
   x : 词频变化超过0.02数
   w : 词频变化总数
3. ML-KEY
   weight = x / w
   x：关键词日志行数
   w：日志总行数
4. ML-NUM
   weight = x * w
   x：异常日志行数
   w：0.1
   开发思路：
   ①获取日志k：v 求v平均值 报错num模型。
   ②对比新日志v值。

## 4. 总结

本期给大家介绍了封神系统运维大脑模块的相关知识，分享了机器学习中两个常用的文本分析算法。目前运维大脑所能达到的效果是可以把日志中报错进行识别并展示，但是我们的最终目标是可以识别出故障，虽然普通的报错可能对平台并没有太大的影响，但是频繁的报警并不利于运维工作的开展。
关于运维大脑暂时就介绍这么多，当前也确实存在一定问题，待后续完善后会再跟大家介绍，然后如果同学有更好的算法或者思路，欢迎讨论！
接下来的文章会陆续给大家介绍封神的其他模块，包括实时告警、运维大盘、报表分析、数据网关、妲己纣王、时序数据库等相关知识，敬请期待！

原文链接