

车流量监控项目

加入尚学堂，一起进步！

车流量监控-数据采集

- 数据从哪里来？

- 我们知道数据来源，比如网站，APP或者工业设备（比如卡口拍摄设备），实现实时数据采集，它首先有非常重要的一点就是所谓的埋点，也就是说，埋点，在网站的哪个页面哪些操作发生时，前端的代码比如javascript或者app android/ios，就通过网络请求Ajax; socket向后端的服务器发送日志数据。

如果是卡口信息，那么每次拍摄的信息都会传输到服务器端。

- 首先就是说网站或者页面设置埋点，那么就是你要跟前端的开发人员约定好，在哪些页面哪些操作发生的时候，网站的话就通过ajax引擎，APP的话就通过Socket网络请求，向后端的服务器发送指定格式的日志数据。卡口数据的话，是和厂商定制数据格式的，数据以指定的格式向服务器发送实时的数据。
- 接着通过Flume监控指定的文件夹，转移到HDFS里面去，实际大多数是放在Hive中因为Hive还有计算的能力，还有另外一条流程，实时数据，通常都是从分布式消息队列集群中读取的，比如Kafka，实时的log，实时的写入消息队列中，然后再由我们后端实时数据处理程序(storm、spark streaming)，实时从kafka中读取数据，log日志（架构图）

车流量监控-数据采集

- 数据从哪里来？

- 数据除了从Fulme中来，也有可能直接使用kafka 的producer角色往kafka中直接生产数据。
- 接下来就是大数据实时计算系统，比如说用storm、spark streaming开发的，可以实时的从kafka中拉取数据，然后对实时的数据进行处理和计算，这里可以有非常复杂的业务逻辑，甚至调用复杂的机器学习，数据挖掘，智能推荐的算法！然后实现实时的车辆调度，实时推荐等等。

数据采集不是我们的重点，我们的重点是用spark streaming开发实时计算的模块程序，从kafka里面去拉数据，拉数据以后来进行实时计算，比如后面会讲广告流量的实时统计

车流量监控-模块介绍

- 卡扣流量分析 Spark Core
- 卡扣车流量转化率 Spark Core
- 各区域车流量最高top5的道路统计 SparkSQL
- 稽查布控，道路实时拥堵统计 SparkStreaming

卡扣流量分析模块介绍

功能点

- 根据使用者（平台使用者）指定的某些条件，筛选出指定的一批卡扣信息（比如根据区域、时间筛选）
- 检测卡扣状态，对于筛选出来的所有的卡口（不代表一个摄像头）信息统计
 - 卡口正常数
 - 异常数
 - camera的正常数
 - camera的异常数
 - camera的详细信息（monitor_id:camera_id）
- 车流量最多的TopN卡扣号
 - 获取每一个卡扣的详细信息（Top5）
- 随机抽取N个车辆信息，对这些数据可以进行多维度分析（因为随机抽取出来的N个车辆信息可以很权威的代表整个区域的车辆）
- 计算出经常高速通过的TopN卡口（查看哪些卡扣经常被高速通过，高速，中速，正常，低速 根据三个速度段进行四次排序，高速通过的车辆数相同就比较中速通过的车辆数，以此来推）
- 跟车分析

实际企业项目中的使用架构

使用架构

- J2EE平台，前端页面，在页面中可以指定任务类型，提交任务的参数（比如时间范围，区域设定）
- 平台会接受到用户的提交请求，会调用底层封装的Spark-submit的shell脚本，怎么调用？
- 运行的作业可以获取到用户指定的筛选条件，然后根据筛选条件进行计算
- Spark任务的计算结果会写入到数据库中，比如MySQL，Redis等
- 最后J2EE平台可以通过前端页面，展示结果（表格或者图表的方式展示数据库中的结果）

数据概念普及

数据概念普及

- 卡扣号：在一条道路相同位置会有两个卡扣，这两个卡扣的编号是不同的，分别拍摄不同方向的车辆
- 摄像头编号：每一个卡扣拍摄的是一个方向的车辆，每一个方向都会有多个不同的车道，每一个车道对应一个摄像头，所以卡扣号与摄像头的对应关系是一对多的关系

基础数据介绍

- monitor_flow_action表
 - date 日期 天
 - monitor_id 卡口号
 - camera_id 摄像头编号
 - car 车牌
 - action_time 某个摄像头拍摄时间 s
 - speed 通过卡扣的速度
 - road_id 道路id
 - area_id 区域ID

基础数据介绍

- monitor_camera_info表 某一个卡扣对应的摄像头编号
 - monitor_id : 卡扣编号
 - camera_id : 摄像头编号

大数据开发流程

- 开发流程
 - 数据调研（对底层的数据的表结构进行调研，分析和研究）
 - 需求分析（与PM讨论需求，画原型图 axure）
 - 基于讨论出来的结果做出技术方案（某个难点用什么技术，数据库选型）
 - 具体实施

需求分析

- 需求分析
 - 按条件筛选卡扣信息
 - 可以指定 不同的条件，时间范围、区域范围、卡扣号等 可以灵活的分析不同区域的卡扣信息
 - 监测卡扣状态
 - 对符合条件的卡扣信息，可以动态的检查每一个卡扣的状态，查看卡扣是否正常工作，也可以查看摄像头
 - 车流量最多的TonN卡扣
 - 查看哪些卡扣的车流量最高，为什么会出现这么高的车流量。分析原因，例如今天出城的车辆非常多，啥原因，今天进城的车辆非常多，啥原因？ 要造反？ 这个功能点里面也会拿到具体的车辆的信息，分析一下是京牌车造成的还是外地车牌？
 - 在符合条件的卡扣信息中随机抽取N个车辆信息
 - 随机抽取N辆车的信息，可以权威的代表整个区域的车辆，这时候可以分析这些车的轨迹，看一下在不同的时间点车辆的流动方向。以便于道路的规划。
 - 计算出经常高速通过的TopN卡口
 - 统计出是否存在飙车现象，或者经常进行超速行驶，可以在此处安装违章拍摄设备
 - 跟车分析
 - 计算出所有车是否被跟踪过，然后将结果存储在MySQL中，以便后期进行查询

技术方案