

信令处理平台

一 离线信令处理平台

目标：使用阿里云平台，数据导入到信息提取展示一键化。

优点：阿里云平台经济，稳定；一键化知识提取加速后续产品迭代。

分解：1 先实现命令行形式的一键数据导入分析，获取处理结果；2 输入，输出可视化。

能做到的：得到后续分析的数据基础，标准化数据。

不能做到的：Ad-hoc形式的分析: SQL/UDF等，但可以持续将通用分析持续集成到处理平台上。

1 配置文件

1 原始信令数据的结构描述

- 列分隔符

- 行分隔符

- 字段意义

2 过滤规则定义

3 原始文件形式

- 单一文件

- 文件夹

- 数据所在位置

4 项目名称，用于建表

2 原始信令数据结构：

定义通用的信令处理平台，目前字段如下：

0 partition

1 uuid

2 time

3 lon

4 lat

5 cell_id

7 cell_name

8 is_roam

9 in_room

10 phone_number

11 x1

12 x2

13 x3

3 平台具有的功能

1 建立项目：调用阿里云API建立相应表、分区

2 定义，修改配置

3 启动数据导入

4 观察数据导入进度

5 数据导入完成后，观察原始数据的分布情况，查看异常数据情况

- 可视化显示各个字段有多少异常数据

6 基础数据去重, 可视化数据去重量

配置关心的字段，进行数据去重，比如time, uuid, lon, lat, is_roam, in_rom, phone_num粒度的去重

7 在去重后的原始数据基础上，计算得到

- 1 基站信息：

- 1 基站总数量：计算总数量

2 各个基站uniq ID: 生成ID

3 单个基站的总数据量, 单个基站按天的数据量: 可视化, 可导出数据

4 基站的区域范围, 给出经纬度范围, 横纵多少公里: 可视化

5 基站密度: 可视化, 可导出数据

能够得到什么区域内, 覆盖尽量多的数据。

2 人口信息:

1 人的总数量

2 各个人的总数据量, 各个人按天的总数据量: 可视化, 可导出 (数据量大)

3 人-数据量的分布统计:

0-10条数据人数, 总数据量占比

10-50条数据人数, 总数据量占比

50-100

100-150

150-200

200-250

250-300

300-350

350-400

等

要有概率密度图, 和累计密度图

3 可导出总体分析报告, 用于分析

8 根据数据分布情况, 设置数据过滤方案, 通过配置启动过滤。

过滤点:

1 基站经纬度: 去掉异常区域外的数据, 输入区域范围

2 基站单个节点数据量: 给出可信值范围

3 基站密度异常检测 (按照给定格子大小, 看单位距离内基站个数): 给出可信值上限, 可以用此来合并基站数据

4 单个用户产生的信令数据量: 输入可信范围

9 经过了去重, 过滤, 至此有了后续分析的基础数据, 可以继续一下分析:

1 得到人的时空信息: 要能可视化, 给定时间范围, 画出路线

xxxx |

14258@0#1114319@1#7114201@8#1114319@9#9114201@18#1114208@19#1114319@20#4 |

2 得到某个区域的热度: 可视化

3 分析某个区域的住宅人口, 得到这些住户白天的去处: 可视化

4 分析某个区域的工作人口, 得到这些工作人口的晚上睡觉地点: 可视化

4 数据导入:

1 使用tunnel工具, 支持文件夹/大文件, 可以设置多项参数(行列分隔符)

tunnel的缺点: 不支持多线程, 要手动“多线程”

2 使用阿里云提供的API, 多线程的数据传输, 输入配置文件

5 数据去重:

SQL API实现

6 数据过滤:

SQL API实现

7 Ad-hoc分析:

SQL/UDF等

8 架构:

- 1 大数据层：存储和计算由阿里云提供
- 2 中间层：用于存储分析结果（聚合数据），可用Mysql/Mongodb存储。NodeJs搭建API Server
- 3 展示层：Angular/Bootstrap等前端框架实现

预计工时：

二 实时信令处理平台

接收实时信令数据，TODO