日志行为分析

5140379026 徐瑾卿

日志行为的记录通过 AOP 来实现，在点击商品详情 点赞 加入购物车 提交订单 付款等重要行为处加入切片，记录日志。

一 大数据分析

- 用户浏览行为

  对于点击详情，还需要记录用户点击详情开始时间和结束时间(点击其他地方)，时长越久说明用户越感兴趣。

  对于加入购物车，要记录是哪几本书一同被加入购物车的。

Userid	Bookid	Time	Behavior
Alice	1	2017-06-03	点击详情/点赞/加入购物车

- 用户购买行为

Userid	Bookid	Time	Behavior
Alice	1	2017-06-03	提交订单/ 付款

  - **购物篮分析**

    分析用户的订单数据，统计哪几本书被一起购买，数据量大了以后用 Aprior 算法做关联分析，找出频繁项集后，计算关联规则，找出符合置信度的关联规则，从而在用户点击某一本书后可以为其推荐另一本书。

  - **协同推荐**

    分析用户的相似度，几个层面上的分析，首先是购物习惯，分析哪些用户经常买类似的书，其次是浏览行为，点击查看了哪几本书的详情，购买行为的权重比浏览行为略高些，计算出用户间的距离后，为用户推荐相似用户喜欢的书籍。

  - **异常行为监测**

    如果用户多次输错支付密码，或者是登录密码，则该用户的信用等级下降。或者用户尝试通过输入进行脚本攻击 sql 注入，应该对该用户发出警告。对于反复取消订单的用户，也应该降低信用等级。

  - **指标统计**

    统计分析每小时的登录人数和下单人数，高峰期多开几台机器做负载均衡。

➢ 用户分级

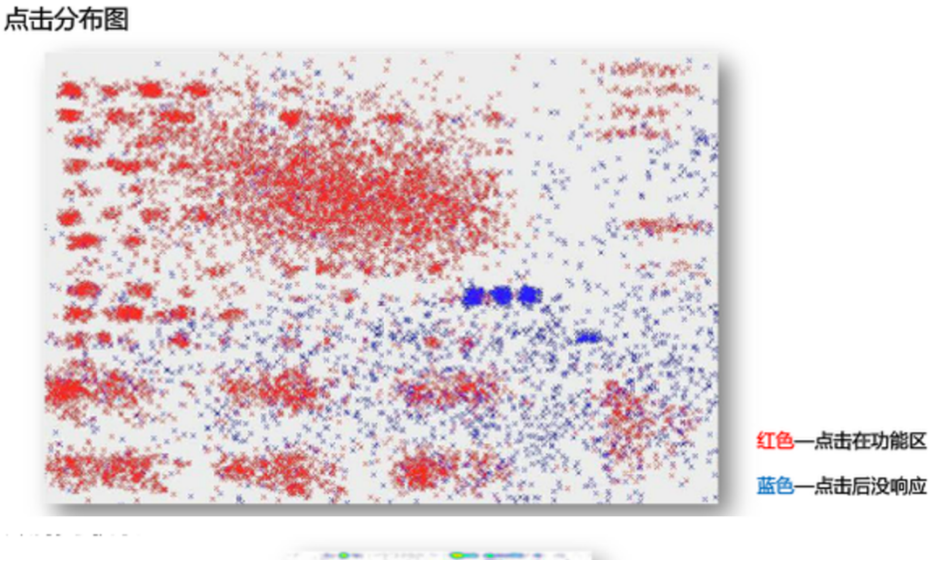
衡量用户行为有三个维度，粘性(访问频率，访问间隔时长)，活跃(平均停留时间，平均访问页面数)，产出(订单数，客单价)。根据访问间隔 访问频率 消费金额，可以将客户分为 8 类。此外，还可以统计访问用户中的新老用户比，来确定网站的用户增长量。还可以统计用户的平均访问次数，来确定流失用户比例。

访问间隔	访问频率	消费金额	客户类型
↑	↑	↑	重要价值客户
↑	↓	↑	重要发展客户
↓	↑	↑	重要保持客户
↓	↓	↑	重要挽留客户
↑	↑	↓	一般价值客户
↑	↓	↓	一般发展客户
↓	↑	↓	一般保持客户
↓	↓	↓	一般挽留客户

注：“↑”表示大于均值，“↓”表示小于均值

➢ 用户行为轨迹

用户行为轨迹分为几个阶段，bookstore 可以在不同的阶段量化指标。在熟悉阶段，可以统计平均停留时长，页面偏好，在试用阶段，统计注册转化率，在使用阶段统计订购量 订购频率，在忠诚阶段，统计流失率。可以借助可视化工具来定性分析。



➢ 聚类对书打标签

除了传统意义上的类别，用户也可以为书打上标签，书可以根据属性和标签进行聚类，假如一本新的书入库，可以计算它与几个簇中心的距离，

从而选取一个最优的。

- 大数据分析不能解决的问题

  大数据不能解释因果关系，比如为什么某类用户喜欢某本书，A 与 B 两本书为什么会被一同购买，为什么网站会有高峰期和低谷期，为什么一些用户成为了活跃用户，一些用户却流失了等等。要解决这些问题需要一些领域知识和生活常识。

二 map reduce 处理数据

形式化描述：

Map： (k1,v1) -> list(k2,v2)
Reduce：(k2,list(v2)) ->list(v3)

Map reduce 适合大数据的批量处理。由于每次需要遍历所有数据，map reduce 适合实时性要求不高的工作，比如日志分析。

以用户偏好书籍种类挖掘点为例

1 用户提交 map reduce 程序到 master，master 把输入文件划分成若干分片(split)。Master 和 worker 启动相应进程。
2 master 分配任务给 map
3 日志数据里包含哪个用户购买了哪本书，把哪些书放入了购物车，点击查看了那本书的详情，为哪本书点了赞等。这几种情况对应的权重不同，Map 读取一个日志数据的分片，搜索这些书所对应的标签，输出所有这样的标签，格式为 key：用户-标签-权重，value：1(value)代表频率。Map 将结果存在本地，把结果分成 R 个分片进行存储，R 对应 reduce 的数量。
4 对所有的中间结果按照 key 排序。
5 map 告知 master 中间结果的位置，master 指定 reduce 远程读取数据，reduce 对中间结果进行汇总处理，得到 key：用户-标签-权重，出现次数。滤去次数小于 2 的结果，就得到用户不同程度的兴趣标签了。

三 bigtable 存储

bigtable 的存储单元是由行键，列键和时间戳决定的，其中时间戳对应的是版本。以用户对图书的行为数据为例，以用户名为行键，书名 操作 时间 3 个为列键，时间戳就是插入的时间。其他的登录数据 搜索数据等也可以类似的方法存储。由于 bigtable 的列式存储特性，按 bookname 查询或者按操作查询效率很高。

插入数据：

```
void insert(string username,string bookname,string operation,Timestamp t){}
//Open the table
Table *T=OpenOrDie("/bigtable/web/logdata");
//write
RowMutation r1(T,username);
r1.Set("bookname",bookname);
r1.Set("operation",operation);
r1.Set("time",t);
Operation op;
Apply(&op,&r1);
```

读取数据：

```
void read(String username){
Scanner scanner(T);
ScanStream *stream;
stream->SetReturnAllVersions();
scanner.Lookup(username);
for (; !stream->Done(); stream->Next()) {
printf("%s %s %lld %s\n",
scanner.RowName(),
stream->ColumnName(),
stream->MicroTimestamp(),
stream->Value());
}
}
```