# 数据结构

淘宝数据：本人注册号，本人支付宝，收货人姓名，收货人地址，收货人固话，收货人手机

京东数据：本人注册号，收货人姓名，收货人地址，收货人手机

有赞数据：微信粉丝ID，本人注册号，收货人城市，收货人区域，收货人姓名，收货人省份，收货人地址，收货人邮编，收货人手机

全局变量：代他人购物比例，注册号字符相近，地址相近

微信数据：openid, unionid, 用户昵称，性别，城市，国家，省份，订阅时间

# 表结构设计

淘宝，京东，有赞的数据表分别以如下形式设计：

主订单表

子订单表

买家表

联系人表

详见数据导入详细设计

微信数据表以如下形式设计：

用户表

互动表

同人用户表

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **备注** |
| id | long |  | 自增主键 |
| gmt\_create | datetime |  |  |
| gmt\_modified | datetime |  |  |
| is\_deleted | char | 2 | y/n |
| created\_by | varchar | 20 |  |
| modified\_by | varchar | 20 |  |
| opUserId | long |  | 同人用户id |
| tbUserId | long |  | 淘宝用户id |
| jdUserId | long |  | 京东用户id |
| wxUserId | long |  | 微信用户id |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 算法步骤

## 计算全局变量

以支付宝账号为手机号的交易为分析源， 以支付宝账号与收件人手机号相等的交易为判断依据。

以有多笔交易记录并且收件人数据不同的为分析源，结合微信的地理位置数据（包含经纬度，城市等）

### 代他人购物比例

### 注册号字符相近

### 地址相近

## 计算计算项

淘宝与微信，包含的计算项有：

|  |  |
| --- | --- |
| 淘宝 | 微信 |
| 买家昵称 | 昵称 |
| 收件人手机 | 绑定手机 |
| 支付宝账户号 | 绑定手机 |
| 收件人姓名 | 昵称 |
| 收件人城市 | 城市 |

京东与微信

|  |  |
| --- | --- |
| 京东 | 微信 |
| 注册号 | 昵称 |
| 收件人手机 | 绑定手机 |
| 收件人姓名 | 昵称 |
| 收件人城市 | 城市 |

## 计算最终同人概率

同人概率 = 1 – (1 - X1) \* (1 – X2) \* …. \* (1 – Xn)

# 同人服务

Boolean OnePersonService.isOnePerson(long wxUserId, long tbUserId)

微信用户与淘宝用户是否是同一人

Boolean OnePersonService.isOnePerson(long wxUserId, long jdUserId)

微信用户与京东用户是否是同一人

Boolean OnePersonService.hasOnePerson(long wxUserId)

是否有其它平台的用户与微信用户是同一人

User OnePersonService.getOnePerson(long wxUserId)

生成同人数据，如果没有在其它平台找到同人数据，返回null

# 其它

模型需要进化，在更多的业务场景应用，分析更多的数据，更大的提高准确性。

# 业务设计

一个收件人对应一个微信账号，对应上的微信账号对应淘宝账号的全部数据

增量数据计算，即客户有数据更新时，绑定重新计算

客户验证页增加淘宝ID与京东ID，输入淘宝ID与京东ID，验证优先级提升许多

# 账号体系

一个速牛账户，对应多个店铺账户，对应一个微信服务号