题目和数据保存在：<https://pan.baidu.com/s/1H0UC3fel3TedZFnLBbqNOA> 提取码：llp3

代码文件夹中实现的功能为以下论文的第二节和第四节内容

1. 引言（写研究背景、目的和文献综述）

二、数据处理（数据来源与Hadoop预处理）

三、数据分析（你们自己起个题目吧）

四、人际关系网络研究

五、总结

参考文献（有就写吧哈哈哈）

附录（贴代码）

二、数据处理

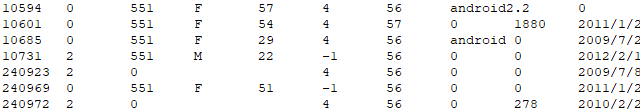
2.1 数据来源

本项目所处理的数据来自于全国大学生数学建模竞赛某赛题的附件，主要的数据集有两个：用户数据和通话数据，它们分别记录的是合肥市电信运营商在2012年2月份公布的部分用户的个人信息以及所有手机号码在2012年2月份的所有通话记录。具体的数据集情况如下表2.1-1所示：

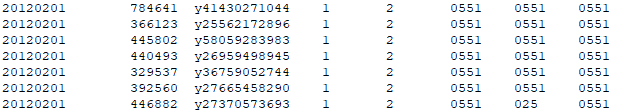
【表2.1-1】原始数据集

|  |  |  |
| --- | --- | --- |
| 名称 | 用户数据 | 通话数据 |
| 格式 | 文本文件 | 文本文件 |
| 大小 | 33.9M | 5.67G |
| 字段 | 本网手机号码、城乡标识、身份证归属地、性别、年龄、用户等级、用户的行业、手机操作系统类型、手机价格、号码办理日期、是否3G用户、是否VIP用户、套餐档次（元）、消费金额、流量费、市话费、长途费、漫游费、通话费、增值费、短信费、停机次数、资费变更次数、上网流量、上网时长 | 日期、主/被叫号码、主/被叫号码运营商（1：电信 2：移动 3：联通 其他为不详）、主/被叫号码归属地、主/被叫号码漫游地、通话开始时间、通话结束时间、通话时长、通话类型（1：市话 2：长途 3：漫游）、主叫蜂窝号码 |

【图2.1-1】用户数据



【图2.1-2】通话数据



2.2 数据结构设计

本项目的研究目的主要有两个：一是为电信客户进行分类画像，方便运营商进行精准营销；二是简单地验证人际关系网络模型中的六人定律，除此之外，就是一些附加的、简单的统计分析任务。针对上述两个主要的目标，尤其是前者，我们除了依赖于原有的三十多万条用户信息之外，还需要从用户的通话记录中提取出更多不同维度的信息；而对于后者，则主要是需要提取出任意两个客户之间的人际关系。基于上述分析，我们对这两个研究目标所需要依赖的数据，也就是我们本次预处理所需要得到的结果，有了如下的预期，数据结构如下表2.2-1和2.2-2：

【表2.2-1】绘制用户画像所需数据结构

|  |  |  |
| --- | --- | --- |
| 键值对 | 字段 | 字段说明 |
| key | 主叫人 | 加密电话号码，全为电信号码 |
| value | 通话总次数:通话总时间 | 本月通话次数和通话时间总和 |
| 最常漫游地 | 主叫人最常通话地，记录的是某地区号 |
| 通话类型:通话次数:平均通话时间 | 共三类：1: 市话 2: 长途 3: 漫游 |
| 通话时段:通话次数:平均通话时间 | 共24个时段：0-23 |

【表2.2-2】构建人际关系网络模型所需数据结构

|  |  |  |
| --- | --- | --- |
| 键值对 | 字段 | 字段说明 |
| key | 主叫人 | 加密电话号码，全为电信号码 |
| 被叫人 | 加密电话号码，全为电信号码 |
| value | 通话总次数:通话总时间 | 本月与被叫人互通电话次数与时间加总 |
| 主叫人最常漫游地 | 主叫人最常通话地，记录的是某地区号 |
| 被叫人最常漫游地 | 被叫人最常接通地，记录的是某地区号 |
| 通话类型:通话次数:平均通话时间 | 共三类：1: 市话 2: 长途 3: 漫游 |
| 通话时段:通话次数:平均通话时间 | 共24个时段：0-23 |

2.3 Hadoop数据处理

通过上文对数据处理预期结果的分析，我们发现，两个研究目标所需依赖的数据结构基本一致，只是在绘制用户画像时不需要用到被叫人信息，这只需要在Map函数中删除这一元素即可。所以下面我们只阐述构建人际关系网络模型所需的数据结构的处理和输出过程，而不对前者所依赖的数据结构加以赘述。

2.3.1 Map函数

通过观察原数据，在将每一条记录按制表符”\t”切分后，我们不需要的信息有：日期(元素0)、主/被叫号码运营商(元素3/4)、主/被叫号码归属地(元素5/6)、通话结束时间(元素10)和主叫蜂窝号码(元素13)。这些信息我们都会在Map函数中将其直接删除，则具体的Map函数数据处理过程如下：

Ⅰ. 读入每一行数据后将其转换为String类型后按制表符”\t”切分成字符串数组；

Ⅱ. 将通话开始时间(元素9)按”:”切分后仅保留小时；

Ⅲ. 将通话时长(元素11)的单位由秒转换为分钟，保留两位小数；

Ⅳ. 按顺序组合处理完的通话开始时间、通话时长、通话类型(元素12)、主叫号码漫游地(元素7)和被叫号码漫游地(元素8)，中间用制表符”\t”拼接，作为值；

Ⅴ. 用制表符”\t”拼接主叫号码与被叫号码，作为键；

Ⅵ. 判断被叫号码是否为电信号码，即首字母不为’y’或’g’，是则输出键值对，反之则不输出；

最终从Map函数中输出的键值对结果示例如下所示：

|  |  |
| --- | --- |
| 结构 | <键, 值>: <主叫人 被叫人, 开始时间 时长 类型 主叫人漫游地 被叫人漫游地> |
| 示例 | <key, value>: <389699 95812, 02 0.25 2 0564 0551> |

2.3.2 Reduce函数

在汇总了相同键的键值对以后，我们相当于是有了两个人之间的所有通话记录，此时按照前文所述的数据结构要求，我们需要找出两人之间总共的通话次数与通话时间、彼此最常通话的地点、和各种类型、在各个时间段内的通话总数与平均通话时间。具体的Reduce函数数据处理过程如下：

Ⅰ. 定义phonecall变量和lastingtime变量记录通话总数与总时间；

Ⅱ. 定义type数组与type\_minutes数组和hours数组与minutes数组分别记录不同类型与不同时间段内的通话次数与通话时间，在加总前要先对通话类型与通话时间段进行判断后分类到数组相应的位置上去；

Ⅲ. 定义start\_place与destination字符串数组列表，记录每通电话的主叫人漫游地与被叫人漫游地；

Ⅳ. 在遍历完所有相同键对应的不同值之后，已经汇总得到了两人间的通话总数与通话时间，再对每种通话类型与各个时间段内的通话时间取平均即可；

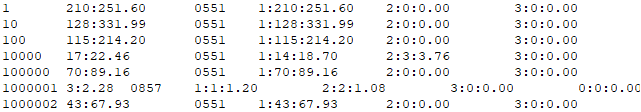
Ⅴ. 在得到了主叫人与被叫人的所有漫游地之后，利用先前设计好的MostPlace函数，结合了哈希表计数求算出主叫人与被叫人的最常漫游地，即在start\_place与destination字符串数组列表中出现次数最多的字符串；

Ⅵ. 将结果用制表符”\t”拼接后输出，其中通话类型/通话时间、通话总数和通话总时间/平均时长之间用”:”连接；

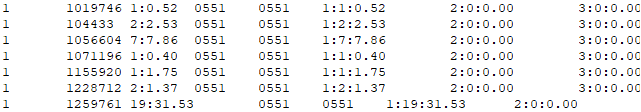
最终从Reduce函数中输出的键值对结果示例如下所示：

|  |  |
| --- | --- |
| 结构 | <键, 值>: <主叫人 被叫人, 通话总数:通话时间 主叫人漫游地 被叫人漫游地 市话:个数:平均时长 长途:个数:平均时长 漫游:个数:平均时长 0点:个数:平均时长 23点:个数:平均时长> |
| 示例 | <key, value>: <1 1019746, 1:0.52 0551 0551 1:1:0.52 2:0:0.00 3:0:0.00 0:0:0.00 23:0:0.00> |

【图2.3.2-1】绘制用户画像所需数据处理结果



【图2.3.2-2】构建人际关系网络模型所需数据处理结果



四、人际关系网络研究

4.1 研究目的

曾经有一个股票专家做过一个实验，他将一封有关股票信息的电子邮件发送给一个陌生人，并要求这个陌生人把这封电子邮件转发给他认识的一个热爱炒股的人。当这封邮件第六次被转发的时候，竟然转发回到了股票专家的手中。经过无数次的实验，股票专家发现，从邮件的发出到收回，平均都只经过了六次转发，由此他得到了一个初步的人际关系结论：任何两个人之间的关系带，基本确定在六个人左右，即两个陌生人之间，可以通过六个人来建立联系。在讲述FaceBook的创始人马克·扎克伯格的个人传记电影《社交网络》中，就有提到，扎克伯格就是受到六人定律的启发才创建了FaceBook的。本项目在前文中已经处理获取到了合肥市电信用户间的通话关系，即可以简单地来验证六人定律的成立与否。

4.2 研究方法

将每一个电信用户都作为图中的一个结点，存在通话关系的两个结点间用无向边连接起来，这样就能够得到一张无权无向图。而验证六人定律的方法就是任选两个结点计算它们之间的最短跳数，即最少需要经历的中间结点个数。通过大量重复的随机选取试验，观察最短跳数是否保持在6次左右。我们选取计算最短跳数的方法是广度优先搜索算法（BFS）。BFS算法的基本步骤如下：

Ⅰ. 选取初始结点入队；

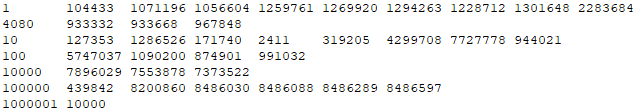
Ⅱ. 出队队列中的第一个结点，遍历其所有边结点，观察是否有终止结点的存在；

Ⅲ. 若无终止结点，则入队所有边结点，对队列中的每一个结点都重复Ⅱ的操作；

Ⅳ. 找到终止结点，或队列为空，搜索结束；

由于本项目的通话记录中涉及到的电信用户数量，目前有发现的，已经超过了一百万，且我们并不知道具体的结点数是多少，所以如果采用邻接矩阵的方式存储所有结点，会造成很大的空间浪费，即便是使用三元组的方式来存储此稀疏矩阵，也不便于我们使用广度优先搜索算法。于是我们决定采用邻接表的方式存储所有的电信用户数据和他们之间的关系。在Map函数中选择保留主叫人作为键，被叫人作为值输出；在Reduce函数中，将相同主叫人的所有被叫人数据用制表符”\t”拼接起来后输出，就能够得到创建邻接表所需的数据结构如下图4.2-1。

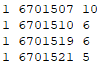
【图4.2-1】人际关系网络研究所需邻接表数据结构



4.3 结论

最终我们选取了第一个结点计算它到该图中其余所有结点的最短跳数，部分结果如下图4.3-1所示：

【图4.3-1】第一个结点到其余所有结点的最短跳数



将数据导入到SQL，对跳数取平均，在1248019个二人关系中，跳数的平均值为6.215829，即对于第一个联系人来说，他只需要平均打电话转拨6次，即可打给合肥市所有的电信用户。按跳数进行分类汇总后发现，绝大多数二人关系之间的跳数都集中在6附近，具体结果如下表4.3-1：

【表4.3-1】第一个结点到其余所有结点的最短跳数统计分析

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 跳数 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 数量 | 43 | 609 | 6656 | 60778 | 273876 | 447336 | 296714 | 113831 |
| 跳数 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 数量 | 34751 | 9688 | 2752 | 700 | 231 | 38 | 12 | 4 |

4.4 缺陷与改进

根据上述研究的结果，我们可以简单地证明六人定律至少在合肥市的电信用户之间是成立的，但事实上，该研究过程还有如下几点不足之处：

Ⅰ. 仅检验了第一个联系人，且区域仅限制在合肥市的电信用户之间，若扩大范围，很有可能需要有更长的跳数才能够遍历到所有的结点；

Ⅱ. 本算法在设计的过程中将二人间的关系视为是有向的而非无向，虽然在大多数情况下，存在一方呼话给另外一方，另一方理应当也会有回拨给第一方的时候，所以大部分的二人关系应该都是双向的，故该算法采用有向图还是无向图对计算结果的影响应该不会太大，有则可能是真实的跳数会比我们目前的计算结果要小一点点；

Ⅲ. 存在推销电话或是类似于10000之类的高频呼出或呼入的干扰电话号码，会使得绘制的人际关系网络存在一些出度或入度非常巨大的中心结点，任意一个呼话人都有可能通过这些中心结点轻而易举地联系到大面积的陌生人，而在现实中这是不允许的；

故该模型后续的优化方法就是加入对推销电话和公众服务类电话号码的识别，将其从图中剔除掉之后，再重复上述研究方法来验证六人定律的成立。