目录

[一、 数据关联界面 1](#_Toc509392288)

[0.总体流程概要 2](#_Toc509392289)

[1.电围经纬度 2](#_Toc509392290)

[1.1 输入内容 2](#_Toc509392291)

[1.2 输出内容 2](#_Toc509392292)

[1.3 算法或逻辑 2](#_Toc509392293)

[1.4 展现形式 6](#_Toc509392294)

[2.短信或其他短文本 14](#_Toc509392295)

[2.1 输入内容 14](#_Toc509392296)

[2.2 输出内容 14](#_Toc509392297)

[2.3 算法或逻辑 14](#_Toc509392298)

[2.4 展现形式 17](#_Toc509392299)

[3.关系网 20](#_Toc509392300)

[3.1 输入内容 20](#_Toc509392301)

[3.2 输出内容 21](#_Toc509392302)

[3.3 算法或逻辑 21](#_Toc509392303)

[3.4 展现形式 21](#_Toc509392304)

[4.现金流 21](#_Toc509392305)

[4.1 输入内容 21](#_Toc509392306)

[4.2 输出内容 21](#_Toc509392307)

[4.3 算法或逻辑 21](#_Toc509392308)

[4.4 展现形式 21](#_Toc509392309)

[5.出行 22](#_Toc509392310)

[5.1 输入内容 22](#_Toc509392311)

[5.2 输出内容 22](#_Toc509392312)

[5.3 算法或逻辑 22](#_Toc509392313)

[5.4 展现形式 22](#_Toc509392314)

[6.住宿 23](#_Toc509392315)

[6.1 输入内容 23](#_Toc509392316)

[6.2 输出内容 23](#_Toc509392317)

[6.3 算法或逻辑 23](#_Toc509392318)

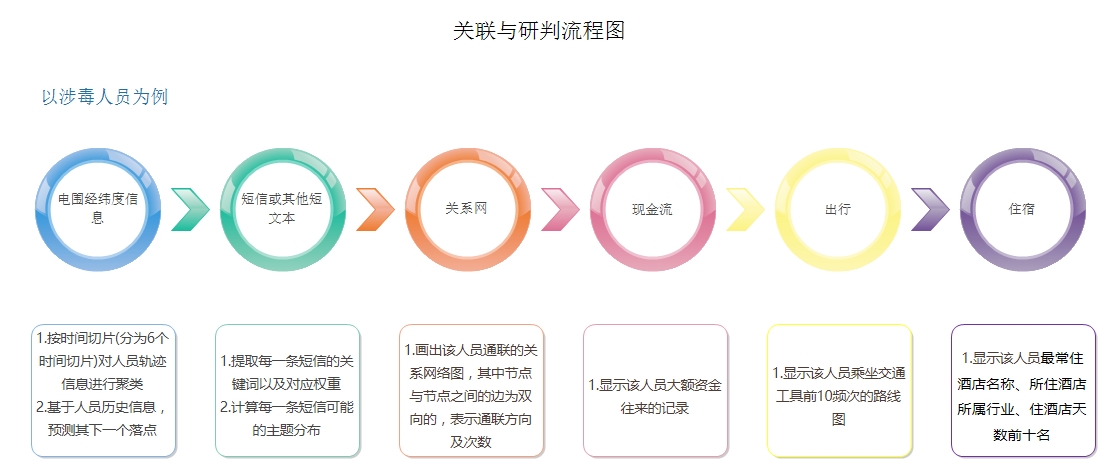
[6.4 展现形式 23](#_Toc509392319)

[二、 多粒度画像界面 26](#_Toc509392320)

1. 数据关联界面

0.总体流程概要

本界面拟采用半自动的展现形式，即点击下一步即可查看下一个数据内容，而按上一步则可再次查看上一个数据内容，其类似于软件的安装过程。大大简化了办案警员的查询流程，避免了他们漫无目的地点击所有内容。同时本界面还提炼了研判某一个专题所需所有的数据，形成了完整的逻辑链。下图为本界面的流程图。



以下以涉毒人员为例，详细介绍每个数据模块输入输出以及其展现形式和背后支撑算法。

1.电围经纬度

* 1. 输入内容

人员手机号，查询时间段

* 1. 输出内容

1）不同时间片下的聚类结果，包括单个人员以及其所属群体的结果。2）对于该人员下个可能去的地点进行排序。

* 1. 算法或逻辑

本功能使用meanshift算法对于该人员经纬度信息进行聚类。大多数聚类算法并不适用于现实数据集。因为它们有的依赖于集簇个数这一先验知识，有的则假设空间中所有集簇都有相同的形状，而不适合处理现实数据的复杂环境。而MeanShift算法是一个非参数聚类技术，它不要求预先知道聚类的类别个数，对聚类的形状也没有限制。

MeanShift算法建立于核密度估计的基础上。核密度估计是一种非参数的随机变量密度函数估计方法。它又被称为Parzen窗算法。给定一个核K,带宽参数h, d维点集的核密度估计如下：



而核函数K通常使用的是放射状对称核函数。



其中是核函数的轮廓函数，且。归一化常量恒为正，其目的是为了使的积分为1。

将1.2中的核函数代入1.1中，可得基于核函数的概率密度函数估计。



MeanShift算法将每个点迭代地沿着数据集合中数据最密集的地方移动，所谓最密集即是概率密度最大的地方，因此我们对概率密度函数求梯度，梯度的方向即为概率密度增加最大的方向。



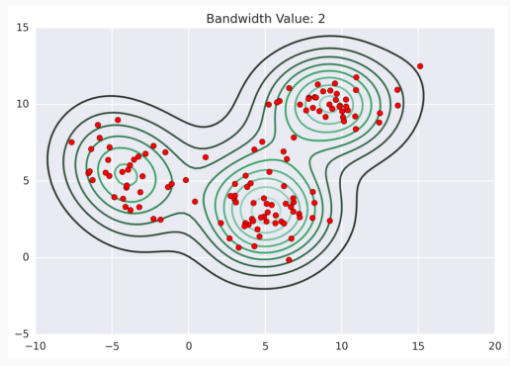
令，假设除了有限个点，轮廓函数的梯度在均存在。以作为轮廓函数，则对应的核函数为：

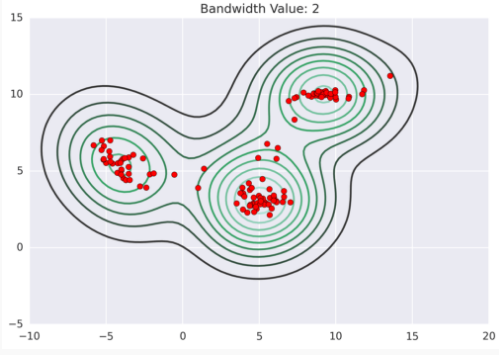


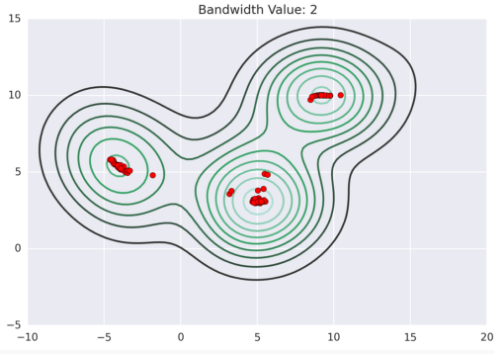
将代入1.4中得：

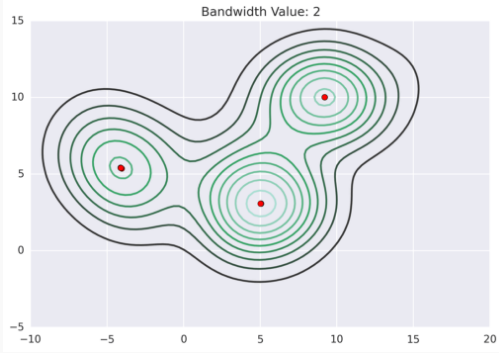


其中最右中括号即为MeanShift向量。记作。所以MeanShift求解的步骤，是逐次迭代执行以下步骤：对于每一个点, （1）计算其MeanShift向量。（2）根据MeanShift向量来平移窗口。循环执行直至收敛。实现结果如下图所示：









本算法通过隐马尔可夫模型对于人员的下个落脚点进行预测。

以下用一个简单的例子来解释隐马尔可夫模型。假设我们有3个盒子，每个盒子里都有红色和白色两种球，这三个盒子里球的数量分别是：

|  |  |  |  |
| --- | --- | --- | --- |
| 盒子序号 | 1 | 2 | 3 |
| 红球数 | 5 | 4 | 7 |
| 白球数 | 5 | 6 | 3 |

按照下面的方法从盒子里抽球，开始的时候，从第一个盒子抽球的概率是0.2，从第二个盒子抽球的概率是0.4，从第三个盒子抽球的概率是0.4。以这个概率抽一次球后，将球放回。然后从当前盒子转移到下一个盒子进行抽球。规则是：如果当前抽球的盒子是第一个盒子，则以0.5的概率仍然留在第一个盒子继续抽球，以0.2的概率去第二个盒子抽球，以0.3的概率去第三个盒子抽球。如果当前抽球的盒子是第二个盒子，则以0.5的概率仍然留在第二个盒子继续抽球，以0.3的概率去第一个盒子抽球，以0.2的概率去第三个盒子抽球。如果当前抽球的盒子是第三个盒子，则以0.5的概率仍然留在第三个盒子继续抽球，以0.2的概率去第一个盒子抽球，以0.3的概率去第二个盒子抽球。如此下去，直到重复三次，得到一个球的颜色的观测序列:



注意在这个过程中，观察者只能看到球的颜色序列，却不能看到球是从哪个盒子里取出的。在此例中有两个随机序列，一个是盒子序列，称为模型的状态序列，一个是球颜色的观测序列称为模型的观测序列。状态序列是隐藏的而观测序列则是显性的。根据例子我们可以定义隐马尔可夫模型的要素：

观察集合

状态集合

其中是可能的隐藏状态数，是所有可能的观察状态数。

初始状态分布为：。

状态转移概率分布矩阵为：



观测状态概率矩阵为:



其中隐马尔可夫的三要素为记作模型。

隐马尔可夫模型能评估观察序列的概率，实际地，即可预测人员下一个落脚点的概率。给定模型以及观测序列可使用前向算法来计算在模型下观测序列的概率。

定义前向概率为：给定隐马尔可夫模型，定义到时刻部分观测序列为，且状态为的概率为前向概率，记作



可以递推求得前向概率及观测序列概率。

前向算法具体步骤为，

输入：隐马尔可夫模型，观测序列

输出：观测序列概率

1. 初始化



1. 递推 对于t=1,2,…,T-1,



1. 终止



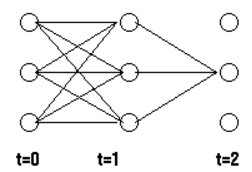


图 递推过程简单示意

* 1. 展现形式

1）人员频繁出入地点分析

聚类，即将物理或抽象对象的集合分成由类似的对象组成的多个类的过程。由聚类所生成的簇是一组数据对象的集合，这些对象与同一个簇中的对象彼此相似，与其他簇中的对象相异。聚类分析又称群分析，它是研究（样品或指标）分类问题的一种统计分析方法。

首先对于一天24小时进行切片，具体分为深夜0时-4时、清晨5时-7时，上午8时-11时，中午12时-13时，下午13时-18时以及晚上19时-23时这六个时间片。对于某一涉毒人员，其在六个时间片下的聚类结果图如下所示。

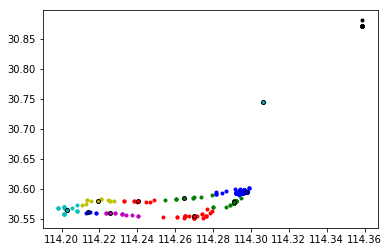


图 该人员深夜0时-4时聚类结果

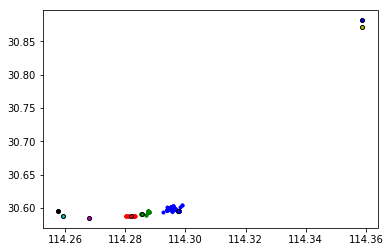


图 该人员清晨5时-7时聚类结果

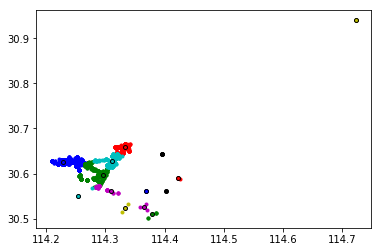


图 该人员上午8时-11时的聚类结果

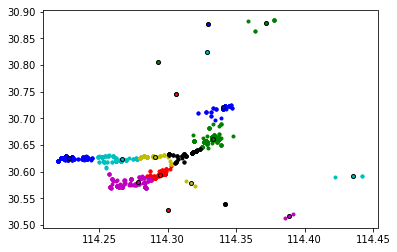


图 该人员中午12时-13时的聚类结果

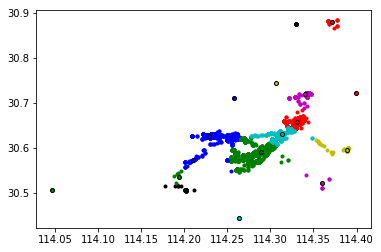


图 该人员下午14时-18时的聚类结果

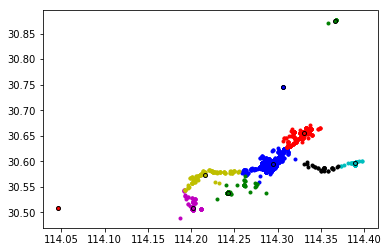


图 该人员晚上19时-23时的聚类结果

同时可由单个人员推广至整个涉毒群体，通过融合群体中每一单个人员的聚类中心点（对应图中空心圆点），再次进行按照时间片进行聚类，可得到群体的按时间划分的聚类结果。

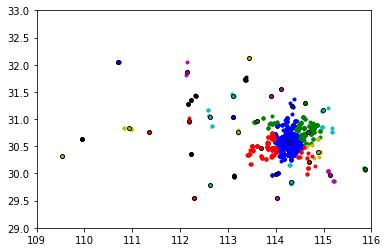


图 涉毒群体所有人员深夜0时-4时的聚类结果

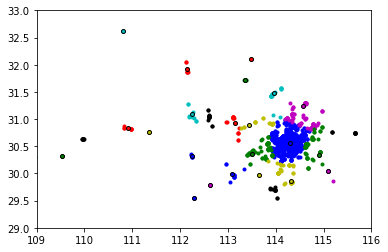


图 涉毒群体所有人员清晨5时-7时的聚类结果

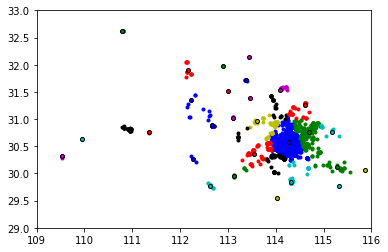


图 涉毒群体所有人员早上8时-11时的聚类结果

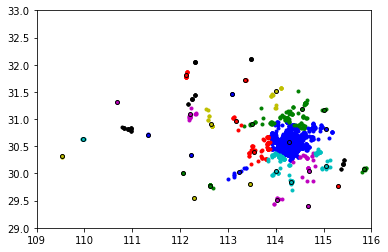


图 涉毒群体所有人员中午12时-13时的聚类结果

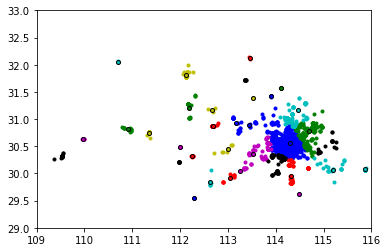


图 涉毒群体所有人员下午14时-18时的聚类结果

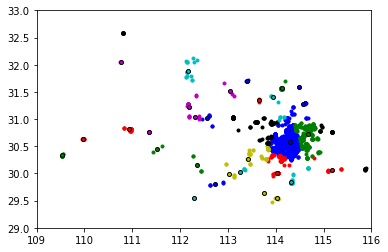


图 涉毒群体所有人员晚上19时-23时的聚类结果

2）人员可能落脚点预测

将人员的聚类中心进行编号，并作为隐马尔可夫模型的观测集合，其中k为该人员聚类中心的个数。而定义模型的状态集合为人员电围数据所划分的区块，其中n为所划分的区块数。通过计算聚类中心到聚类中心、聚类中心到区块的概率，可得状态转移概率分布A和观测概率分布B。再令初始状态分布，即确定了模型。

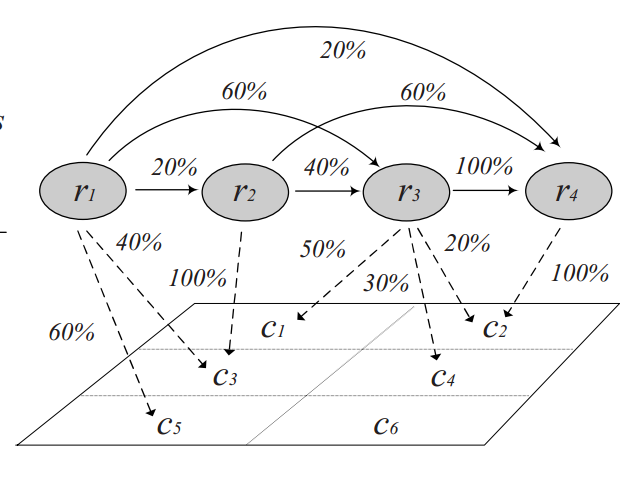


图 一个简单的隐马尔可夫落脚点预测模型

基于该人员的历史轨迹，即可用前向算法预测该人员可能落脚的聚类中心点。例如通过聚类得到该人员有8个聚类中心(编号从0开始)，且人员历史轨迹为[2,3,7,2,4,6,0,1,0]，其中数字代表聚类中心的编号。则可分别计算[2,3,7,2,4,6,0,1,0,0]、[2,3,7,2,4,6,0,1,0,1]、[2,3,7,2,4,6,0,1,0,2]、[2,3,7,2,4,6,0,1,0,3]、[2,3,7,2,4,6,0,1,0,4]、[2,3,7,2,4,6,0,1,0,5]、[2,3,7,2,4,6,0,1,0,6]、[2,3,7,2,4,6,0,1,0,7]、的概率

观察序列为: [2 3 7 2 4 6 0 1 0 0] 概率为: 9.76533203418e-05

观察序列为: [2 3 7 2 4 6 0 1 0 1] 概率为: 3.25511067806e-05

观察序列为: [2 3 7 2 4 6 0 1 0 2] 概率为: 6.5102799487e-05

观察序列为: [2 3 7 2 4 6 0 1 0 3] 概率为: 0.0

观察序列为: [2 3 7 2 4 6 0 1 0 4] 概率为: 0.0

观察序列为: [2 3 7 2 4 6 0 1 0 5] 概率为: 0.0

观察序列为: [2 3 7 2 4 6 0 1 0 6] 概率为: 0.0

观察序列为: [2 3 7 2 4 6 0 1 0 7] 概率为: 0.0

并对他们进行排序，以推测该人员最可能的落脚点。

对于两个实际数据而言，算法的预测准确率也比较理想。

对于序列[0 0 0 0 0 0 1 0 1 1 2 1 1 2 2 2 2 2 1 1 0 0 0 0 3 3 3 3 0 0 0 0 0 0 0 0 0

0 0 2 1 2 1 1 1 1 2 2 1 0 4 0]而言，下一个地点可能性如下：

地点:0概率: 2.20609670218e-25

地点:1概率: 5.28008274519e-27

地点:2概率: 3.5381997777e-27

地点:3概率: 7.24389961909e-27

地点:4概率: 6.25609512558e-27

地点:5 概率: 2.96341348054e-27

地点:6 概率: 1.97560898702e-27

地点:7 概率: 1.31707265802e-27

地点:8 概率: 9.87804493512e-28

地点:9 概率: 6.58536329008e-28

地点:10 概率: 6.58536329008e-28

地点:11 概率: 3.29268164504e-28

地点:12 概率: 3.29268164504e-28

地点:13 概率: 5.44338427339e-29

地点:14 概率: 3.29268164504e-28

预测最大概率结果:地点0

实际结果:地点0

[0 0 0 0 0 0 1 0 1 1 2 1 1 2 2 2 2 2 1 1 0 0 0 0 3 3 3 3 0 0 0 0 0 0 0 0 0

0 0 2 1 2 1 1 1 1 2 2 1 0 4 0 0]

对于序列[0 0 0 0 0 0 1 0 1 1 2 1 1 2 2 2 2 2 1 1 0 0 0 0]

地点:0概率: 5.43654788744e-11

地点:1概率: 1.30118605705e-12

地点:2概率: 8.71928801119e-13

地点:3概率: 1.78513512722e-12

地点:4 概率: 1.54170760987e-12

地点:5 概率: 7.30282552044e-13

地点:6 概率: 4.86855034696e-13

地点:7 概率: 3.24570023131e-13

地点:8 概率: 2.43427517348e-13

地点:9 概率: 1.62285011565e-13

地点:10 概率: 1.62285011565e-13

地点:11 概率: 8.11425057827e-14

地点:12 概率: 8.11425057827e-14

地点:13 概率: 1.3414289248e-14

地点:14 概率: 8.11425057827e-14

概率排序结果：地点0，地点3，地点4，地点1,…

实际预测结果: 地点3

0 0 0 0 0 1 0 1 1 2 1 1 2 2 2 2 2 1 1 0 0 0 0 3

对一个人员200段长为50的历史路径（路径[0:50],路径[1:51],…..路径[200,250]）进行测试,结果top1准确率为173/200=86.5%，top3准确率为184/200=92%。

2.短信或其他短文本

2.1 输入内容

同上一步的输入，或者可重新手动设置手机号及查询时间段

2.2 输出内容

1）输出文本的关键词以及关键词所对应的权重。

2）根据所有短信提取隐含主题，输出主题以及文本所属的主题概率。

2.3 算法或逻辑

1）本模块使用基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图 (DAG) 再使用动态规划查找最大概率路径, 找出基于词频的最大切分组合来对中文文本进行分词。

前缀词典预先储存了根据人民日报语料库提取的词语及其前缀，例如[“北”，“北京”，“北京中医医院”，…..] 根据前缀字典我们对一句句子中所有可能词的情况构成DAG。例如对于句子“去北京大学玩”，将其看作为数组S，则S[0]对应的字为“去”，S[1]对应的字为“北”，S[2]对应的字为“京”….以此类推S[5]对应的字为“玩”。句子对应的DAG为：{0 : [0], 1 : [1, 2, 4], 2 : [2], 3 : [3, 4], 4 : [4], 5 : [5]} 。其中冒号前的数字m代表的是字S[m]，而[]内的数字n1,n2,….则代表从S[m]到S[n1]是一个在字典中出现过的词，从S[m]到S[n2]也是一个在字典中出现过的词。例如1:[1,2,4]就是指以“北”开头，字典中出现过的词有“北”、“北京”、“北京大学”。

因此一句句子可以有多种备选的划分路径，例如划分方案1：去/ 北/ 京/ 大/ 学/ 玩/,划分方案2：去/ 北京/ 大学/ 玩，划分方案3:去/ 北京/ 大/ 学/ 玩，划分方案4: 去/ 北京大学/ 玩，等等划分。对应的图如下图所示：

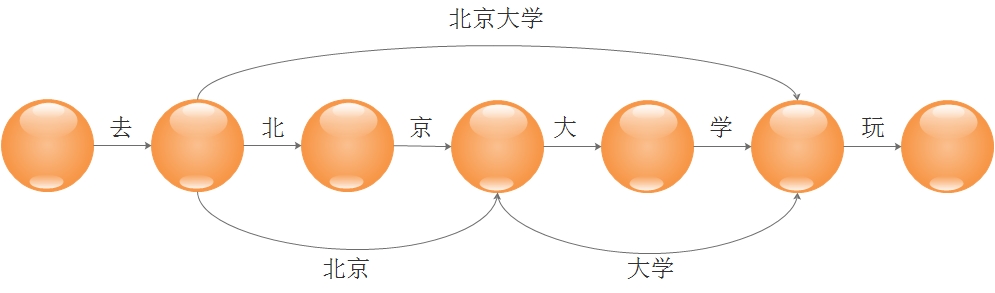


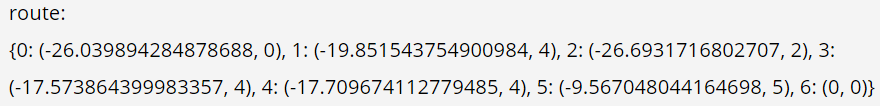
图 “去北京大学玩”对应的DAG图

接着利用动态规划算法对句子从右往左反向计算最大概率，从而查找出最大概率路径。对于句子“去北京大学玩”其过程如下,其中freq()表示词在字典中的频率：

1. 令route[6]=0。
2. 因DAG为4:[4],5:[5],因此对于route[4]和route[5]只有一种计算方式。即route[5] = log(freq(“玩”))-常数+route[6] route[4]=log(freq(“学”))-常数+route[5]。
3. 而route[3]则有多种选择。route[3]=log(freq(“大”))-常数+route[4] 或 route[3]=log(freq(“大学”))-常数+route[5],通过取这两种情况最大值赋给route[3]，并记录最大值对应的组合方式。
4. 同理 route[2]=log(freq(“京”))-常数+route[3]
5. route[1] =max( log(freq(“北”))-常数+route[2] , log(freq(“北京”))-常数+route[3] ,log(freq(“北京大学”))-常数+route[5] )。

f）route[0] = log(freq(“去”))-常数+route[1]

如上从右到左，自底向上计算所得route结果如下图所示：



对应的句子最大概率划分路径为：去/ 北京大学/ 玩/。

2）本模块使用TextRank算法来提取短文本中的关键词。

关键词提取应用的任务是自动提取文本中的关键词集合，使得该集合能最好地描述文本内容。这些关键词可以用于对文本分类，也可以作为给定文档的简要总结。此处我们使用TextRank算法来提取关键词。实验证明基于图的排名模型在关键词提取上表现出色。

TextRank算法对于文本中的每个单词进行排序，并将这些单词添加到文本图的顶点。而两个单词（顶点）之间的连接（边）使用共现关系，由单词之间的距离来衡量: 如果两个顶点对应的词汇单元同时出现在一个单词窗口中（单词窗口通常可包含2-10个单词），则连接这两个顶点。

值得注意的是，可使用语法过滤器对添加到文本图的顶点进行限制，即只选择特定词性的单词。例如可以只考虑将句子中的名词和动词添加到文本图的顶点，且只根据名词和动词之间的共现关系来连接顶点。

TextRank算法完完全全是一种无监督学习算法，它的具体步骤如下：

首先对文本进行分词以及词性标注。

接下来将所有通过语法过滤器的单词添加到文本图的顶点中，并根据共现关系画出无向无权的边，从而完整地构造出文本图。并使用基于图的排序算法计算每一个顶点的分数。基于图的排序算法本质上是在一个图中决定顶点的重要性。基于图形的排名模型实现的基本思想是“投票”或“推荐”。当一个顶点连接到另一个顶点时，可视作它为另一个顶点投票。一个顶点所获票数越高则其重要性也就越高。此外，顶点本身的重要性也决定了投票的重要性。因此一个顶点的分数既取决于它所获票数，又取决于为它投票顶点的重要性。正式地，令图，其中为图的顶点集合，为图的边集合。对于一个给定的顶点，记为指向的顶点集合，为指向的顶点集合。顶点的分数由以下公式定义：



其中d为0-1之间的阻尼因子，其代表从给定顶点跳转到另一个图中随机顶点的概率，d通常设定为0.85。对于图中每一个,设定的初始值为1。

通过对于文本图迭代执行排序算法20-30次直至收敛，可计算得到图中每一个顶点的最终得分。按照分数高低进行排序所得前T个单词即为提取的关键词，其中T的取值一般为句子单词数的1/3。

3）本模块使用LDA主题模型算法对短文本进行主题推测。

LDA主题模型能够以自发无监督的方式有效提取一篇文档中的主题，其本质上是一种三层贝叶斯概率模型，自下而上包含特征词、主题和文档三个层次。在LDA模型中每个特征词都由两次抽取得到：先以一定概率抽取某个主题；再在此主题中以一定概率抽取某个特征词，重复进行以上操作获取足够的特征词以组成一篇文档，这样在一篇文档中每个特征词出现的概率为：

其中是已知的，而 和 是未知的，所以LDA主题模型是通过一个已知参数来推断出两个未知结果的。下面分析一下LDA模型的工作原理，如图所示。

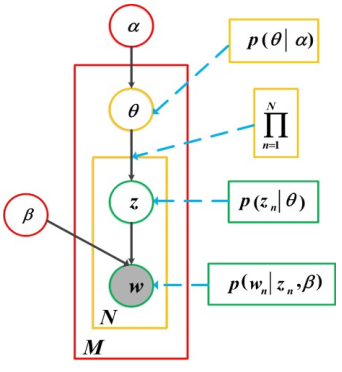


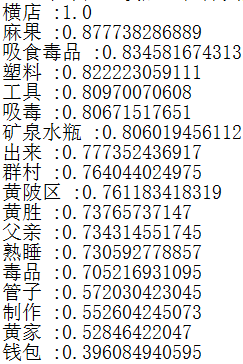
图 LDA模型工作原理

图中自上而下红色表示语料层，黄色表示文档层，绿色表示特征词层，而上文中提到的主题层是LDA要提取的结果。在语料层中M为模型中文档总数；α 为文档中主题分布信息，β为主题中特征词分布信息，利用α和β能够推算出有价值的主题层中的信息。在文档层中N为一篇文档中特征词数量；θ为一篇文档所对应的主题向量，θ中的元素为这篇文档中各个主题的出现概率；表示主题向量θ在参数α下的概率分布。在特征词层中w和z分别为指定的特征词与主题；为主题在指定向量θ下的概率分布；为特征词在指定主题与参数β下的概率分布。由此可知LDA主题模型联合概率为：

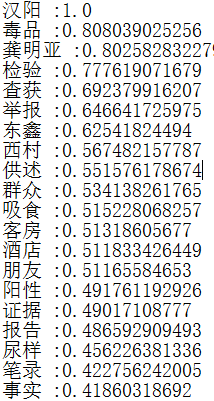
通过分析LDA模型的工作原理可知其中的特征词是这样生成的：抽样生成服从α分布的K维向量，其中K为生成文档所包含的主题数目；从多项式概率分布中抽样生成目标特征词所在的主题，，再从中抽样生成目标特征词，=。

2.4 展现形式

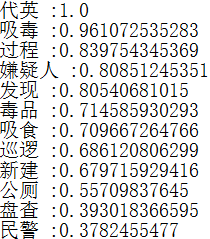
1）对于每条文本提取使用算法自动提取其关键词以及关键词所对应的权重。例如对于案件内容一“2012年1月8日凌晨1时许,黄胜趁其父亲熟睡,将事先藏在钱包里的毒品“麻果”一颗拿出来,在黄陂区横店街建群村黄家湾自己的家中,用自己制作的简易吸毒工具(一个矿泉水瓶子,两根塑料管子),吸食毒品“麻果”一颗。”其对应的关键词与权重如下所示：



又如案件内容二“2010年10月19日晚19时许,龚明亚在汉阳玫瑰园西村一个叫“飞飞”的朋友家里吸食了毒品麻果,2010年10月20日10时许,我所民警通过群众举报在汉阳东鑫酒店一客房将其查获。以上事实有龚明亚本人供述笔录、查获经过、其尿样毒品检验(MAMP)呈阳性检验报告等证据为证。”其对应的关键词与权重如下所示：



又如案件内容三“2009年3月10日10时许,我所民警在巡逻过程中发现吸毒嫌疑人代英在新建街公厕旁.经盘查代英于3月9日晚7时许吸食过毒品.2009年3月10日10时许,我所民警在巡逻过程中发现吸毒嫌疑人代英在新建街公厕旁.经盘查代英于3月9日晚7时许吸食过毒品”其对应的关键词与权重如下所示：



同时根据关键词及权重绘制词云图如下：

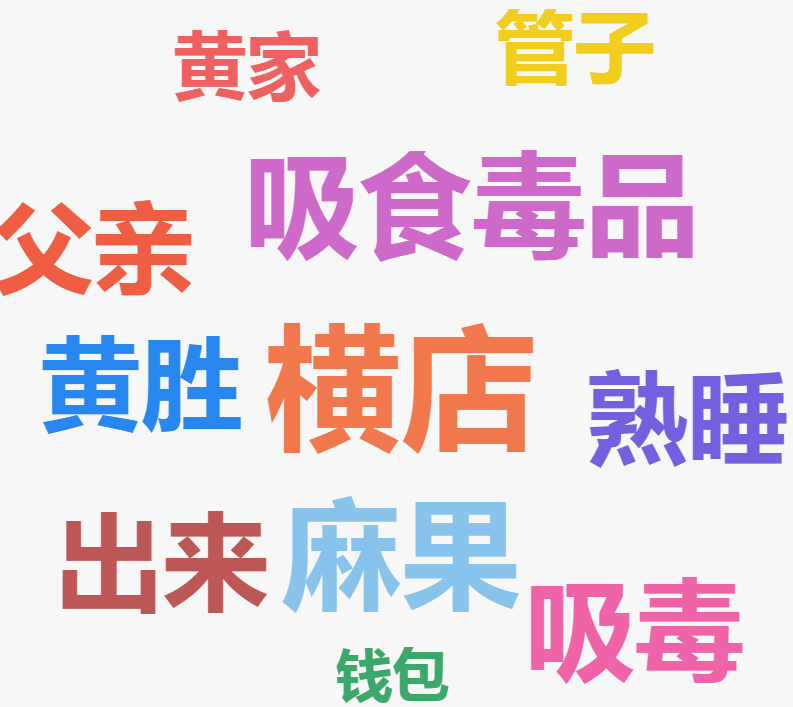


图 案件一关键词词云图，其中权重大的关键词在图中显示也较大



图 案件二关键词词云图

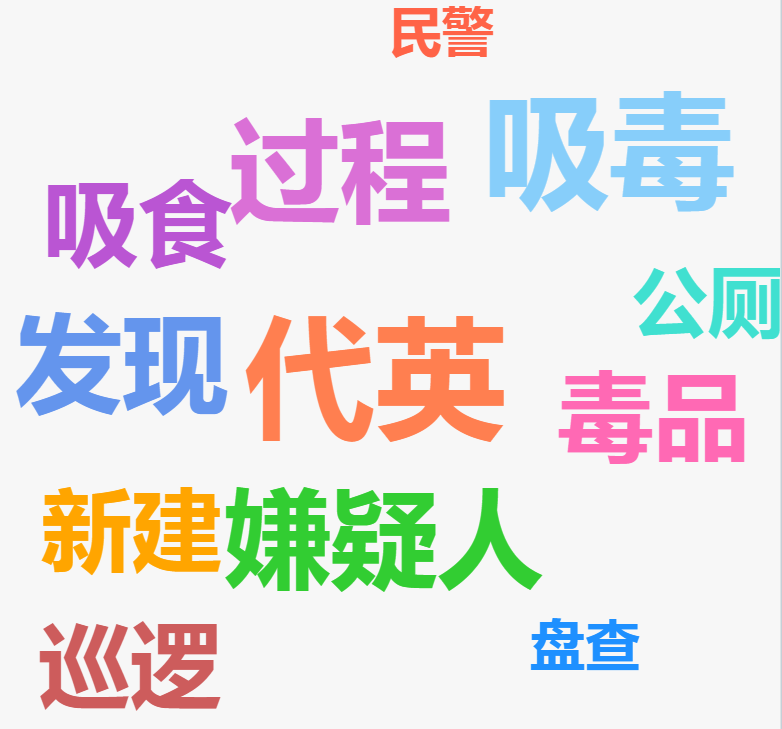


图 案件三关键词词云图

2)对于该人员所有文本使用LDA主题算法提取隐含主题，同时显示每个主题的前30个单词以及某篇文本所属的主题概率。

对于整个吸毒人员的案件描述进行隐含主题提取后所得的每个主题前30个单词如下：

Topic #0: 在 被 吸食毒品 麻果 “ ” 查获 的 我 所 许 吸食 方式 海洛因 内 违法 号 后 毒品 家中 武汉市 注射 青山区 被查获 宾馆 抓获 武昌区 一 晚时许 房间内

Topic #1: ” “ 包房 人 K 岁 粉 KTV 此 杨园 阳逻街 等 余 旅社 当场 为 不讳 鄂 ; 村 三人 的 张华玲 十里铺 东方 一辆 男 音乐会 一辆车 粮道

Topic #2: 在 所 我 的 吸食毒品 0 查获 将 其 月 时 违法 年 日 麻果 号 事实 许 吸食 内 毒品 发现 对 带回 人员 被 海洛因 抓获 审查 供认不讳

Topic #3: 有人 将 的 东西湖区 发现 进行 " 至 涉嫌 纸坊 传唤 调查 大队 检查 现场 了 询问 称 吴家山 到 经 该 后 街 队 男子 武汉市 对 江夏区 公安机关

Topic #4: K 粉 OK 酒吧 卡拉 厅 内 大厦 有人 行政拘留 大道 人 江岸区 在 等 大 戒毒 强制 三角 桃花岛 京汉 建议 抓获 歌厅 查处 洪山区 本 塘 对面 南京路

同时对于每一篇文本，LDA算法计算其属于上述主题的概率。对于案件内容一而言其对应的每个主题概率为：



其含义为，该篇文本属于上述Topic #0的概率为87.186482%，属于上述Topic #1的概率为0.393127%，属于上述Topic #2的概率为11.614796%，属于上述Topic #3的概率为0.404001%，属于上述Topic#4的概率为0.401595%

对于案件内容二而言其对应的每个主题概率为：



对于案件内容三而言其对应的每个主题概率为：



3.关系网

3.1 输入内容

同上一步的输入，或者可重新手动设置手机号及查询时间段

3.2 输出内容

与该人员通联人员的关系图谱

3.3 算法或逻辑

本功能基于图数据库构建已知团体的关系网络。图形数据库是一种非关系型数据库，它应用图形理论存储实体之间的关系信息。最常见例子就是社会网络中人与人之间的关系。关系型数据库用于存储“关系型”数据的效果并不好，其查询复杂、缓慢、超出预期，而图形数据库的独特设计恰恰弥补了这个缺陷。

本功能使用pagerank算法挖掘团体中关键节点。Pagerank是是Sergey Brin与Larry Page于1998年在WWW7会议上提出来的，用来解决链接分析中网页排名的问题。Pagerank原理则是完全通用的，适用于任何领域中的网络或者图。其核心思想即是1.某个节点被链接的次数越多，它的重要性越高。2.被重要性高的节点所链接的节点，重要性也高。PageRank算法总的来说就是预先给图中每个节点一个值，其中值物理意义上为一个节点被访问概率，所以一般是，其中为节点总数。一个节点的值计算如下：



其中是所有对有出度的节点。是节点的出度，为节点总数。根据上面的公式，我们可以计算每个节点的值，通过不断迭代趋于平稳的，所得结果即为最终结果。

3.4 展现形式

本功能基于人员主被叫通联频次建立关系图谱。

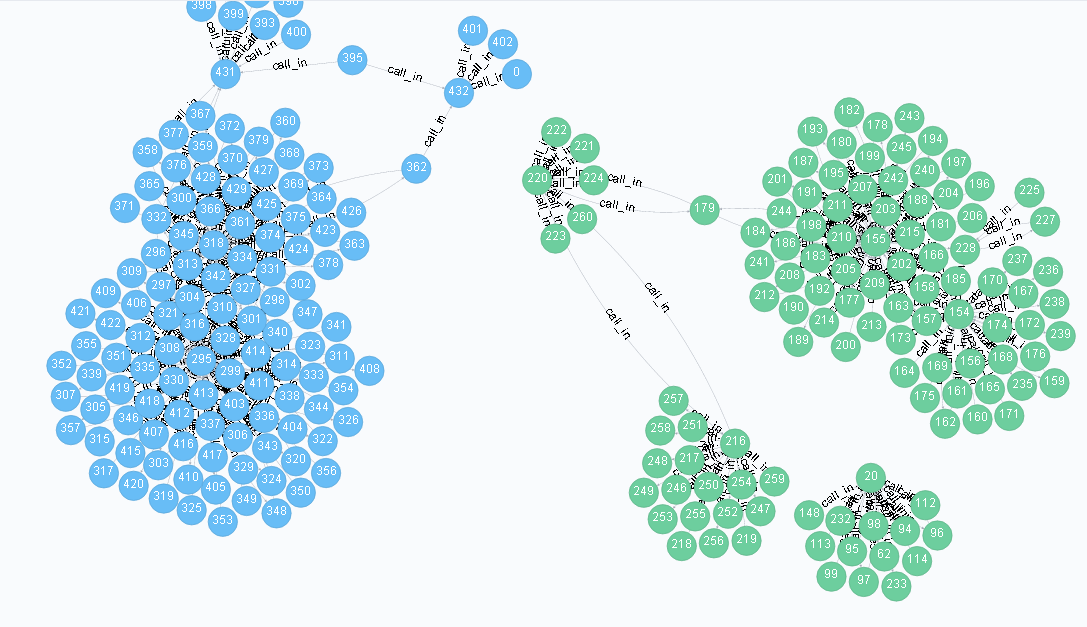


图 一个双向的关系图谱简单示意图

本功能基于Pagerank算法计算每个团体中人员的重要性。其结果如下所示：

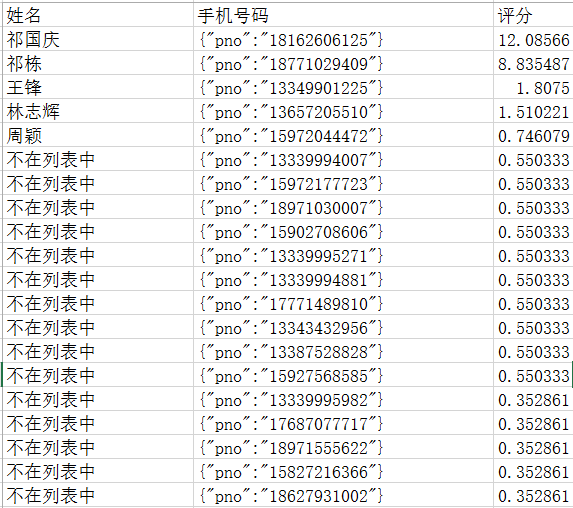
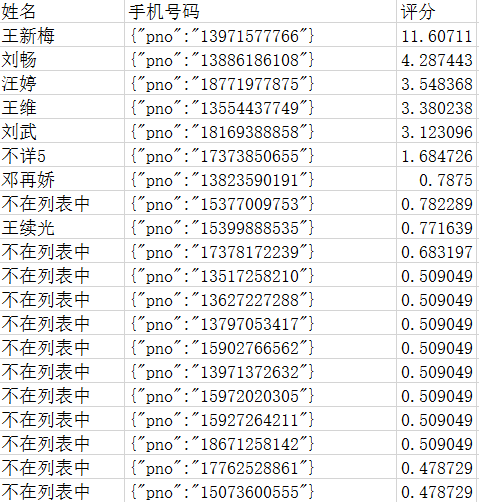


图 两个团体各自人员Pagerank重要性列表

4.现金流

4.1 输入内容

同上一步的输入，或者可重新手动设置手机号及查询时间段

4.2 输出内容

该人员在该时段内大额资金（可设置金额）往来记录

4.3 算法或逻辑

通过大额金额设置来过滤该人员该段时间的资金往来，忽略小额零碎资金往来记录，而着重关注大额的资金往来。

4.4 展现形式

以表格及双向柱图的形式展现该人员该时间段大额资金（可设置金额）往来记录

|  |  |
| --- | --- |
| 日期 | 收支 |
| 2016-12-31 | -40000 |
| 2017-1-2 | 50000 |
| 2017-3-5 | 70000 |
| 2017-3-8 | -60000 |
| 2017-3-10 | 20000 |
| 2017-3-12 | -30000 |

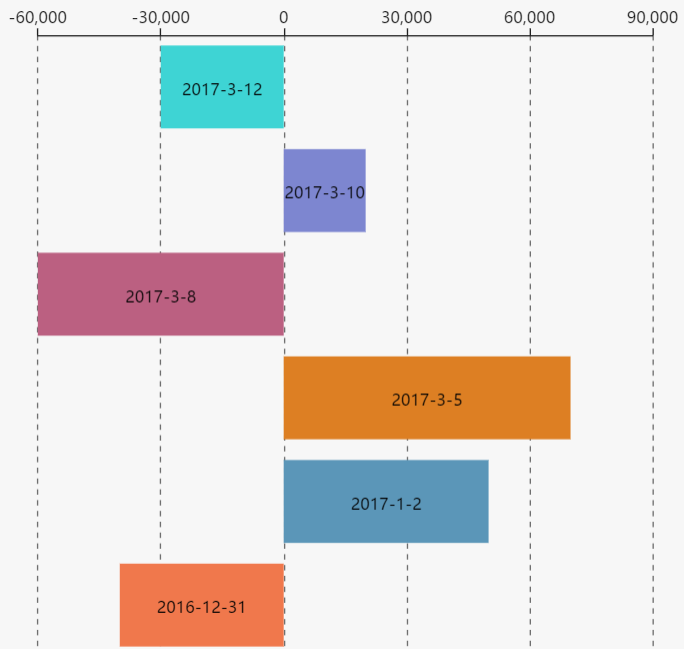


图 某人员某时间段内的资金往来记录

5.出行

5.1 输入内容

同上一步的输入，或者可重新手动设置手机号及查询时间段

5.2 输出内容

该人员该时段出行方式环状图

5.3 算法或逻辑

基于该人员的各种出行方式，对其所有记录分类别统计频次，并绘制环状图

5.4 展现形式

主要展现形式为该人员最常用路线前十名环状图。

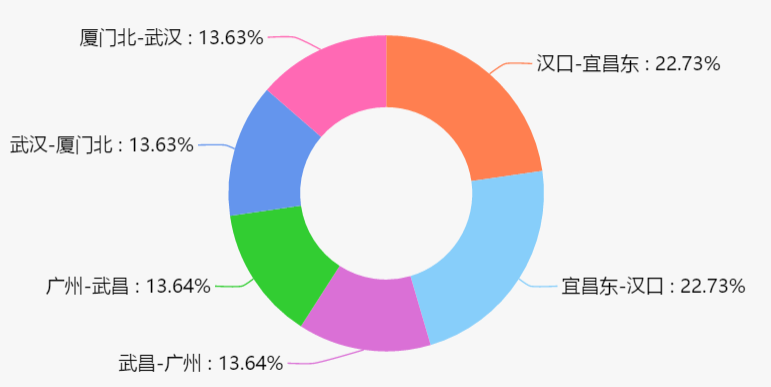


图 某人员火车常用路线前6名饼状图

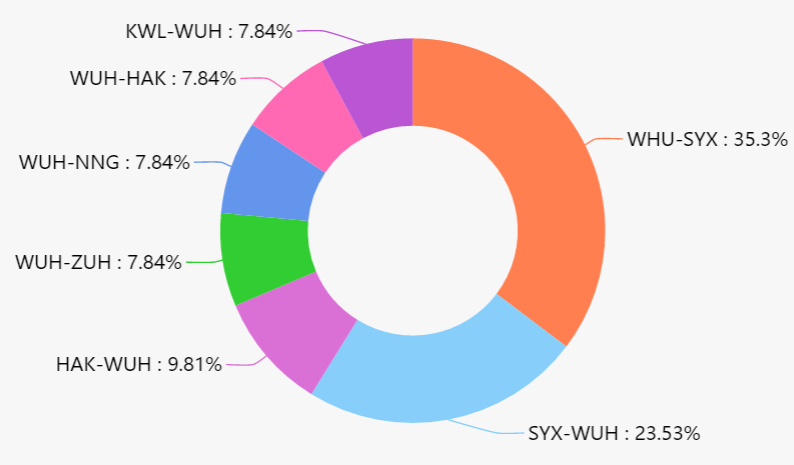


图 某人员民航常用路线前7名饼状图

6.住宿

6.1 输入内容

同上一步的输入，或者可重新手动设置手机号及查询时间段

6.2 输出内容

该人员该时段住宿情况环状图

6.3 算法或逻辑

基于该人员的各种住宿情况，对其所有记录分类别统计频次，并绘制环状图

6.4 展现形式

主要展现形式为该人员最常住酒店名称、所住酒店所属行业、住酒店天数前十名的表格及环状图。

|  |  |
| --- | --- |
| 酒店名称 | 住店次数 |
| 江汉区武汉嘉鸿白金酒店有限公司 | 11 |
| 武汉汉正瑞鑫酒店管理有限公司 | 5 |
| 湖北威玖酒店管理有限公司 | 2 |
| 武汉芭堤雅酒店管理有限公司 | 2 |
| 武昌区武汉胜家酒店管理有限公司 | 2 |
| 武汉市梨铧出租（精途） | 1 |
| 空军武汉蓝天招待所 | 1 |
| 武汉时光道酒店管理有限公司 | 1 |
| 武昌区巴黎之吻情侣酒店 | 1 |
| 武汉艳阳天商贸发展有限公司后湖店 | 1 |

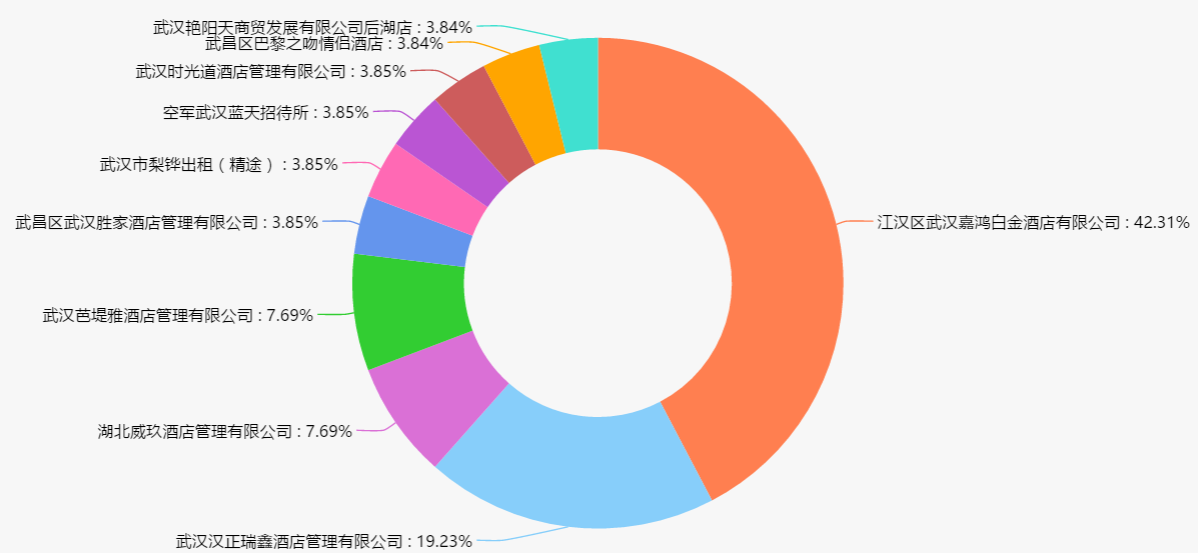


图 该人员最常住酒店名称前十名

|  |  |
| --- | --- |
| 酒店类别 | 频次 |
| 旅馆业 | 35 |
| 旅店式出租屋 | 2 |
| 新装未分类 | 1 |

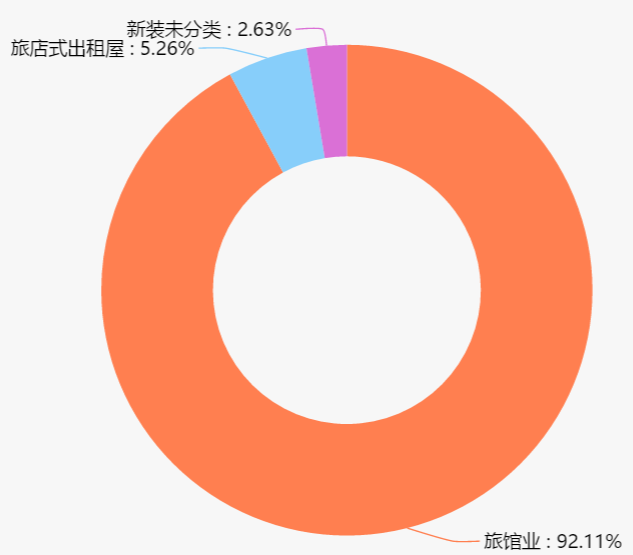


图 该人员最常住酒店类别前十名

|  |  |
| --- | --- |
| 住宿天数 | 频次 |
| 0天 | 20 |
| 3天 | 4 |
| 4天 | 2 |
| 1天 | 1 |
| 6天 | 1 |
| 平均天数：0.964286天 | |

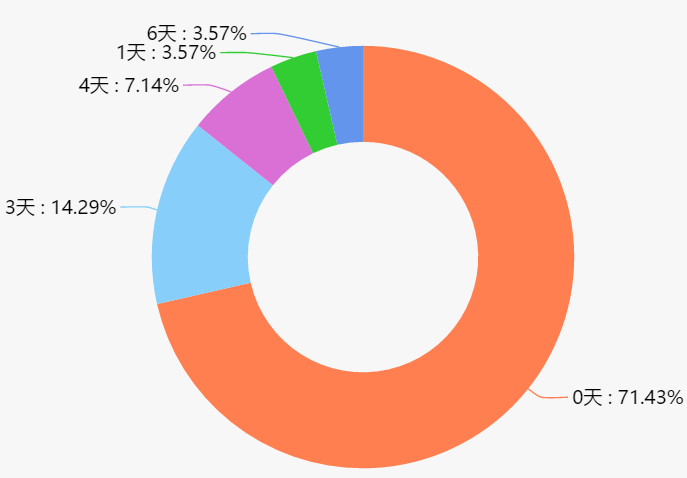


图 该人员住酒店天数前十名

1. 多粒度画像界面

用户画像界面整体效果图如下图所示：

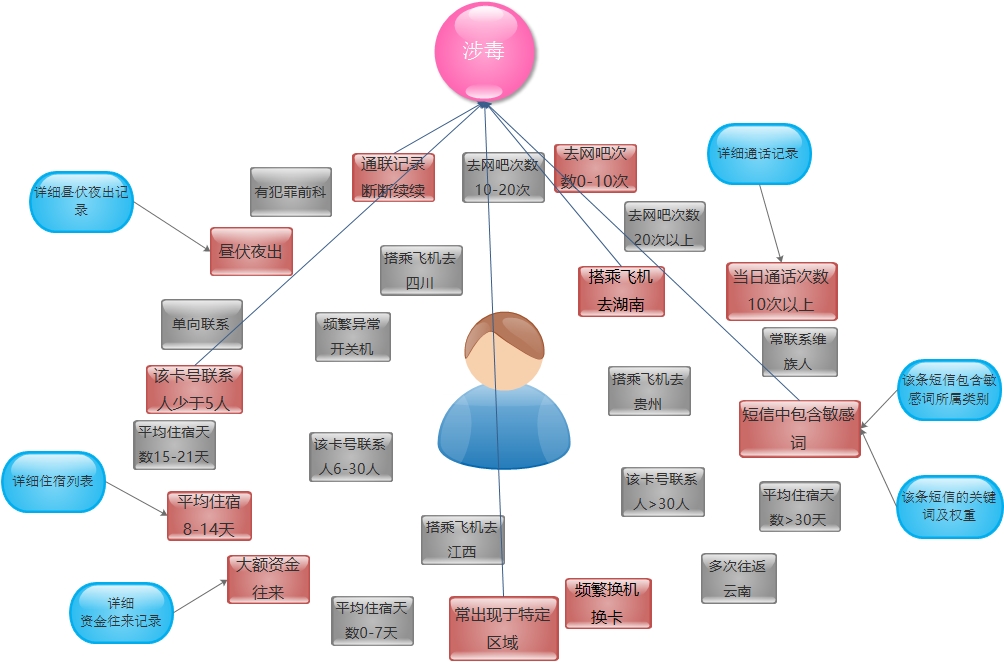
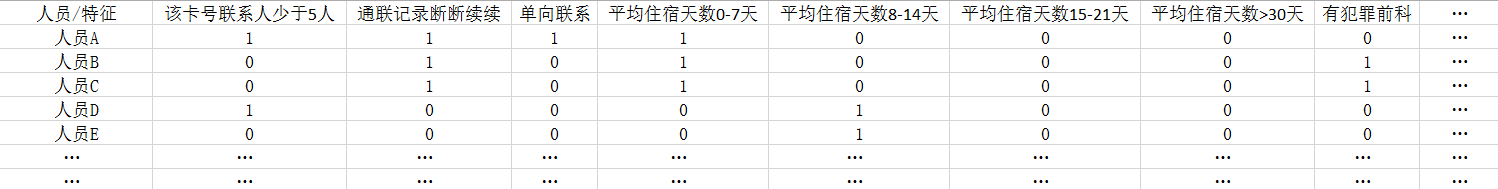


图 用户画像界面概念设计图

其中图中蓝色椭圆为最底层的内容，即为原始数据经过统计或者切片或者某些特定算法预处理后而得的加工数据。

红色矩形与灰色矩形都是与各个专题相关的标签，其本质是蓝色椭圆中数据的语义抽象。是中间层内容。其包括但不限于单向联系、频繁换机换卡、频繁异常开关机、昼伏夜出、常联系维族人、当日通话次数10次以上、短信中包含特定关键词、出现于特定区域、平均住宿天数0-7天、平均住宿天数8-14天、平均住宿天数15-21天、平均住宿天数>30天、有犯罪前科、多次往返云南、大额资金往来、该卡号联系人少于5人、通联记录断断续续等等基于人工经验或者基于机器学习分类效果修正得到的标签。红色矩阵指的是该人员在所查询的时间段内，行为符合该标签的逻辑，因此被点亮为红色。而灰色矩阵则是指该人员行为不符合该标签的逻辑。

而粉红色圆形则为最高层功能——研判。粉红色圆形根据与其研判主题相关的红灰色矩阵对于该人员进行编码、建模得到特征向量。基于红灰矩阵构建特征向量的示例如下：



即对于红色矩阵该标签下的值为1，而对于灰色矩阵该标签下的值为0。最后通过学习分类器得出所研判专题的研判结果。

实际对于吸毒人员的模型来说，定义的特征维度为：1.是否火车去过云南 2.是否火车去过湖南贵州 3.是否火车去过四川柳州江西 4.是否有火车出行记录 5.是否飞机去过云南 6.是否飞机去过湖南贵州 7.是否飞机去过四川柳州江西 8.是否有飞机出行记录 9.是否去过涉毒人员常去旅馆库 10.是否去过涉毒人员聚类点附近11.是否晚上通联记录数（19点-第二天7点）大于早上通联记录数，12.在案件中心是否有>=2件案底，构建12维的特征向量。对这12维特征向量使用SVM算法分类得准确率为90%。

对于涉黑人员模型来说，我们对于案件库中根据案件描述中的关键词如“装修材料”、“工地”、“邀约”以及根据“三人三起”即一起案件最少有三人共同参与，同时此三人还参与最少两起其他案件建立了涉黑子类型数据库，如“敲诈勒索”人员库、“寻衅滋事”人员库、“聚众赌博”、“组织卖淫”等五类人员库。

为了对于这五类人员进行建模，我们需从hive平台读取数据。具体步骤为：1.根据人群类型生成相应的sql语句 2.远程连接hive平台，输入生成的sql语句 3.由于电围数据体量太大，因此需根据表的时间分区来筛选最近的时间段，并新建表 4.进行联表查询后得到我们所需的表，同时更改表的压缩格式 5.从hdfs系统中找出联表查询后表的位置，随后使用getmerge命令将该表有关所有内容合并为一个txt文件 6.再将该txt文件下载至本地