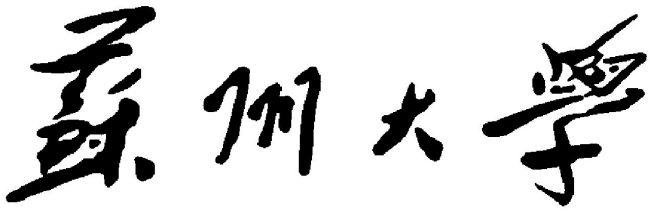

SOOCHOW UNIVERSITY

硕士专业学位论文

论文题目 基于Python的户外通讯设备连接关系的挖掘研究

研究生姓名 张正阳

指导教师姓名 唐煜

专业名称 应用统计学

研究方向 应用统计

论文提交日期 2015年4月

苏州大学学位论文独创性声明

本人郑重声明：所提交的学位论文是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含其他个人或集体已经发表或撰写过的研究成果，也不含为获得苏州大学或其它教育机构的学位证书而使用过的材料。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本人承担本声明的法律责任。

论文作者签名：张正勇　　日　期：2015.4.20

苏州大学学位论文使用授权声明

本人完全了解苏州大学关于收集、保存和使用学位论文的规定，即：学位论文著作权归属苏州大学。本学位论文电子文档的内容和纸质论文的内容相一致。苏州大学有权向国家图书馆、中国社科院文献信息情报中心、中国科学技术信息研究所（含万方数据电子出版社）、中国学术期刊（光盘版）电子杂志社送交本学位论文的复印件和电子文档，允许论文被查阅和借阅，可以采用影印、缩印或其他复制手段保存和汇编学位论文，可以将学位论文的全部或部分内容编入有关数据库进行检索。

涉密论文□

本学位论文属　　在_____年___月解密后适用本规定。

非涉密论文□

论文作者签名：张正翔　日　期：2015.4.20

导 师 签 名：[signature]　日　期：2015.4.20

基于 Python 的户外通讯设备连接关系的挖掘研究

摘 要

随着通讯行业的不断发展，个人和企业对通讯业务需求的不断提升，各运营商都在加大信息基础设施的建设力度。当一块区域设施建设完毕并投入使用之后，各运营商要对该区域内设备连接关系进行管理，并设定一些指标值对该区域进行监控，一旦超过阈值，管理人员要能够迅速找到该区域内的具体地址并作出反应。本文以苏州市某通讯运营商在苏州市农村区域设备连接情况为测试数据，通过 Python 语言平台，设计一个能够对户外通讯设备连接关系进行数据挖掘的系统。这个系统，从设备连接关系归纳和设备连接关系查询两方面，为深入了解某一区域设备连接状况及通过相关指标进行连接节点筛选提供技术工具。测试结果表明，本系统能够准确、有效的归纳出设备连接关系，并通过查询功能合理的展示这些连接关系，以及筛选出与相关指标有关的连接节点位置编号。

关键词： Python 语言、数据挖掘、通讯设备、连接关系

作 者：张正阳

指导教师：唐煜

Python-based mining research of connected relations between outdoor communications facilities

Abstract

With the development of the communications industry, individuals and businesses' demand for communications services are growing, the operators are increasing their efforts in building the information infrastructure. Once facility construction is completed and put into use in an area, the operators have to manage the connected relations of these facilities within the region,and set the value of some indicators to monitor the region, once the indicators goes up to the threshold value, managers should be able to quickly find the region and respond to the specific address. In this paper, the writer try to design a Python-based data mining system for connected relations of outdoor communications facilities,the data of facilities connected relations in rural area in Suzhou of a Suzhou telecommunications operator will be used as test data.This system,provide a technical tool for in-depth understanding of communications facilities connected relations status in a particular area. Test results show that the system can accurately and effectively induce the relationship between the facilities, and shows the relationship between these connections through rational inquiry, as well as filter out some location IDs of connecting nodes which are related to relevant indicators.

Key Words：Python language,Data mining,communications facilities, connected relations

Written by: Zhang Zhengyang

Supervised by: Tang Yu

目　录

第一章 引 言

1.1 课题背景

随着中国经济的快速发展，人民生活水平日益提高，通讯行业技术日渐成熟，人们对通信网络使用的需求程度几乎呈“指数级”增长。然而，与日益增长的需求相对立的是我国信息基础设施之落后，其程度我们自己都很难想象。根据国际电信联盟（ITU）的评估，我国通信带宽水平在世界范围内的排名在 80 位以后，远远落后于日本、韩国和欧美国家，通信带宽水平是通信网络质量的重要衡量指标。而中国 2014 年的 GDP 已经突破 10 万亿美元大关，位列世界第二，显然我国的信息基础设施建设与经济发展水平严重脱节。

面对这样的情况，国家层面和社会各界对加强信息基础建设的呼声日益高涨。李克强总理在全国政协十二届三次会议的经济、农业界联组讨论时就提到：“信息基础设施建设是重要的公共服务，应当加大建设力度。”此外，企业家丁磊也在 2015 年一季度经济形势座谈会上发言说，中国经济走到今天，需要大量的拥有扎实专业技能的蓝领工人，应该利用移动互联网平台，把专业技能的知识讲授、在线培训传递给需要的人。要实现这样的目标就必须拥有高质量的信息基础设施建设体系。

为拥有高质量的信息基础设施建设体系，首先要加快信息基础设施建设，扩大设施覆盖面，让通讯技术的发展成果惠及更多百姓和企业。其次要提升通信网络质量，这就要求优化通信网络连接状况，其中最根本的一点就是对各户外通讯设备连接状况进行掌握。

1.2 国内外现状分析

通讯连接设备。通讯连接设备为通讯主设备之间、通讯主设备与线缆之间、线缆之间的数据通讯提供传输媒介，并实现相关设备的保护功能。通讯连接设备是通讯网络系统的基础，为数据传输提供可靠的环境。

通讯连接设备的数量日益增长。以近年来移动互联网通讯连接网络的增长来看，随着全球 4G 业务的飞速发展，通讯设备的连接网络也在以惊人的速度扩

张。以美国为例，美国的 4G 网络的推出已经超过 3 个年头，在这三年里，美国的 4G 网络发展可以说已经达到巅峰，四大运营商的 4G 信号网络已经覆盖了美国人口的 97%。再看国内，就三大运营商之一的中国移动来看，4G 用户已经超过 650 万，网络覆盖达 300 个城市，投入使用的基站超过 32 万个。

通讯网络的迅速扩张，必然对通讯设备连接关系的管理提出更高要求。据了解，国内的三家运营商都有基于各自业务平台的设备连接管理系统，但如何提高设备连接管理的精度和效率，一直是一个研究课题。

1.3 本文工作概述

本研究拟以苏州市某通讯运营商在苏州市农村区域设备连接情况为测试数据，通过 Python 语言平台，设计一个能够对户外通讯设备连接关系进行有效挖掘的系统。通过这个系统，从设备连接关系归纳和设备连接关系查询两方面，为深入了解某一区域设备连接状况及通过相关指标进行连接节点筛选提供技术工具。

本文结构如下：第二章介绍与本文相关的 Python 语言知识和数据挖掘知识；第三章介绍了基于 Python 语言设计的设备连接关系挖掘系统的设计思路和实现办法；第四章以苏州市某通讯运营商在苏州市农村区域设备连接情况作为测试数据，对系统进行测试；第五章对本文的工作作出总结。

第二章 基于 Python 的数据挖掘相关技术

2.1 Python 语言

2.1.1 Python 语言概述

Python 语言简单易学，功能强大，数据结构高级、有效。Python 语言更是一个面向对象的解释性动态编程语言，包含大量的函数库，用于完成各种高层任务。Python 可以运行在 Windows，Linux 等多种操作系统上，是许多领域的理想脚本语言。

选择 Python 进行数据挖掘基于其一下特性：

语言简练。使用 Python 编写的代码通常比其他语言如 C/C++,Java 等更简短，这意味着开发人员可以将更多的精力放在数据挖掘的算法上，而不是繁琐的书写语法上。

易于阅读。Python 有严格的缩进规定，在书写 Python 代码时严格遵守缩进规定会使得代码非常容易阅读，有经验的程序员可以很快弄清各部分层级，理解其中代码含义。

易于扩展。Python 语言附带许多标准库函数，包括数学函数、XML 解析以及 HTML 获取分析等等，另外还可以加载大量大三方函数以供使用，开发人员可以根据需要选择不同的扩展库函数。

可移植性。Python 是由 C 开发的，任何带有 ANSI C 编译器的平台上都可以运行 Python 的程序。

2.1.2 系统开发环境介绍

本系统开发使用的 Python 代码编辑器为 Sublime，具体配置如下表：

表 2.1 系统开发环境配置

名称	版本号
操作系统版本	64 位 Windows 8.1
Sublime	Text 2
Python 版本	2.7.9

2.2 数据挖掘

随着信息技术的发展，商业企业、科研机构和政府部门在过去若干年的时间里都积累了海量的数据，这些数据存储地点不同，存储结构多样，如何从这些数据中发现有用的信息就成了一个极具挑战的问题。当前多数的数据库系统只是可以对已有的数据进行增、删、改、查等简单操作，通过这些操作人们仅可以获得数据的一些简单的、表面的信息，对于隐藏在数据背后的信息，如关系、特征和趋势就不能通过传统的方法获得，而这些信息对于现今的工作和研究有着巨大价值。

数据挖掘这一概念的提出，深化了人们的对数据的理解程度，为认识数据的真正价值，发现数据蕴藏的信息提供了强有力的工具。数据挖掘（Data Mining），普遍认为是由 W.J.Frawley 和 P.Piatetsky.Shapiro 等人提出的：数据挖掘就是从大兴数据库的数据中提取人们感兴趣的知识，这些知识是隐含的，事先位置的潜在有用的信息，提取的知识表示为概念（Concept）、规则（Rules）、规律（Regularities）、模式（Patterns）等形式。这样的定义把数据挖掘的对象定义为数据库，而在实际工作中数据挖掘的对象绝不仅仅是数据库，更可以是文件系统甚至是任何组织在一起的数据集。

2.2.1 数据挖掘的任务

在实际工作和研究中，我们把数据挖掘的任务分成两大类：预测任务和描述性任务。预测性任务的目标是根据其他属性的值来预测特定属性的值；描述性任务的目标是概括出数据中潜在的关系模式，这类任务通常需要后期的验证和结果的解释。利用数据挖掘技术可以获得决策支持所需的多种知识以满足用

户的期望和实际需要。

2.2.2 常用数据挖掘技术

（1）聚类分析：聚类分析方法是数据挖掘中依据数据集间关联的量度标准将其自动分成几个簇，使得同一个簇内的数据点之间尽可能相似，不同簇的数据点之间尽可能相异。常用的聚类分析方法有快速聚类法（K-means）、两步法等。

（2）分类分析：分类是数据挖掘中的一项重要数据分析方法，在实际工作和研究中被广泛使用。分类的目的是学会一个分类函数或分类模型，我们也可以把它们统称为分类器，分类器能把数据集中的数据点映射到某个特定的类上。分类模式往往表现出来施一棵分类树，根据数据的值从根部开始搜索，沿着数据满足的分支往下走，最终走到树叶就决定了类别。常用的分类算法有①ID3 和 C4.5 判定书归纳的贪心算法；②朴素贝叶斯分类算法；③后向传播分类算法；④基于遗传算法的分类算法；⑤模糊集分类算法等等。

（3）关联规则挖掘：关联规则是大量数据中项集与项集之间有趣的关联或相关联系，关联规则挖掘就是在大量数据中发现这种有趣的联系。许多企业和公司就会用到这种方法帮助决策，例如商家利用大量的客户购物记录挖掘出商品潜在的关联性，从而推出一些捆绑销售策划方案。

（4）回归分析预测：回归分析方法是用当前数据对未来进行预测。在最简单的情况下，回归分析方法可以使用线性回归的技术。但是在很多实际问题上，数据的特点并不一定呈现出线性的特点，不能简单的用线性回归来解决。更复杂的方法包括逻辑回归、决策树算法、神经网络等，这些算法往往既可以用来分类又可用来做回归分析。

（5）时间序列分析：在时间维度上对未来进行预测。时序分析和上述预测分析不同之处在于：时序分析是以时间作为预测和目标变量的载体，即预测和目标变量本身的变化依赖于时间的变化，因而都是时间的函数：而预测分析仍然是一个截面数据分析，虽然考虑了时间因素，但是最终它还是取某段时间内预测变量或目标变量发生的值的度量，如均值、总和等，而不关注在该段时间内的变量值的度量的变化情况。

第三章 基于 Python 的通讯设备连接关系挖掘系统的设计与介绍

3.1 通讯设备连接关系挖掘功能设计

通讯设备连接关系挖掘系统是一种数据挖掘系统，既然从事数据挖掘，就要求既要能梳理出通讯设备连接中父子节点的连接状态，还要能提供对实际工作有价值的数据。因此通讯设备连接关系挖掘系统的结构设计主要分成两个部分，一部分是通讯设备连接关系挖掘功能，另一部分是通讯设备连接关系查询功能。

3.1.1 设备连接关系挖掘功能的目标

通讯设备的连接主要有分种类、分层级的特点，虽然每一家运营商的设备连接方式不尽相同，但主要连接结构是类似的。在某一区域内，设备依照上下层关系建设完成后就会形成一个设备网络，运营商会给这个设备网络中的每个连接节点冠以位置编号，最终会形成一份详细的数据，包括每个子节点的位置编号，上层父节点的位置编号，以及该子节点对应的设备数，用户数和具体的地址信息。然而对于未参与现场施工的后台工作人员来说，这份数据本身并不能良好的展示各节点的连接关系，可以通过下面一张图来展示：

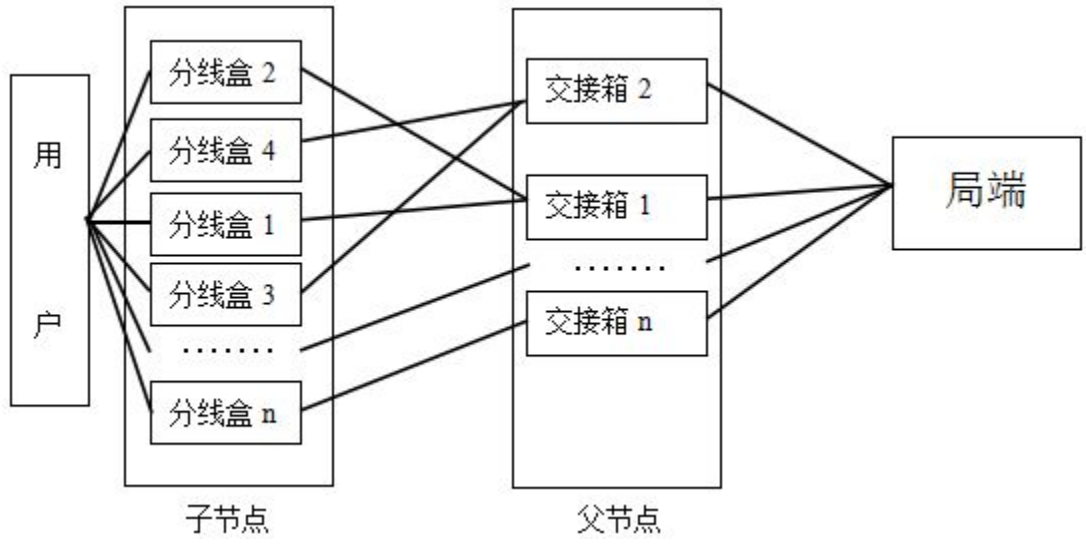

图 3.1 原始数据对应设备连接关系概念图

基于这样一份原始数据，我们试图挖掘各子节点与上层父节点的连接关系，将对应同一父节点的所有子节点都归纳在一起，在归纳的同时依然要保留每个子节点上的信息。最终形成的连接关系如下图所示：

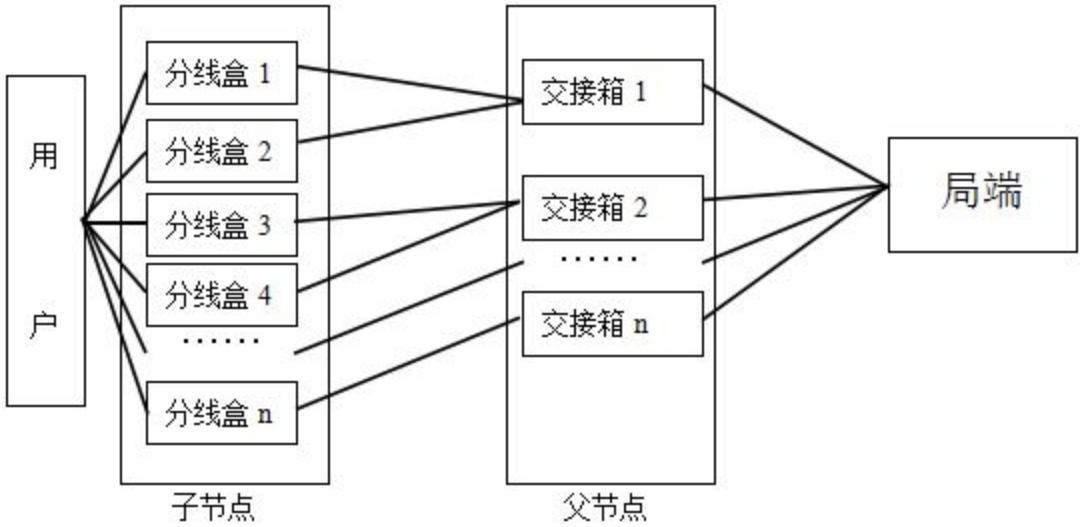

图 3.2 数据挖掘后对应设备连接关系概念图

3.1.2 设备连接关系挖掘功能的原理

基于 Python 软件，根据每一个子节点上包含的信息量，我们首先定义 5 个数组，父节点 parent []，子节点 child[]，子节点地址信息 child_address[]，用户数 user[]，设备数 facility[]。接着打开数据文件，将数据按行读取到新的数组 all_file[]中。利用 for 语句遍历 all_file[]的每一条信息，利用 if 语句完成对子节点从属关系的判断和子节点所包含的信息的归纳。需要注意的是，由于父节点数组拥有更高层级，直接使用数组即可，而其余 4 个数组由于层级低，必须使用数组相互嵌套的方式，才能完成从属关系的对应。也就是说实际上数组 parent[]中的每个元素是一个位置编号，而数组 child[]，user[]，facility[]和 child_address[]中的每一个元素是一个子数组。

具体来说，当 if 语句判定出一条信息的父节点位置编号不在数组 parent[]中时，首先，将此父节点位置编号作为新元素新增到数组 parent[]中，接着，将这条信息中所包含的子节点位置编号，用户数，设备数，地址信息同样的作为新元素分别加入到数组 child[]，user[]，facility[]和 child_address[]中新子数组的第一个位置中。

当 if 语句判定出一条信息的父节点位置编号已经在数组 parent[]中时，只要取出这个父节点位置编号在数组 parent[]中的索引号，并将这条信息中所包含的子节点位置编号，用户数，设备数，地址信息按照索引号新增到数组 child[]，user[]，facility[]和 child_address[]对应的子数组中。

3.2 通讯设备连接关系查询功能设计

在已经归纳好所有父子节点连接关系的基础上，我们可以通过查询的方式，展示每一个父节点上连接着的所有子节点，我们称作查询模式 a；同时还可以根据实际需要，设定一些指标，在输出节点连接关系的时候自动计算，利用自动计算出的指标来衡量该区域设备使用情况，我们称作查询模式 b。

此外，该查询功能还是实现查询区域切换。当一个区域的情况已经了解完毕，想要了解其他区域时，我们要退出当前查询模式，选择查询模式 c 进行区域切换。最后，当所有查询进行完毕，我们可以选择查询模式退出。

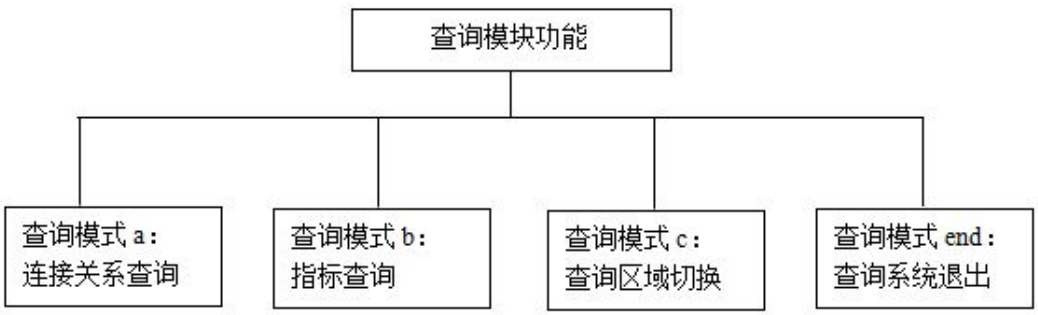

图 3.3 查询模块功能

3.2.1 查询模式 a：通讯设备连接关系查询

在查询模式 a 中，我们可以根据父节点的位置编号，查询其下属的所有子节点。其实现原理是通过界面的交互，取得输入的父节点位置编号，这个父节点的位置编号一定是数组 parent[]中的元素之一，因此我们可以用它取得该元素的索引号 index，这个索引号同时又是该父节点包含的所有子节点在其相应的子节点 child[]，子节点地址信息 child_address[]，用户数 user[]，设备数 facility[]数组中的索引号。这样就可以输出所有连接到这个父节点上子节点的各项信息。

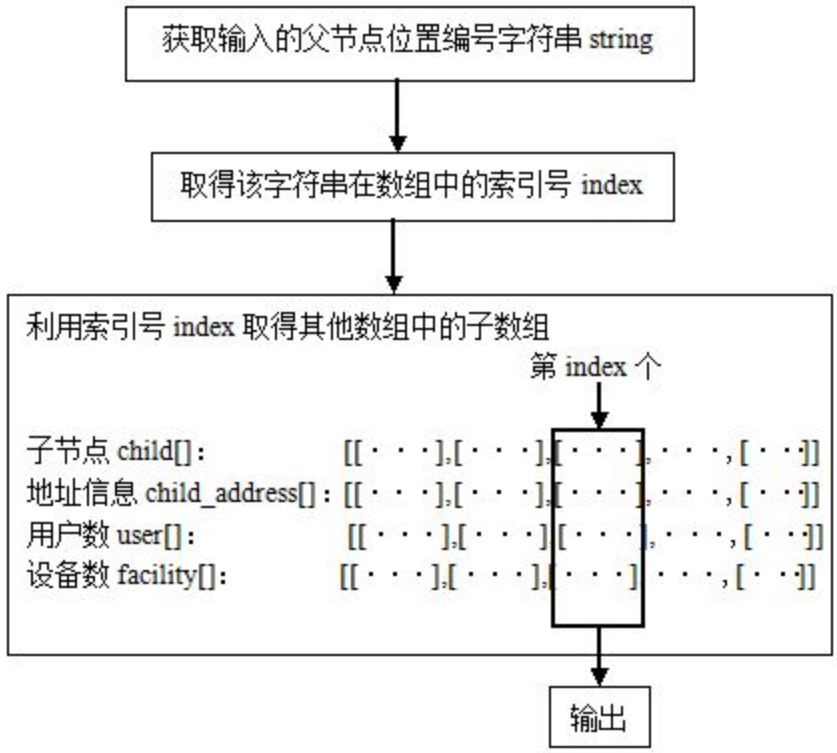

图 3.4 父节点查询原理示意图

此外，该查询模式还实现了对所有连接到同一个父节点上的设备数和用户数求和的功能。如上文所述，在取得索引号 index 以后，我们利用 sum 函数对用户数 user[]，设备数 facility[]数组中对应索引号的子数组求和，将结果保存在新的变量 num_user 和 num_facility 中，这样在输出了子节点的编号和地址信息之后，还可以追加输出整个父节点上所有的用户数总量和设备数总量。

最后，我还可以设立一些指标。由于实际使用的指标和临界值是各家运营商的机密，这里我们人为的设定一个指标为总用户数与总设备数之比 r，将它的临界值设为 a。在计算出 r 的值之后与临界值 a 进行比较，若大于临界值则输出一则警告字符串。

3.2.2 查询模式 b：通讯设备连接指标查询

如上文所说，我们可以人为的设定一些指标值，这里仍以总用户数与总设备数之比 r 为例，将它的临界值设为 a。与查询模式 a 不同的是，在查询模式 b 中我们可以直接输出该区域内所有超过指标临界值的父节点位置编号以及相应的指标 r 的值，并在最后统计超过指标的父节点总数。

上述功能的实现原理如下：用 for 语句遍历用户数 user[]数组和设备数 facility[]数组，利用 sum 函数对用户数 user[]，设备数 facility[]数组中对应位置的子数组求和，将结果保存在新的变量 num_user 和 num_facility 中，这样我们就得到相应父节点下总用户数和总设备数，接着我们可以计算指标值，令指标 r 是该父节点下总用户数与总设备数之比，接着再用 if 语句判断每一个父节点对应的指标值，如果超标则输出父节点位置标号并进行个数累加。

3.2.3 查询模式 c：通讯设备连接关系查询区域切换

如果用户已经对整个区域内的连接状态查询完毕，并需要查询其他区域时，我们就要用到查询模式 c：查询区域切换。在查询模式选择界面键入 c 之后，系统会回到请求输入数据文件名称界面。

其实现过程是在用 if 语句进行判断的时候，如果模式选择判断成 c 模式，那么用 break 语句退出查询模式选择模块，回到设备连接关系挖掘模块。

3.2.4 查询模式 end：通讯设备连接关系挖掘系统的关闭

当所有查询工作都进行完毕之后，我们需要退出这个查询系统。这个过程也是通过 break 语句来实现的。需要注意的是，这个查询系统是建立在 Python 上的，它本身并不是自动关闭，如果一直处于打开状态，会持续占据内存，当数据量比较大的时候，会影响 PC 机的性能，并且为了养成良好的软件使用习惯，设置系统退出环节是必要的。

3.3 交互功能优化

3.3.1 交互功能优化的必要性

设备连接关系挖掘系统包含着与使用者交互的模块。交互过程中使用者面临输入数据文件名称和选择查询模式的操作，这样的操作有可能会出现输入错误，如果因为一个字母键入错误就导致整个系统的中断，那么使用者对这个系统的用户体验将是非常差劲的。从整体程序的流程来说，不进行交互优化的程序是一个非闭合环。为了避免这样的事情发生，笔者在设计这个系统的过程中

增加了一些交互功能优化机制。

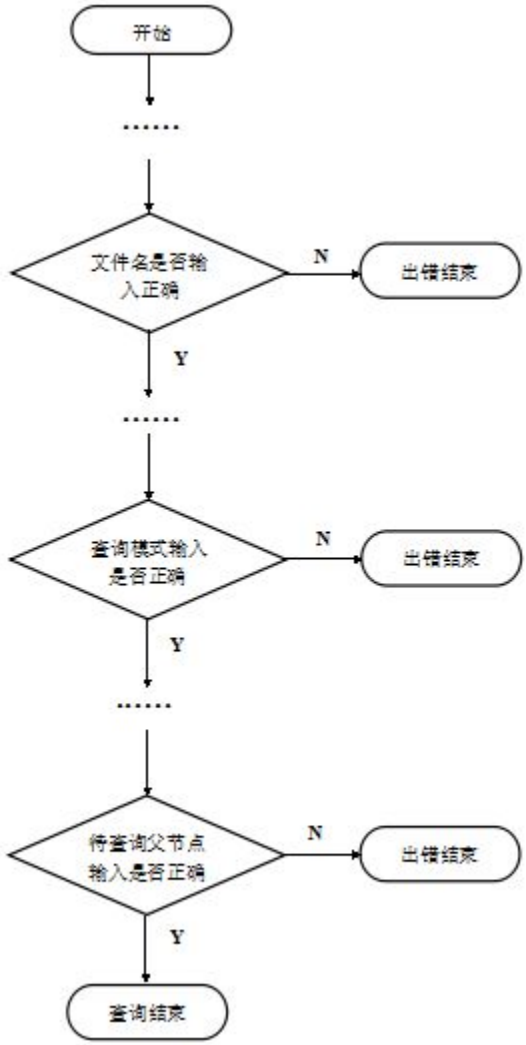

图 3.5 未优化的交互流程

3.3.2 交互功能优化机制

系统内的交互优化机制主要体现在简化了待查询文件文件名的输入以减小输入错误的可能性，例如，文件名为 xxx_backup.txt 的文件，只需输入 xxx（空格）txt 即可。其次，当使用者输入了错误的指令之后，系统给出纠错提示，并给予重新输入的机会。这样一来整个程序流程就变成了一个闭环，无论输入是否正确都不会中断程序，且都会得到相应的反馈。这个过程的实现主要依靠 if 语句和 print 语句。

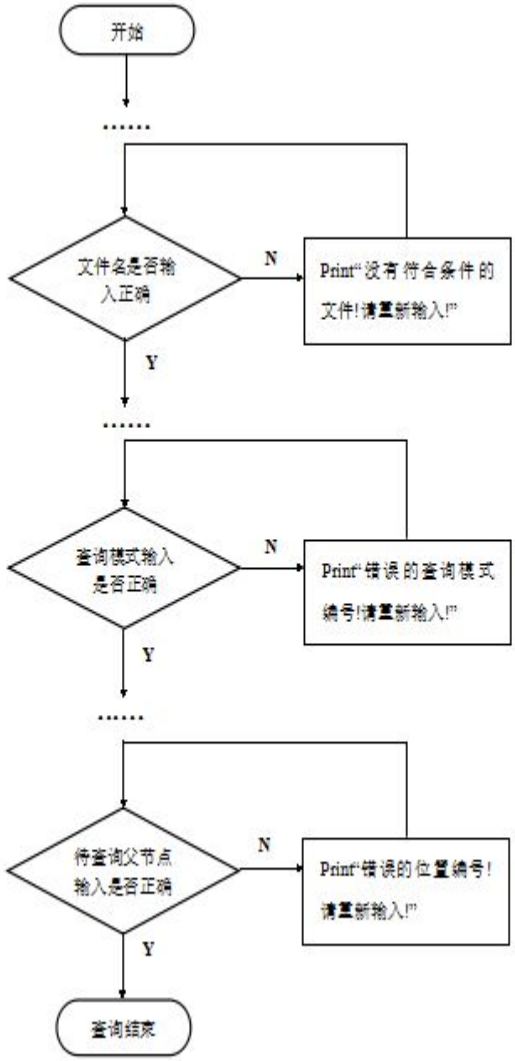

图 3.6 优化后的交互流程

3.4 本章小结

本章介绍了整个设备连接关系挖掘系统的设计与实现过程。首先，介绍了该系统的总体架构及相关详细设计。其次，对 4 种查询模式包括连接关系查询、指标查询、切换查询区域和查询的退出进行了详细的描述，并配以示意图。最后，对于系统中交互的优化部分，阐述了交互部分进行优化的必要性，并用交互流程图进行对比。

第四章 系统测试

4.1 测试数据的介绍与预处理

4.1.1 测试数据介绍

供测试的原始数据取自苏州市某运营商，数据范围涵盖了苏州市所有农村区域，数据内容包括该节点所属行政区域，子节点位置编号，父节点位置编号，用户数，覆盖数，标准地址名称等 15 个属性，数据量为 425637 条记录。数据的原始格式是 xlsx 格式。

	A	B	C	D	E	F	G	H	I	J	K	
1		区域名	ADDRESS_ID	LOCATION_ID	PARENT_LOC_ID	用户数	覆盖	标准地址名称	删除	级别	地址分行	创建时间
2	486100	工业园区	120000007795	120000007795	120000000403	0	0	苏州市工业园区胜浦镇宋巷村		村	60	2006
3	528675	高新区	120000009538	120000009538	120007538926	0	0	苏州市高新区枫桥镇新村茶巷		尾级地址(	90	2006
4	818224	吴中区	120000010979	120000010979	120000027853	2	7	苏州市吴中区越溪镇张桥村巷		组、队	70	2006
5	604667	金阊区	120000020418	120000020418	120000000055	0	0	苏州市金阊区长青镇艾家村		村	60	2006
6	671208	吴中区	120000020419	120000020419	20014	0	0	苏州市吴中区爱国村		村	60	2006
7	747251	吴中区	120000020420	120000020420	120000021261	1	0	苏州市吴中区金庭镇东村村爱国村		路	40	2006
8	522909	沧浪区	120000020458	120000020458	20001	0	0	苏州市沧浪区安全村		村	60	2006
9	515768	吴中区	120000020460	120000020460	20014	0	0	苏州市吴中区安全村		村	60	2006
10	670661	高新区	120000020464	120000020464	120000000114	0	0	苏州市高新区东渚镇安山村		村	60	2006
11	697505	吴中区	120000020465	120000020465	120000000169	1	0	苏州市吴中区光福镇安山村		村	60	2006
12	786582	相城区	120000020472	120000020472	120000000503	11	0	苏州市相城区湘城镇岸山村		村	60	2006
13	483215	相城区	120000020473	120000020473	120000000534	0	0	苏州市相城区阳澄湖镇岸山村		村	60	200
14	471945	相城区	120000020510	120000020510	20015	0	0	苏州市相城区白龙桥村		村	60	200
15	628508	金阊区	120000020528	120000020528	20006	0	0	苏州市金阊区白洋湾路南村		村	60	2006
16	530778	金阊区	120000020529	120000020529	20006	0	0	苏州市金阊区白洋湾民主村		村	60	2006
17	477768	平江区	120000020560	120000020560	20008	0	0	苏州市平江区班竹村		村	60	2006
18	430237	高新区	120000020578	120000020578	20003	0	0	苏州市高新区保卫村		村	60	2006
19	807105	高新区	120000020579	120000020579	120000000629	4	0	苏州市高新区浒关镇保卫村		村	60	2006
20	610028	高新区	120000020580	120000020580	20003	0	0	苏州市高新区宝山村		村	60	2006
21	697378	高新区	120000020581	120000020581	120000000629	0	0	苏州市高新区浒关镇宝山村		村	60	2006
22	642505	金阊区	120000020595	120000020595	120000000055	0	0	苏州市金阊区长青镇北村村		村	60	2007
23	526335	高新区	120000020597	120000020597	120000000114	0	0	苏州市高新区东渚镇北村村		村	60	2006
24	667667	吴中区	120000020598	120000020598	120000000179	0	0	苏州市吴中区郭巷镇北村村		村	60	2006
25	633119	相城区	120000020599	120000020599	120000000477	0	0	苏州市相城区渭塘镇北村村		村	60	2008
26	566769	吴中区	120000020612	120000020612	20014	0	0	苏州市吴中区北港村		村	60	2006
27	573450	相城区	120000020637	120000020637	20015	0	0	苏州市相城区北朗村		村	60	2006

图 4.1 原始数据

4.1.2 测试数据预处理

原始数据的数据量较大，根据测试的实际用途，提高关系挖掘的效率，我们要对数据进行预处理，这里的预处理主要是对数据进行筛选、删除和格式转化。

筛选地址级别。由于本系统挖掘的是运营商户外通讯设备连接关系，既不是连接到用户家里的设备，也不是连接局端的高层级设备，根据设备连接关系如图 3.1 所示，因此子节点的地址级别应该限定在“组、队”一级。

筛选行政区域。原始数据涵盖的农村区域分布在苏州的 7 个行政区，分别为工业园区、高新区、吴中区、金阊区、沧浪区、平江区和相城区。其中沧浪

区中“组、队”一级的农村地址只有 3 条，不具备研究价值，故将其删除。将数据按照行政区域分割后，6 个行政区满足条件的子节点数情况如下表所示：

表 4.1 各行政区农村区域“组、队”一级包含的子节点数一览表

区域名称	满足筛选条件的子节点数
工业园区	561
金阊区	319
高新区	3238
相城区	12617
平江区	47
吴中区	14847
总计	31629

删除多余信息量。原始数据中每条记录的属性量达 15 个之多，但对于本次测试来说，有用的属性量只有 5 个，分别是子节点位置编号、父节点位置编号、用户数、设备数和子节点地址信息，因此我们对不必要的信息进行删除，减少测试时的内存占用量，提高运行速度，减小出错的可能性。

转换数据格式。基于 Python 的设备连接关系挖掘系统需要使用文本格式的数据源，因此将 6 个行政区域的数据在去除表头之后分别转换成 txt 文本格式文件。

至此，我们就得到了 6 个 txt 文本格式文件，分别命名为：gxq_backup.txt、gyyq_backup.txt 、 jcq_backup.txt 、 pjq_backup.txt 、 wzq_backup.txt 和 xcq_backup.txt。每个文本文件都是 5 列多行的数据，5 列数据依次代表着子节点位置编号、父节点位置编号、用户数、设备数和子节点地址信息。这 6 个数据将会在接下来的测试中使用。

4.2 苏州市区农村区域户外通讯设备连接关系挖掘测试

4.2.1 设备连接关系挖掘测试

首先，将文件全部保存在系统程序的同一目录下，免去输入文件路径。接着，用 Python 运行我们的系统，进入请求文件输入界面。

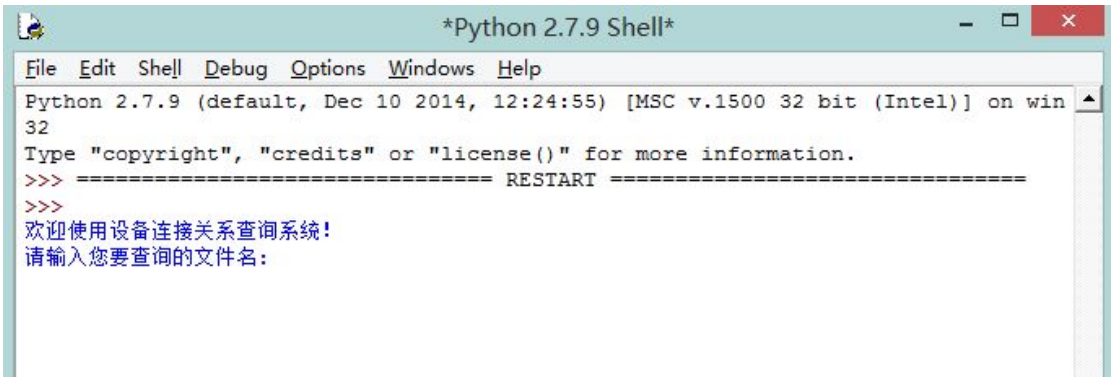

图 4.2 请求输入文件名界面

输入文件名称，以文件 wzq_backup.txt 为例，这里只需要输入“wzq（空格）txt”，我们就能将文件导入，并且系统会按照上文阐述的挖掘和归纳办法对文本里的连接关系进行挖掘，并在挖掘成功之后给出提示反馈。

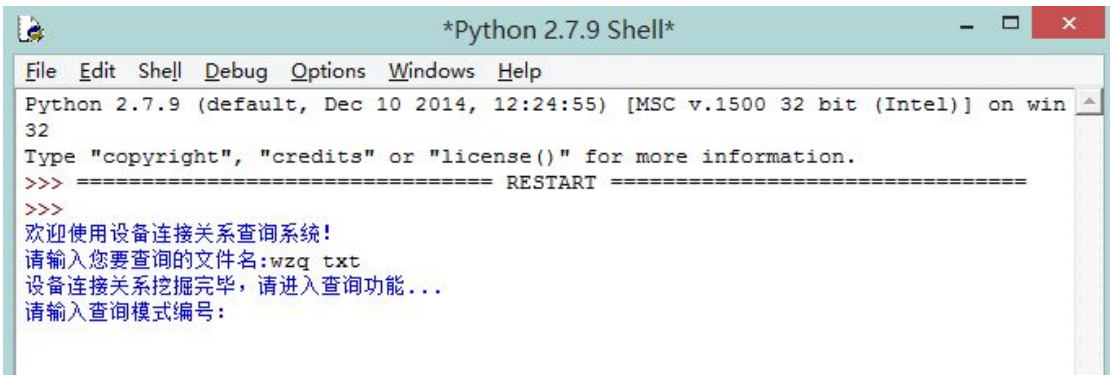

图 4.3 文件导入及关系挖掘

可以看到，wzq_backup.txt 已经被成功挖掘，那么我们依次测试所有文件，观察是否都能得到被成功挖掘的反馈。

表 4.2 6 个测试文本挖掘结果

文件名	挖掘结果
gxq_backup.txt	成功
gyyq_backup.txt	成功
jcq_backup.txt	成功
pjq_backup.txt	成功
wzq_backup.txt	成功
xcq_backup.txt	成功

4.2.2 文件的保存

文件内容被成功挖掘之后，得到的结果会从内存中保存到同目录下的一个名为“save”的文件夹里，生成一个格式为 save 的文件。这么做的优点在于每一个文件只需要进行一次挖掘，下一次再进行使用的时候只需调用 xxx_backup.save 文件即可，跳过了再一次计算的过程，提高了系统运行效率。其次，save 格式文件的大小要小于 txt 格式文件，当面对巨大数据量的时候，调用 save 格式文件可以节约内存。

表 4.3 两种格式文件大小对比

文件	txt 格式下大小（KB）	save 格式下大小（KB）	节约空间百分比
gxq_backup	246	198	19.51%
gyyq_backup	44	38	13.64%
jcq_backup	24	19	20.83%
pjq_backup	4	3	25.00%
wzq_backup	1008	935	7.24%
xcq_backup	968	775	19.94%

4.3 通讯设备连接关系查询功能测试

数据成功导入并进行挖掘之后，系统会自动跳出查询模式选择请求。这时，

我们进行查询功能的测试。

4.3.1 苏州市区农村区域户外通讯设备连接关系查询

设备连接关系查询，也即查询模式 a。这里我们以相城区设备连接关系作为测试对象，输入数据集，成功进行关系挖掘，并在查询模式选择界面输入“a”，出现输入需要查询的父节点位置编号的请求，查阅相城区数据的原始文件，发现“120000027872”是一个父节点位置编号，就以此为例，输入该父节点编号。

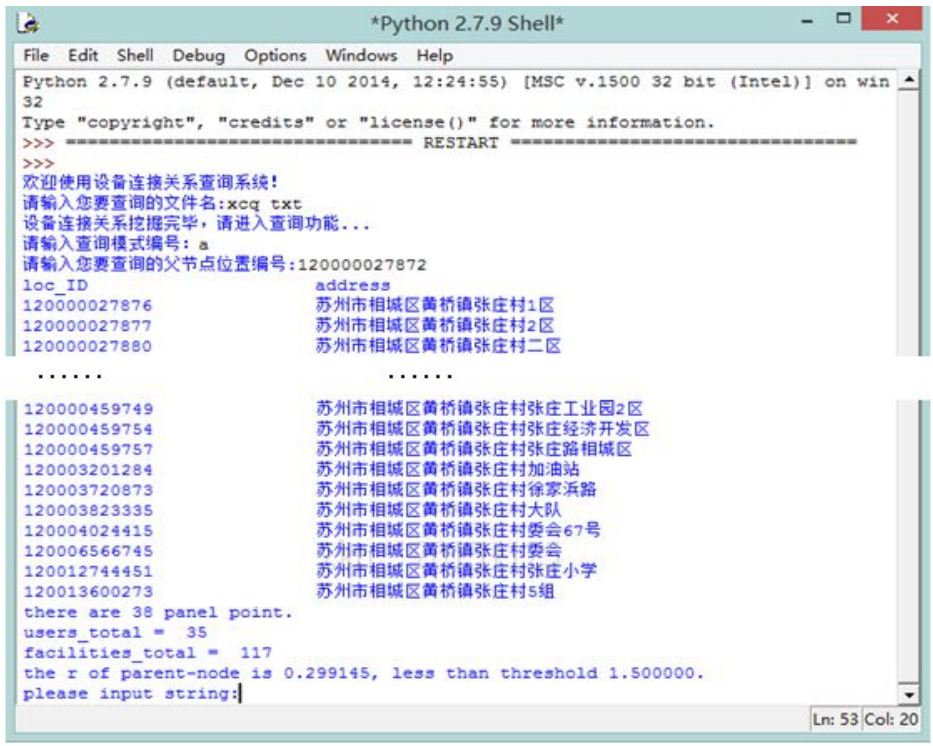

图 4.4 查询模式 a 测试示例

结果显示，父节点“120000027872”下共连接了 38 个子节点，程序输出所有子节点位置编号及相应的地址信息。这 38 个子节点上共有用户数 35 户，设备 117 台，总用户数与总设备数之比 r=0.299145，小于在测试之前事先设定好的临界值 1.5。

表 4.4 父节点“120000027872”连接信息一览表

父节点“120000027872”连接信息	
子节点连接数	38
总用户数	35
总设备数	117
设定指标是否超标	否

接下来，我们回到原始数据，验证通过连接关系挖掘系统挖掘出的连接关系是否正确，用户和设备统计量是否正确。利用 Excel 软件中的筛选功能，筛选出相城区中父节点位置编号为“120000027872”的信息。

图 4.5 关系挖掘结果验证图 1

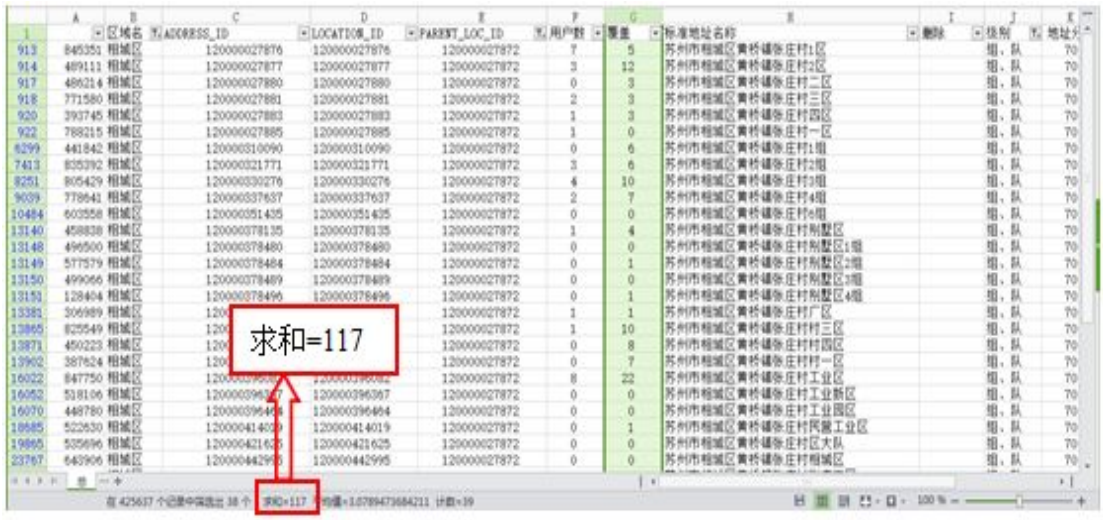

图 4.6 关系挖掘结果验证图 2

根据图 4.5 和图 4.6 我们可以知道父节点“120000027872”下连接的子节点数是 38，图 4.5 中的计数计算了表头，所以实际子节点数仍是 38，用户数求和是 35，设备数求和是 117，与表 4.4 中父节点“120000027872”的信息完全一致，说明我们的连接关系挖掘是准确、有效的。

通过重复上述的操作，我们可以在查询模式 a 中查询所有父节点位置编号下的连接情况。当所有父节点查询完毕需要离开查询模式 a 的时候，只需输入“stop”即可。

4.3.2 苏州市区农村区域户外通讯设备连接指标查询

设备连接指标查询，也即查询模式 b。这里的指标 r 我们仍然是设定为连接到一个父节点位置编号上的总用户数与总设备数之比，临界值 a 的设定在实际工作中是具有实际意义的，但它的具体值要受到诸多因素的影响，例如区域内的用户总数，区域内的设备总数，区域内整个通讯网络建设成熟程度等等。因此，临界值 a 的取值最终还是由各运营商决定。在测试中我们暂时将 a 的值设定为 1.5。测试区域以吴中区为例，由于在 4.2.1 节中已经挖掘过吴中区的数据文件，这里只需要输入 wzq(空格)save 即可，并选择查询模式 b。

```
欢迎使用设备连接关系查询系统!
请输入您要查询的文件名:wzq save
请输入查询模式编号: b
parent-node ID          r          threshold
120000024831            2.000              1.500
120000027316            2.300              1.500
120000602495            2.000              1.500
120000025262            2.000              1.500
120000024954            3.000              1.500
120000025552            1.625              1.500
120000026527            4.000              1.500
120000026260            6.000              1.500
120004391849            3.000              1.500
there are 9 parent-node's r more than threshold.
请输入查询模式编号:
```

图 4.7 查询模式 b 测试示例

从图 4.7 中我们可以看到，在指标临界值设定为 1.5 的情况下，有 9 个父节点上的指标超标，并且系统也给出了这 9 个超标父节点的位置编号和 r 值。系统的循环仍停留在选择查询模式处，这时，我们可以再选择查询模式 a 对超标的父节点进行逐个查询，查询模式 a 会输出子节点对应的地址信息，在实际工作中，运营商可以根据这个信息派遣装维人员前往该地进行处理。

通过重复利用查询模式 b，在临界值设定为 1.5 的情况下，我们可以得到整个苏州市区农村区域内设备连接指标超标的位置节点编号及个数。整理如下表：

表 4.5 各区域超标父节点位置编号统计

区域名称及节点个数统计	超标父节点位置编号	指标值
吴中区 总数：9	120000024831	2.000
	120000027316	2.300
	120000602495	2.000
	120000025262	2.000
	120000024954	3.000
	120000025552	1.625
	120000026527	4.000
	120000026260	6.000
	120004391849	3.000
相城区 总数：5	120000026526	1.636
	120000252827	4.750
	120000256444	12.000
	120000023716	2.667
	120003825483	2.000
高新区 总数： 2	120000025841	2.000
	120000273736	4.000
工业园区 总数：1	120000023404	1.667
金阊区 总数：0	无	无
平江区 总数：0	无	无
总计：17		

4.3.3 苏州市区农村区域户外通讯设备连接查询区域切换

当一个区域查询完毕需要查询下一个区域的时候，我们需要选择查询模式 c。测试过程与 4.2.2 节衔接，这时输入“c”，这时，系统退出了查询模式选择环节，请求输入文件名，这里我们把查询区域切换成高新区，输入“gxq(空格)save”，数据成功导入之后有进入到查询模式选择环节，我们选择查询模式 b，程序顺利输出了高新区连接指标超标的父节点位置编号。

```
欢迎使用设备连接关系查询系统!
请输入您要查询的文件名:wzq save
请输入查询模式编号: b
parent-node ID          r          threshold
120000024831               2.000               1.500
120000027316               2.300               1.500
120000602495               2.000               1.500
120000025262               2.000               1.500
120000024954               3.000               1.500
120000025552               1.625               1.500
120000026527               4.000               1.500
120000026260               6.000               1.500
120004391849               3.000               1.500
there are 9 parent-node's r more than threshold.
请输入查询模式编号: c
请输入您要查询的文件名:gxq save
请输入查询模式编号: b
parent-node ID          r          threshold
120000025841               2.000               1.500
120000273736               4.000               1.500
there are 2 parent-node's r more than threshold.
请输入查询模式编号:
```

图 4.8 查询模式 c 测试示例

4.3.4 通讯设备连接关系挖掘系统的关闭

当所有挖掘和查询请求都使用完毕之后，我们需要退出系统，这时只要在查询模式选择中输入“end”即可退出系统。测试中以平江区数据为例，输入“pjq(空格)txt”，程序反馈出设备连接关系挖掘完毕，并且求情输入查询模式编号，这时我们直接输入“end”，程序成功退出。

```
>>>
欢迎使用设备连接关系查询系统!
请输入您要查询的文件名:pjq txt
设备连接关系挖掘完毕，请进入查询功能...
请输入查询模式编号: end
感谢您使用本系统，再见!
>>>
```

图 4.9 查询模式 end 测试示例

4.4 系统操作容错测试

在使用系统的时候，难免会出现操作错误，当出现操作错误的时候，系统是否能给出正确提示并给予重新操作的机会是本节的测试重点。

4.4.1 数据文件导入操作容错测试

重新运行程序，在请求输入查询文件名的环节人为的输入一个错误的文件名称，如输入“aaa(空格)txt”，程序所在路径下并没有此文件，所以程序反馈了“没有符合条件的文件!”的提示，并回到请求输入文件名环节。

接着测试文件名输入中文件格式输入错误。程序所在路径下有 wzq_backup.txt 文件，但在文件名输入的过程中只输入“wzq”，这时，程序无法识别该文件名，所以依然反馈了“没有符合条件的文件！”的提示，并回到请求输入文件名环节。

```
>>>
欢迎使用设备连接关系查询系统!
请输入您要查询的文件名:aaa txt
没有符合条件的文件!
请输入您要查询的文件名:wzq
没有符合条件的文件!
```

图 4.10 容错测试示例 1

4.4.2 查询模式操作容错测试

测试过程衔接 4.4.1，我们输入一个正确的文件名，如“gyyq(空格)txt”，进入查询模式选择环节之后，输入一个错误的查询模式编号，如“q”，这时系统会提示“错误的查询模式编号，请重新输入!”并回到请求输入查询模式环节。接着，我们输入正确的查询模式编号“a”，进入请求输入父节点查询环节，我们输入一个不存在的父节点位置编号“123456789”，程序反馈“错误的位置编号！请重新输入！”并回到请求输入五节点位置编号环节。

```
请输入您要查询的文件名:gyyq txt
设备连接关系挖掘完毕，请进入查询功能...
请输入查询模式编号: q
错误的查询模式编号，请重新输入!
请输入查询模式编号: a
请输入您要查询的父节点位置编号:123456789
错误的位置编号!请重新输入!
请输入您要查询的父节点位置编号:
```

图 4.11 容错测试示例 2

将操作过程中可能发生的错误及其程序反馈整理如下表：

表 4.5 操作错误及程序反馈表

操作错误类型	程序反馈
文件名输入错误或文件不存在	输出“没有符合条件的文件!请重新输入!”并重新请求输入
查询模式输入错误	输出“错误的查询模式编号!请重新输入!”并重新请求输入
父节点位置编号输入错误	输出“错误的位置编号!请重新输入!”并重新请求输入

4.5 本章小结

本章对设备连接关系挖掘系统进行了测试。首先，对被测试数据进行预处理，使用了筛选，删除和格式转换的方法。其次，利用实际数据分别对系统的关系挖掘模块和查询模块进行了测试，结合实际操作的截图，证实了本系统的准确性、有效性。最后，测试了系统的容错性，表明本系统对错误操作有合理的反馈，有一定的交互性。

第五章 总 结

本文介绍了基于 Python 语言设计的一个能够对户外通讯设备连接关系进行挖掘研究的系统，并用苏州市某运营商的苏州市农村区域户外设备连接关系数据对系统进行了测试，得到了比较满意的结果。

本系统主要由连接关系挖掘部分和连接关系查询部分组成。连接关系挖掘部分根据各节点相互的关联关系，将连接在同一个父节点位置编号上的子节点全部进行归纳，于是子节点上包含的位置编号、用户数、设备数及地址信息都可以被归纳到相应的数组下的子数组中。于是，一定区域内的设备连接关系就全都被保存在这些数组中，以待查询环节使用。进入查询环节之后，有 4 种查询模式可供选择，分别是查询模式 a：通讯设备连接关系查询；查询模式 b：通讯设备连接指标查询；查询模式 c：通讯设备连接关系查询区域切换；查询 end：通讯设备连接关系挖掘系统的关闭。此外，系统中交互部分也得到了优化，使得系统在面对交互过程中错误输入时，不会轻易中断。

本研究中利用苏州市某运营商的苏州市农村区域户外设备连接关系数据对上述系统进行测试。测试中，该系统能够成功导入预处理过的文本数据，并对文本中的户外设备连接关系数据进行挖掘，同时生成挖掘结果文件，以备以后直接调用。这样既节约了存储空间，又免去了再一次计算的资源浪费。在查询模式的测试中，设定了相关指标和临界值。查询模式 a 的测试是以吴中区的父节点“120000027872”为例，结果准确呈现了该节点上所有连接状态及信息：子节点数 35，用户数 38，设备数 117，没有超过相关指标，经查原始数据，准确率达 100%。 查询模式 b 中，系统成功挖掘出区域内所有超过相关指标的父节点位置编号。利用查询模式 c，可以在苏州市多个行政区的设备连接数据之间切换，使查询更加便捷。选择查询模式 end，系统成功关闭。最后，进行了操作容错测试，一般的输入错误并不会导致系统中断。

结果表明，利用户外设备连接关系挖掘系统可以有效的对关系数据进行挖掘，归纳出关联关系。当然，本系统还有很多可以改进的地方。比如利用 Python 自带的 Tkinter 模块或者利用开源软件 wxPython 制作 GUI 图形界面，使得交互

更加愉快。其次，还可以利用 Python 的 pypyodbc 模块建立数据库，将所有的挖掘和查询操作都建立在数据库之上是更加行之有效的办法。

在实际工作中，通讯行业的装维人员可以利用这个工具，对辖区内的设备进行查询和监控，可以方便快捷的对整个辖区内的设备连接状况进行宏观的掌握，相信这个工具对于相关工作具有实际意义。

参考文献

[1] 张建光．国际电信联盟《衡量信息社会报告 2014》解读及建议[J/OL]．http://mall.cnki.net/magazine/Article/ZGXN201412012.htm

[2] 陈秋喜，2014 年通信网络物理连接设备行业分析报告[J/OL].2014.http://max.book118.com/html/2014/0608/8652543.shtm

[3] 康计良．Python 语言的可视化编程环境的设计与实现[D]．西安：西安电子科技大学硕士学位论文，2012．

[4] 于洁．基于 Python 的 LTE 多模数据卡 PC 侧 UI 的设计与实现[D]．西安：西安电子科技大学硕士学位论文，2013．

[5] 林晓丽，胡可可，胡青．基于 Python 的微博用户关系挖掘研究[J]．情报杂志，2014，33（6）：144-148

[6] 齐鹏，李隐峰，宋玉伟．基于 Python 的 Web 数据采集技术[J]．电子科技，2012，25（11）：118-120

[7] 武秀玲．Python 在海洋科学中的应用[D]．青岛：山东科技大学硕士学位论文，2013．

[8] 吴立，蔡小庆．使用 Python 语言分析金融数据的研究[J]．中国电子商务，2011．04：242

[9] 齐鹏，李隐峰，宋玉伟．基于 Python 的 Web 数据采集技术[J]．电子科技，2012，25（11）：118-120

[10] Jiawei Han, Micheline Kamber 著，范明，孟小峰等译．数据挖掘概念与技术[M]．北京：机械工业出版社，2007．

[11] 绍洛姆·韦斯，霓廷·因杜尔亚，张潼著，赵仲孟，侯迪等译．预测性文本挖掘基础[M]．西安：西安交通大学出版社，2012．

[12] 方新丽．浅议数据挖掘技术在计算机审计中的应用[J]．电脑知识与技术，2013，9（15）：3445-3446

[13] 丁兆云，贾焰，周斌．微博数据挖掘研究综述[J]．计算机研究与发展，2014，51（4）：691-706

[14] 喻云峰．数据挖掘算法的分析与研究[J]．科技广场，2010，09：54-56

[15] 张丽丽．数据挖掘技术的应用分析[J]．山西经济管理干部学院学报，2003，11（4）：75-76

[16] Magnus Lie Hetland 著，司维，曾军崴，谭颖华译．Python 基础教程（第 2 版）[M]．北京：人民邮电出版社，2010．

[17] 嵩天，黄天羽，礼欣．程序设计基础:Python 语言[M]．北京：高等教育出版社，2014．

[18] Liang, Y. Daniel．Introduction to programming using Python[M]．北京 : China Machine Press, 2013.

[19] Wes McKinney．Python for data analysis[M].南京：东南大学出版社, 2013.

[20] Michael Dawson 著，王金兰译．Python 编程初学者指南[M]．北京：人民邮电出版社，2014．

[21] Mark Lutz 著，李军, 刘红伟译．Python 学习手册[M]．北京：机械工业出版社，2011．

[22] John E.Grayson 著 陈文志等译．Python 与 Tkinter 编程[M]．北京：国防工业出版社，2002．

致　谢

时光如白驹过隙，不经意间，两年的硕士研究生学习生涯即将结束，值此论文完成之际，我要向所有指导、关心和帮助过我的师长、同学、朋友及家人表示最诚挚的感谢！

首先，我要特别感谢我的论文指导老师唐煜副教授。本文从选题、设计到撰写都是在导师唐煜副教授的悉心知道下完成的。在硕士研究生期间，唐煜老师以严谨的治学态度和深厚的专业造诣教育我，以崇高的个人品德和积极进取的精神引导我面对学习和生活中的困难。在硕士研究生生活即将结束之际，谨对导师一直以来对我的辛勤栽培和关心表示最崇高的敬意和最真挚的感谢。

其次，要感谢数学科学学院统计系的严继高、汪泗水和 Anandamayee Majumdar 等老师，感谢他们对我的指导和帮助，使我得以顺利完成各项研究工作和学业。感谢计算机科学学院的朱添宁同学，正是朱添宁同学对我在 Python 语言上的帮助才能使本文的工作能够顺利进行。

同时，要感谢应用统计专业硕士班的各位同学邓建卫、田少龙、朱晓楠、季敏杰等，正是有了他们给我在学习和生活中的帮助和支持，让我度过了快乐又充实的研究生时光。感谢我的父母和家人，他们对我一如既往的关心和支持，是我专注学业，追求理想的最大动力。

最后，再次感谢数学科学学院为我提供的这次珍贵的研究生学习生活，让我在学识和人生阅历上都更上一层楼。