**基于python的旅游信息爬取以及数据分析**

**Tourism information crawling and data analysis based on Python**

摘要

随着计算机网络技术的发展，近年来，新的编程语言层出不穷，python语言就是近些年来最为火爆的一门语言，python语言，相对于其他高级语言而言，python有着更加便捷实用的模块以及库，具有语法简单，语句清晰的特点，使得它在代码的编程中，变得更加简洁容易上手，另外，python应用特别广泛，作为是一门应用性广泛的语言，无论是游戏的开发，还是数据的爬取，再到网站的搭建，python都能轻松驾驭，其中，爬虫的应用，更加使得python这门语言为人所知。

作为网络搜索引擎的组成成分之一，爬虫能够有效的为我们搜索和爬取有用的信息，减少人工的操作，十分方便，在自己定义的条件下，采集得到某些网页的信息，比如房价、股票、招聘信息等，对于这些信息，我们可以对数据进行处理，从而得到我们所需要的信息。

本文通过python实现了一个马蜂窝旅游网站的爬虫信息搜集，并对马蜂窝网站中采集得到的数据进行分析处理，得到想要的数据。

**关键词：** Python Html 爬虫 旅游 马蜂窝

**Abstract**

With the development of computer network technology, in recent years, new programming languages have emerged endlessly. Python is the most popular language in recent years. Python has more convenient and practical modules and libraries than other high-level languages. , Has the characteristics of simple syntax and clear statements, which makes it more concise and easy to use in the programming of code. In addition, Python is particularly widely used as a versatile language, from website construction to data processing to gadgets. The design of small games can be done by python. Among them, the application of reptiles makes python a language more known.

Web crawlers can not only collect network information for search engines, but also act as a directional information collector. Under certain conditions, they can collect information about certain web pages, such as house prices, stocks, and recruitment information. For these information, we can analyze the data Process it to get the information we need.

This article is to implement a crawler's information collection and analyze the data collected by python to get the desired data.

**Key words：** Python Html reptile tourism Mafenwo

目录

[第一章 引言 6](#_Toc38124677)

[1.1 背景 6](#_Toc38124678)

[1.2 国内外研究情况 6](#_Toc38124679)

[第二章 爬虫语言以及工具的介绍 8](#_Toc38124680)

[2.1 Python 介绍 8](#_Toc38124681)

[2.1.1 Python语言的产生以及发展 8](#_Toc38124682)

[2.1.2 Python语言的特色 8](#_Toc38124683)

[2.1.3 Python语言的缺点 9](#_Toc38124684)

[2.2 URL介绍 9](#_Toc38124685)

[2.2.1 URL的定义 9](#_Toc38124686)

[2.2.2 URL的解读 9](#_Toc38124687)

[2.3 HTML介绍 10](#_Toc38124688)

[2.3.1 HTML的定义 10](#_Toc38124689)

[2.3.2 爬虫与HTML 10](#_Toc38124690)

[2.4 爬虫工具介绍 11](#_Toc38124691)

[2.4.1 Chrome 11](#_Toc38124692)

[2.4.2 Pycharm 11](#_Toc38124693)

[第三章 项目的需求以及设计分析 13](#_Toc38124694)

[3.1 项目的需求 13](#_Toc38124695)

[3.2 项目的任务 13](#_Toc38124696)

[3.3 设计分析 13](#_Toc38124697)

[3.3.1 城市编号的获取 13](#_Toc38124698)

[3.3.2 城市信息的爬取 13](#_Toc38124699)

[3.3.3 爬取信息的处理 14](#_Toc38124700)

[第四章 项目的实现 15](#_Toc38124701)

[4.1 城市编号获取的实现 15](#_Toc38124702)

[4.1.1 城市编号获取使用库的介绍 15](#_Toc38124703)

[4.1.2 城市编号获取过程 16](#_Toc38124704)

[4.2 城市信息获取的实现 22](#_Toc38124705)

[4.2.1 城市信息获取使用库的介绍 22](#_Toc38124706)

[4.2.2 城市信息获取过程 23](#_Toc38124707)

[4.3 数据可视化处理 31](#_Toc38124708)

[4.3.1 数据可视化处理所使用的库 31](#_Toc38124709)

[4.3.2 柱状图数据可视化代码实现 31](#_Toc38124710)

[4.3.3 饼状图可视化代码实现 32](#_Toc38124711)

[4.3.4 热力图数据可视化代码实现 33](#_Toc38124712)

[4.4 可视化图片展示 33](#_Toc38124713)

[第五章 不足之处 39](#_Toc38124714)

[第六章 期望与总结 40](#_Toc38124715)

[参考文献 41](#_Toc38124716)

[致谢 42](#_Toc38124717)

1. **引言**
   1. **背景**

随着近些年来网络的快速发展，网络进入了大信息时代，网络上的信息呈现爆炸式的飞涨，五花八门的信息显示，这使得人们在网络上寻找自己所需要的信息时，显得越来越困难，当然，一个问题的出现，必然有一个方法去对应，信息的剧增，与之对应的，便是搜索引擎的出现，比如google、百度等等，搜索引擎通过搜集网络上数以万计的不同类型的网页信息，并为其建立起索引，通过搜索引擎，即使网络上的信息种类繁多，我们还是能够通过关键词的搜索，搜索得出与之对应的信息的网页。

网络爬虫是一个自动化的程序，也是搜索引擎的组成部分之一，不同的搜索引擎，可以通过不同的搜索需求，选择合适的爬虫方法来搜集网络上的信息，传统网络爬虫主要从一个url开始，通过爬取目标网页的url，观察其组成结构特点，按照结构规律，构建新的url，不停的将新的url放入队列中，循环爬取，最后直到需求完成为止。优秀、高效的爬虫程序，能够使人们网络上找到更加精准的信息。

本文通过python语言，实现了一个对于马蜂窝旅游网站的信息采集分析，通过对马蜂窝旅游城市的城市编号爬取，根据马蜂窝网站的网址规则，通过拼接得到马蜂窝旅游网站的城市url，根据获取得到的url，进入马蜂窝旅游城市页面，观察页面组成结构，通过标签定位，爬取我们所需要的页面信息，将其保存入本地文件，再对文件里面的数据进行数据处理，可视化分析，告诉你旅游去哪儿好。

* 1. **国内外研究情况**

python作为搜索引擎技术组成部分之一，自九十年代初诞生了第一个[网络](http://www.lwfree.cn/jisuanjilunwen/)爬虫以来，python技术已经经过了20多年的发展，经过这么多年的发展，python技术也趋于成熟，逐渐覆盖了网页开发、游戏开发、爬虫分析可视化、脚本开发等等专业领域, 并通过其简单易懂、代码编写效率高的特点，成为目前最为火热的开发编程语言，也正是其应用广泛，上手容易，语法强大，代码可读性强的特点，受到了很多人的喜爱，让很多人爱不释手。同时，由于不同的人对于数据爬取内容的差异，常常会产生不同的需求，因此爬虫主要可以分为以下两种：

1. 通用爬虫：通用爬虫其实很简单，它的作用就是将网络上的网页下载下来，保存到本地，其实就是生成一个副本，作为网页的备份。所以通用搜索一般得到的都是网页，对于用户来说，网页的内容十之八九都是没用的，要的只是网页中的某些特定的信息。所以就有了我们第二种爬虫方法。
2. 聚焦爬虫：聚焦爬虫是比通用爬虫更加复杂一点，它不同与通用爬虫，通用爬虫是将网页保存到本地，但是聚焦爬虫，它不会将整个网页爬取下来，它会筛选得到网页中我我们所需要的东西，而筛选的条件规则是由我们自己定义的，相比通用爬虫，聚焦爬虫它做的工作更多，但是得到的数据更加符合我们的要求，本文所实现的爬虫，就是通过聚焦爬虫所实现的。
3. **爬虫语言以及工具的介绍**
4. **Python 介绍**
5. Python语言的产生以及发展

Python是上世纪八十年代末开始由Guido开发，九十年代初发行的一门编程语言，其作者参与过ABC语言的开发，所以对ABC语言其中的弊端有所了解，认为ABC语言虽然语言优美，功能强大，但是却受限于它的非开发性，没有得到广泛的应用，因此想开发出一门新的语言，能够摒弃掉ABC语言的弊端，打破这个限制，正是如此，python就诞生了，python刚刚推出不久，就迅速得到了各行人士的青睐，经过多年以来的不断改善，python语言已经逐渐稳固它在编程语言中的地位。

根据从TIOBE网站上得到的数据，从2002年开始，python语言就趋于一个稳定上升的状态，呈线性增长，正是如此，python成为了近年来最受欢迎的编程语言之一。

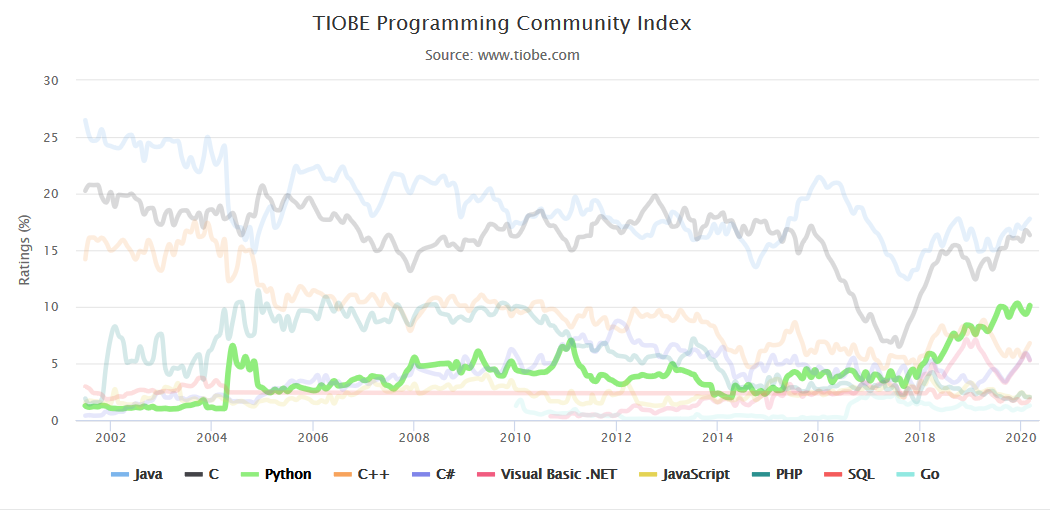


图2.1.1 计算机编程语言热门排行榜

1. Python语言的特色

首先，Python是一门强大、灵活的语言，与C、C++、Java等语言相比较，python的语法简单优雅，往往不用通过很复杂的代码，就能实现强大的功能，例如Java需要100行代码才能实现的功能，python往往只需要一半或者更加少，大幅度降低了学习与使用的难度，很适合人们的学习。

其次，python是一门开源免费的语言，简单的来说，就是不花钱就能使用，我们可以直接下载安装运行，也能对其源码进行修改，十分便捷。

再次，效率极高，python被称为胶水语言，拥有强大的第三方库，人们往往只需要将库下载调用，在库的基础上进行开发，就可以实现很多复杂的功能，节省了不少时间，避免了重复造轮子的现象。

最后，python语言可移植性强，正是由于python的开源性，使得python拥有了良好的可移植性性能，让其可以被移植到其他的平台之下，例如我们所熟知的window系统，Linux系统，安卓系统和苹果系统等等。总而言之，python是一门用途广泛，又容易上手的编程语言。

1. Python语言的缺点

Python虽然拥有众多的优点，但是事物往往很难做到十全十美，编程语言也一样，python虽然简单优美，但是运行速度相比其他编程语言来说，却显得缓慢，对比C语言来说，C语言是先编译后运行的，编译的时候就已经将代码转化成了CPU所能执行的机器码，而python是解释性语言，因此它的代码在运行的过程中，需要先将代码编译成CPU所能理解的机器码，这个过程就需要耗费很多时间，导致了python的运行缓慢。

另外，由于python的开源性，导致了python代码无法加密，实际上，如果我们要发布一个python程序，其实就需要将我们所敲打的源码发布出去，这样别人就直接得到了你的源码，能够轻易的使用并修改你的源码，而C语言是先编译后运行的，它的发布是发布已经编译后的机器码，别人是很难逆推得到源码的，其实，基本所有的编译性语言，都不需要担心源码的泄露，而像python这种解释性的语言就比较难了，因此在保密度这一块，python还是略显劣势。

1. **URL介绍**
2. URL的定义

URL，通俗的来讲的话，URL其实就是网络地址，每一个URL，就代表着特定的资源在网络或者电脑上的位置，这个位置既可以是电脑的本地磁盘路径，也可以是局域网上的某个计算机，但是，总的来说，URL更多的是指网络上的网页站点。而这里所说的特定的资源，可以是一个HTML界面、也可以是一张照片、一个文本、或者是一个文件等等。

1. URL的解读

URL是由不同的组件组成的，就拿本文所要爬取的网站马蜂窝来说，马蜂窝的网站地址为：<https://www.mafengwo.cn/mdd/>，这个网站就相当于一个URL，我们可以这样解释它：

（1）这是一个https协议的网站，它告诉了浏览器访问资源的时候，应当使用https协议，在目前，网络上使用的协议大多为HTTP或者https（加密）。

（2）[www.mafengwo.cn](http://www.mafengwo.cn)，是域名，指向你所要访问的资源的服务器端，这里也可以使用IP地址显示。

（3）mdd，这里表示的是访问资源在该服务器上的路径，但是有时这个路径会出错，最常见的问题就是该路径的资源已经被移走或者不存在，导致寻找失败。

1. **HTML介绍**
2. HTML的定义

HTML，是一种超文本链接标示语言，是网站搭建的基本骨架，简单的来说，HTML就是网页的载体，它主要由两部分组成，一个是head，另一个是body，head是描述浏览器的主要信息，body是用来装载网页界面的主体内容，网页上所看到的文字信息或者图案，都是存在body里面，在HTML界面中，存在着超级链接，所谓的超级链接，其实就是将URL地址放入了界面中，人们可以通过鼠标的点击，可以使浏览器对其进行解析，并能够快速便捷的跳转到URL指定的资源界面，正是如此便捷的操作，使得HTML受到了很多编程人士的喜爱，也是它成为应用广泛的原因之一。

1. 爬虫与HTML

本文之所以介绍HTML界面，其实是和我们爬虫的设计有一定的关联，在HTML中，为了方便的网页内容的处理，在HTML加入了标签这种东西，标签是由<>组成，其中分为起始标签和结束标签，结束标签在<>的基础上，加多一个/，即</>，而<>中的内容，则是根据自己所要编写的内容决定，例如我们想要定义一个标题，则需要这么写：<title>，类似的例子还有很多，具体要思考我们标签是要干什么再来决定。

在我们的爬虫中，我们就是需要根据这些标签的位置，来爬取我们所需要的信息，根据标签的定位，观察标签地位的规则，获取需求标签位置下的内容，通过字符分割或者直接爬取的方式，将其内容中有用的信息存储下来，达到我们爬取的目的。

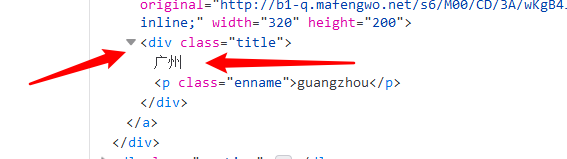


图2.3.2 HTML下的标签

1. **爬虫工具介绍**
2. Chrome

本文进行爬虫的浏览器选择的是chrome浏览器，之所选择chrome浏览器，首先，chrome浏览器的界面简介，在所有浏览器中，chrome浏览器算是性能最好的浏览器，其次，chrome浏览器全面支持HTML，这对于我们之后的爬虫有良好的帮助，在兼容性方面，chrome浏览器的兼容性问题很好，使用过程会给人一种良好的体验，另外，爬虫所使用的selenium库，与其他浏览器的驱动存在一定的bug，导致某些功能无法实现从而出错，我们在使用selenium时，需要在chrome浏览器对应的版本下载安装Chromedriver驱动程序，并放入python文件下，便于我们之后的调用，最后，chrome浏览器推出了headless模式，即无界面浏览模式，在没有浏览器界面的情况下，依旧能实现与有浏览器界面相同的操作，这对于我们爬虫抓取来说，是一个十分便捷的功能，但是在本文中，为了更加直观的表现出爬虫过程，我并没有把无界面浏览的代码写上去。

1. Pycharm

我们在做一件事情的时候，既要有一定的头脑，也需要趁手的工具，才能有更好的效率，编程也是如此，虽然python安装后有自带的编辑器IDLE，但是对于我们来说，IDLE对于大多数新手乃至学习过程中的人们并不太友好，没有提示，没有补全，只有最后的运行结果或者报错，这对于大多数人来说，是一种很难受的体验，大大的降低了人们学习过程中的效率和导致了宝贵时间的流逝。

所以说，只有选对了正确的工具，才能使工作的效率以及代码的质量得到提升，在本文的爬虫过程中，选择的工具是Pycharm，Pycharm是python的一种IDE，能够帮助人们在进行python语言编写的过程中，节省时间，提高生产效率，提高代码的质量，在使用的过程中，如果出现一些语法错误或者拼错代码，Pycharm能够实现实时监测，一旦发现错误，将会有明显的下划线报红提示，能够使人第一时间发现错误，修改代码。另外，Pycharm的补全功能也十分便捷，一行代码在编写的过程中，并不需要全部敲打，只需要有首字符提示，便能显示出与之对应的代码语言，十分的节约时间。在程序运行出错的同时，会显示语句出错的地方，鼠标点击对应的出错点，就能实现智能跳转。对于文件的管理来说，Pycharm也是优于IDLE，代码文件直接就在文件夹下，可以有多种不同的代码文件夹存在，需要哪些代码文件，很方便直接点击就能打开。最后，Pycharm的debug检查，有时候会遇到代码不报错，但是却得不到自己想要的数据，这个时候，debug检测就派上大用场了，使用debug功能，能够看到代码的执行过程，从细节上找到代码陷入错误的地方，是一个很好用的功能。所以，本文的爬虫爬取信息所使用的工具就是Pycharm。

1. **项目的需求以及设计分析**
2. **项目的需求**

如今，人们的生活质量越来越好，水平条件越来越高，也使得更多的人注意到身心的放松与娱乐，很多人在自己闲暇的时候，都希望能度过一个好的时光，正因为如此，旅游行业逐渐火热了起来，渐渐的进入到了人们的生活之中，但是很多人在选择旅游的时候却不知所措，因为他们不知道去哪里玩好，有什么好玩的，有什么好吃的，所以，本文设计了一次爬虫，爬取马蜂窝旅游网站的旅游信息，将信息数据进行分析，筛选出马蜂窝旅游网站的热门城市、热门城市景点、热门小吃等等的数据信息，让人们得以在众多的旅游城市选择中，找到那些热门景点以及热门的小吃，少走弯路，找到一个最适合自己旅游的城市，拥有一个愉悦快乐的假期旅游。

1. **项目的任务**

首先，旅游，我们需要先确定下来一个城市，因此，我们任务要求第一件就是马蜂窝旅游城市中排行前10的热门城市各自为什么，其次，我们需要得到各个城市的景点数据，根据数据分析得出排行前15的热门景点为哪些，最后，爬取与美食相关的信息，得出旅游中最具代表性的美食前15的排行。

1. **设计分析**
2. 城市编号的获取

首先，我们爬取旅游信息，肯定是爬取众多城市的旅游信息，不同的城市在马蜂窝旅游网站中的URL是不同的，但是，通过对比我们可以发现，在马蜂窝旅游网站中，所有的城市以及城市景点信息，都是由特定的五位数字或者六位数字组成的，这对于我们爬取不同城市的旅游信息是一个突破口，根据这个数字，我们就能拼接得到不同城市的不同URL地址，得到城市旅游的界面。

1. 城市信息的爬取

在获取得到城市编号后，我们就能得到马蜂窝旅游网站的城市的URL地址，通过地址我们就可以进入到城市旅游的界面，这个时候我们就需要考虑，我们应该爬取什么，抓取哪些信息，哪些信息是有用的，能够支持我们爬取信息后数据分析的可信度，在这里，我们是根据马蜂窝旅游网站里面 不同城市的游记的数量、印象的标签数、特色美食的排行、购物娱乐的排行等等来得出的信息。

1. 爬取信息的处理

在我们得到城市的具体旅游信息后，最后就是数据的可视化处理，首先，我

需要将马蜂窝旅游网站中游记前10的热门城市、前15的景点标签类热门城市、前15的的餐饮标签热门城市、前15的娱乐购物标签类热门城市可视化处理，采用柱状图显示。接着，我们对景点人气前15的城市景点、餐饮人气前15的城市美食、娱乐购物人气前15的城市娱乐可视化处理，采用的也是柱状图显示。最后，我们对热门城市前20的热门城市进行热力图显示，这就是全部的信息处理可视化了。

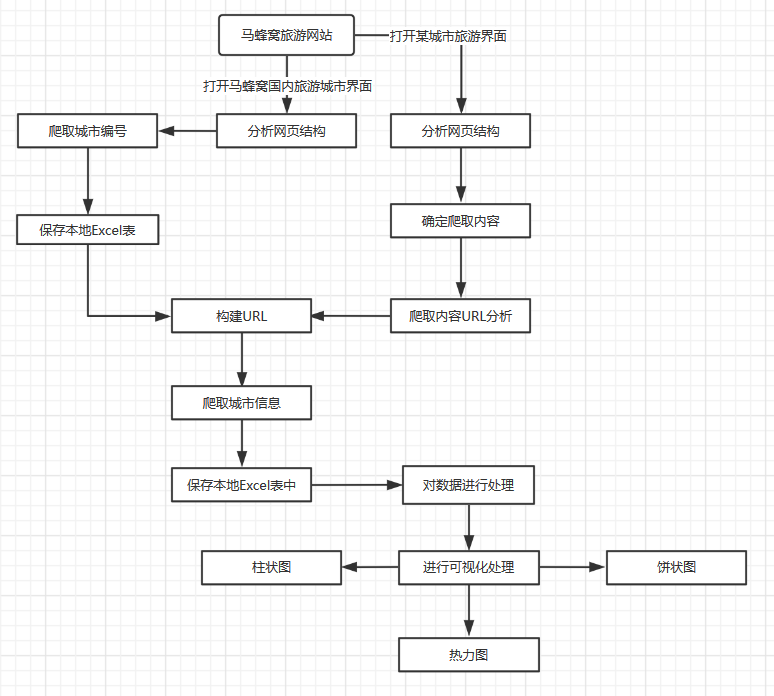


图3.3 程序流程图

1. **项目的实现**
2. **城市编号获取的实现**
3. 城市编号获取使用库的介绍

os库，python里面os库是用来处理各种文件和目录的，在os库里面，python提供了很多种方法，例如， access(path, mode)，用于文件的权限检查，path是指文件路径，mode则拥有四个参数，可以分别检测出path路径下的文件是否存在，是否可读，是否可写，是否可执行。与之对应的， chmod(path, mode)，则是用于文件的权限修改。Remove方法，则是用来删除path路径下的文件，os库的方法还有很多，这里就不一一举例出来了，os库其实就是python对于我们操作系统功能的一个普通操作，本文这里我们使用到的os库里，是调用了一个os.chdir(path),用来改变文件路径，使得我们爬虫保存下来的数据文件能够存储到指定的位置。

urllib库，python里面urllib库十分的便捷，它为我们提供了一系列的URL操作，这里要说明的一下就是，以前的python版本中，urllib与urllib2是分开来的，换句话说，urllib2是urllib的升级版，不过这两个又互相存在着不同点，它们所包含的函数有些是不互通，所以在python2.x中，它们是混合搭配使用的，无法相互替代的，但是在python3.x后，urllib2是合并到了 urllib中，这里我们安装的python是3.x的，所以我们直接使用的就是urllib，在urllib模块中，我们需要的用到的就是方法是request，request可以帮助我们抓取URL下的内容，是最基础的HTTP请求模块，它也可以为我们爬虫实现一个伪装操作，我们可以通过添加Headers，将我们的操作伪装成一个浏览器，模拟我们人为的操作，类似于我们平常上网的操作，输入网址点击跳转，就能跳转到我们所需要访问的网页的步骤是相同的。

BeautifulSoup4库，BeautifulSoup4是我们爬虫所经常使用的一个库之一，在HTML界面中，数据的位置是有一定的规则和结构的，BeautifulSoup4就是一个可以根据HTML页面结构抓取我们所需要的数据的库之一，类似像BeautifulSoup4还有lxml和pyquery等等，这里使用BeautifulSoup4是因为BeautifulSoup4简单容易上手，而且它的文档容错率较高，对于我们这种数据抓取保存还是比较合适的，BeautifulSoup4使用soup可以直接获取得到节点的元素或者名称，但是在同级中有多个相同节点的时候，soup只会选择第一个，我们所爬取的内容很多都是同一节点下多个元素的数据，所以这里使用BeautifulSoup4里面的函数最多的就是find\_all（）与find（），find\_all（name,attrs,recursive,text,\*\*kwargs）能帮助我们查找符合括号中条件的所有元素中的参数，name所指的是命名为name的所有标签，attrs表示属性，例如id、title、class等等，这里的class由于是python的专有词，所以使用的时候，需要特殊处理下，我们只需要对其后面加下划线，就能解决，即class\_，find（）和find\_all（）使用差不多，但是使用find匹配的是第一个元素，有时候我们需要的是某一个元素下面下属的多个元素，所以这里我们有时候会使用到find（）与find\_all（）的混合使用抓取所需要的元素下的数据。

Pandas库，pandas库是python中的一个科学计算库，它可以说是基于Numpy库的一个工具，这里的Numpy后面也会用到，他们最大的区别在于，Numpy是用于数值计算的，它是一个多维数组的容器，而pandas库更多的是用来做数据处理的，可以说是为了解决数据分析任务而衍生出来的一个工具，同时它也算是一个表格容器，pandas可以对数据进行导入、处理、输出、统计等等一系列的操作，能够对数据进行一个有序的整理，这里pandas库，首先我们使用到的是它的DataFrame，由于我们爬取的数据很多，如何将这些数据进行处理保存，这是一个问题，我们这里使用的是pd.DataFrame()，用于建立一个二维数组对象，再将数据放置进去这个对象里面，其次，有了数据之后，不同数据是在不同的地方爬取下来的，我们就需要对某些数据进行合并才能显示出效果来，这里我们就需要用到pd.concat()方法，可以对多个DataFrame对象进行合并，concat方法里有多个参数需要我们选择，具体怎么做要看我们需要什么，最后，pandas也为我们提供了文件数据的读取功能，我们第一步是将马蜂窝各个城市的专属数字爬取下来保存成一个Excel文件，后面第二部分会调用到这些数据，所以pandas的read功能就很好的解决了我们这个问题，通过pd.read\_excle()，我们就能很好的打开excel文件，从而从中拿取我们所需要的数据。

Selenium库，Selenium用于网站的自动化测试，它适用于当前很多的主流浏览器，它的作用是用于模拟我们对浏览器的操作，如点击、下滑、跳转等等，同时，我们还需要下载对应的浏览器驱动来配合Selenium操作，实现操作目的。

1. 城市编号获取过程

首先，我们进入马蜂窝网站，我们需要爬取的是国内旅游信息，总所周知，中国的地区模块是由省份以及直辖市组成，我们可以在马蜂窝热门目的地国内省份开始寻找有用的信息。

按F12打开开发者工具后，我们将箭头点击页面中的省份名称，开发者工具就会直接定位跳转到箭头点击位置的当前页面标签下。

图 4.1.2.1 省份URL抓取

通过图片中我们可以看到，省份的URL信息是放置在一个属性为<div class=“hot-list clearfix”>下的dt标签里a标签里面，所以我们需要做的就是，定位到<div class=“hot-list clearfix”>标签下，寻找所有dt标签，将dt标签里面的a标签的href属性内容抓取下来，这就是我们爬取马蜂窝城市URL的第一步，抓取省份URL。

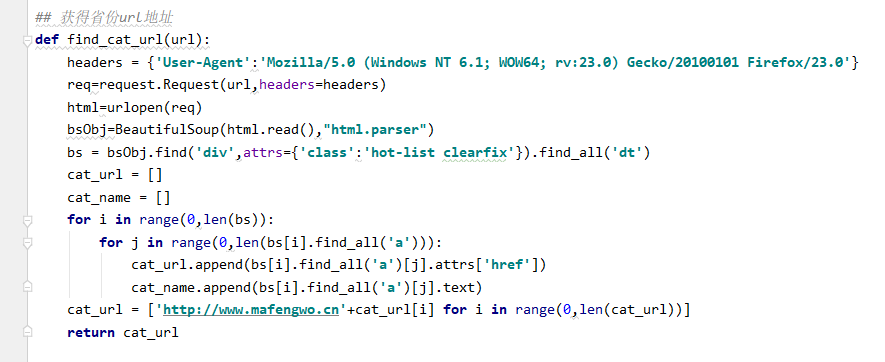


图 4.1.2.2 省份URL抓取代码实现

在Python里，python为我们提供了很多的函数，这些函数是python内部以及包装好了的，使用起来非常方便，同样，我们也可以自己在python中写自己的函数，这样是为了方便在其他地方使用同样的代码时，不需要重新写一边，而是只要调用这个函数就能实现，减少了代码累赘。

这里我们将获取省份URL地址的代码写进了一个名为find\_cat\_url函数里，首先，很多网站是不允许爬虫爬取数据的，这样会给网站额外增加运行负担，所以为了不被网站发觉我们是爬虫，一般都会用表头伪装成浏览器，就像代码截图中一样，在find\_cat\_url函数中，我们首先使用了一个表头伪装成火狐浏览器，假装是认为浏览网站信息，而不是爬虫，其次，我们使用了request方法，向马蜂窝网站发送请求，得到响应后，我们使用beautifulsoup库的html.parser解析器进行解析，按照属性和标签名找到我们所需要找到的省份数据，所以这里我们使用find（）方法，找到属性为 “class”：“hot-list clearfix”的标签，在这个标签下，使用find\_all（）方法，找到当前标签下的所有dt标签，新建两个数组，命名为cat\_url和cat\_name，cat\_url用来放置省份的URL，而cat\_name用来放置省份的名称，这里使用了一个for循环来爬取数据，for循环的长度为dt标签的长度，毕竟我们数据的多少就取决于dt标签里多少个省份，for循环遍历所有的省份，根据我们之前的截图，我们可以知道省份的信息是放置在dt标签下的a标签中，所以我们遍历所有的a标签，将a标签中的属性为href的信息添加到cat\_url中，将a标签的文本信息添加到cat\_name中，将cat\_url返还，这样我们就得到了马蜂窝网站的省份url地址。

接着，得到省份的URL地址后，我们进入到马蜂窝省份界面，查看界面源代码，查找下如何抓取省份下城市URL。

图 4.1.2.3 省份城市界面分析

进入到省份界面后，这里以云南省为例，我们可以发现，城市的信息是位于网页界面的目的地下的热门城市里。



图 4.1.2.4 省份URL分析

进入到省份热门城市里后，我们会发现，省份的URL和我们之前爬取的的URL有所不同，之前的URL里的部分组成为travel-scenic-spot/mafengwo，但是进入省份热门城市界面后，URL组成部分变为了mdd/citylist，不过其他组成部分还是一样的，所以这里我们需要对之前爬取的省份URL进行处理，将travel-scenic-spot/mafengwo替换成mdd/citylist/。

图 4.1.2.5 城市URL分析抓取

进入省份热门城市后，通过开发者工具我们可以发现，例如我们打开云南的丽江看看，标签中已经蕴含了丽江的跳转链接，即为travel-scenic-spot/mafengwo/10186.html，与我们单独手动打开丽江链接对比可以发现<http://www.mafengwo.cn/travel-scenic-spot/mafengwo/10186.html，a>标签中的属性data-id就是我们所需要爬取的城市URL特定数字，有了这些数字，我们只要通过<http://www.mafengwo.cn/travel-scenic-spot/mafengwo/data-id.html>，就能去到任何马蜂窝热门城市界面，所以，这里的属性data-id下的数据就是我们所需要爬取的，同时还有标签属性为class=“title”下的文本内容，即城市名字是我们所需要爬取的。



图 4.1.2.6 城市URL抓取实现代码

同样，我们也是将城市URL抓取的方法用函数包装起来，函数名为find\_city\_url，这里我们也是新建了两个列表，一个city\_name\_list，用来保存城市名称，一个city\_url\_list，用来保存城市URL中的特定数字，首先也是一个循环语句，对url\_list进行遍历,这里的url\_list其实是函数调用前，已经对find\_cat\_url函数中的省份URL传入到了url\_list中，我们这里遍历后的第一件事就是将第一次爬取下来的省份URL中的travel-scenic-spot/mafengwo全部换为mdd/citylist/，这里使用的是replace进行替换，可以将旧字符串替换成新的字符串，再将替换后的URL传递给浏览器，接着是一个while循环，还是和之前一样使用的是beautifulsoup进行解析，根据之前的开发者工具我们可以看出，我们所需要的属性data-id，是位于a标签下，在a标签下，所有的a标签都有一个共同属性，那就是data-type=“目的地”，所以我们这里使用find\_all，找到所有a标签下属性为data-type=“目的地”的位置，将城市的名称，即a标签的文本内容通过循环添加到city\_name\_list中，将属性data-id的数据也通过循环添加到city\_url\_list，一个省份有很多个城市，所以不可能一页就能列出来的，所以这里我们就用到了之前介绍的Selenium，通过Selenium模拟滚动，我们页面向下滚动800像素，然后调用js脚本，通过开发者工具我们可以得到马蜂窝网站下一页的标签，<a rel="nofollow" data-page="2" href="#" class="pg-next \_j\_pageitem">后一页</a>，所以我们只要找到class为pg-next的地方，模拟点击就能自动翻到下一页，返回city\_name\_list和city\_url\_list，从而实现整个省份的城市URL获取。



图 4.1.2.7 城市URL抓取函数调用

这部分代码是函数的调用，首先是url地址，为马蜂窝旅游网站地址，然后调用find\_cat\_url函数，将获取的省份URL传递给了url\_list，接着调用了find\_city\_url函数，将城市名和城市URL通过二维数组保存，再输出到excle表中。

图 4.1.2.8 城市URL爬取数据

这里就是代码运行后所获取的数据，由于马蜂窝网站结构的设计，我发现属于省的城市数据能够爬取得到，但是直辖市因为标签位置的不同而无法抓取，由于马蜂窝旅游网站直辖市数据量较小，只有四个，所以这里就直接人为的添加了上去，这里一共爬取了一千八百多条数据，实际上中国并没有那么多城市，主要是马蜂窝旅游网站将城市与旅游景点也一块算进热门城市里面了，所以导致了数据的溢出。

1. **城市信息获取的实现**
2. 城市信息获取使用库的介绍

这里使用的库与之前我们城市编号获取的部分库相同，也使用了request库，os库，beautifulsoup4库和pandas库，这里就不再介绍了，这里介绍的是城市编号获取代码部分所没有使用到的库。

Numpy库，前面我们说过，pandas可以说是基于numpy的一个工具，但是它们运用于不同的地方，numpy应用于多维数组的运算以及矩阵运算，这里我们之所以需要用到numpy库，是因为我们需要对数据进行降维，将多维数组降为一维，在numpy中，降维有两种方法，一种是np.flatten(),使用这个函数，它返回给我们的是降维数组的一份降维拷贝，即原本的数组是不会改变的，但是这里我们对降维后的数组是有要求改变的，所以我们这里使用的是第二种方法，就是np.ravel()，使用这个函数，我们就能对原数组进行降维。

Socket库，一开始在进行爬虫的时候，经常程序会卡死不动弹，这就对我们数据的爬取造成的一定的影响，毕竟数据较为庞大，一旦进程卡死，前面所爬取的数据就前功尽弃，所以这里我们使用了socket库的一个setdefailttimeout方法，我们可以设置时间，当HTTP或者Socket获取时间超时时，自动跳转到下一次操作，防止爬虫某个页面的时候，时间过长，导致程序卡死，自从使用了socket.setdefailttimeout()后，爬虫程序在运行的过程中没有出现卡死的情况了。

1. 城市信息获取过程

首先，我们需要爬取的是城市的游记数量多少，来证明马蜂窝旅游网站哪一座城市是热门旅游地点。



图 4.2.2.1 游记网站位置

这里我们可以看出，游记是位于社区里面的游记，进入之后拉到最下就有游记的总量，我们需要做的就是将这个总量爬取下来，所以我首先需要知道这个总量的位置是位于HTML标签哪里。

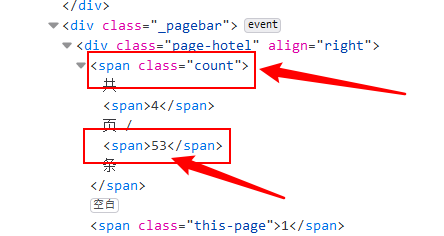


图 4.2.2.2 游记标签位置获取

这里我们可以看到，游记的总量是位于属性为count下的第二个标签里面，这里标签的位置是从0开始计算的，所以我们需要获取的就是下标为1的标签数据。这是我们爬取分析马蜂窝热门旅游城市的数据之一。

接着，我们要爬取的是城市的印象标签，爬取各种印象标签的总数，来对一座城市的景点、美食、娱乐购物进行数据爬取。



图 4.2.2.3 印象网站位置

我们可以通过马蜂窝网站的社区→行程里面得到印象的位置



图 4.2.2.4 印象标签位置获取以及分析

打开开发者工具我们可以看到，城市印象是位于属性为class=“m-box m-tags”标签的a标签里面的em标签里面，所以我们第一步是将em标签里面的文本内容爬取下来，这个时候我们发现了一个问题，就是a标签里面的href属性有所不同，例如胡同儿的href属性里面是jd，我们通过首字母可以猜测得出是景点，北京烤鸭的href属性里面是cy，我们通过首字母可以猜测得出是餐饮，所以这些城市印象是由不同的成分组成的，所以我们对这城市印象要进行处理，分清楚哪些是jd，哪些是cy，哪些是yl、gw，这里我们打算将娱乐与购物合并一块，如何将这些城市印象区分开呢，我们可以采用切片的方法，通过对比我们可以知道，无论是jd还是cy，它们都位于href属性内容的第2、3位，所以我们使用切片的方法，将第2、3位的内容切片出来，就能获得城市印象是由什么构成，这里我们打算分成四个部分，一个部分是城市的总体印象，将所有印象不分分类全部加起来的总和，第二部分是景点印象，第三部分是餐饮印象，最后的部分是娱乐购物印象。

在爬取完城市印象之后，出门旅游，到一定回去一些热门的景点看看，尝试下当地的美食，去购物放松下自己，所以下一步我们就是对城市的景点、美食以及娱乐购物进行数据爬取，看看哪些景点最热门，哪些美食最诱人，哪些地方是放松身心的好地方。



图 4.2.2.5 城市热门美食获取以及分析

这里，我们可以通过马蜂窝旅游网站的城市餐饮→餐饮，都得城市的美食排行榜，同样，我们打开开发者工具，我们可以发现，城市的美食排行榜中，美食名称数据是位于class属性为list-rank中的h3标签中，而美食热度是位于class属性为trend下的文本内容，所以这里很简单，只需要把所有h3标签的内容以及class属性为trend的美食热点爬取下来就行。

另外，娱乐模块的排行榜获取与分析与餐饮基本是一模一样的，只不过在URL中，娱乐的URL是yl，而餐饮是cy，但是数据的位置以及爬取都是一样的，代码也基本一致，所以这里就不对娱乐排行榜进行分析了。



图 4.2.2.6 城市热门景点获取以及分析

这里我们可以看到马蜂窝城市景点的信息位于景点界面，页面包含了城市热门景点的前五名以及点评，通过开发者工具我们可以发现，城市的热门景点信息以及点评数是位于class属性为row roe-top5下的h3标签里面，所以我们需要将h3标签里面的a标签的文本内容爬取下来，这样我们就能得到城市的景点名称，接着，我们要爬取的是点评数，这里点评数是位于class属性为rev-total下的文本内容，但是这里我们需要的只是数字，而不需要文字，所以我们可以 将标签中固定有的“条点评”替换成为空，再将数据保存下来，这样我们就得到了城市热门景点的数据。

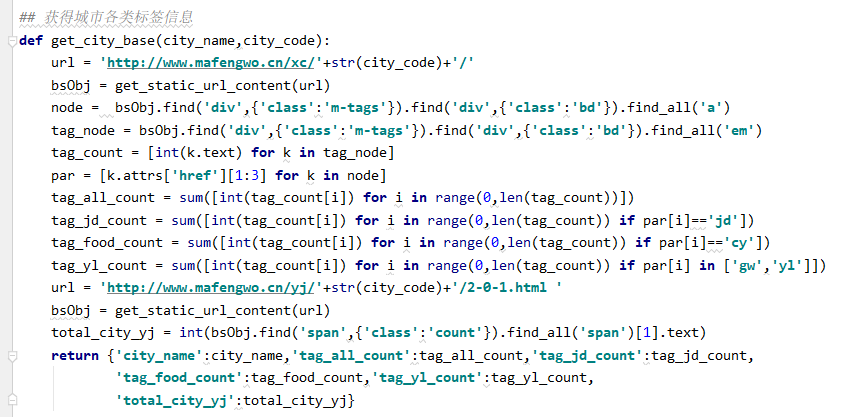


图 4.2.2.7 城市标签数据以及游记获取实现代码

这里代码实现的是城市各类印象标签的获取，这里我们定义了一个get\_city\_base函数，首先是url的传入，通过马蜂窝旅游网站的行程模块我们可以很简单的发现出url的规律，url的规律是：<http://www.mafengwo.cn/xc/>'+城市编号+'/'，这里的城市编号就是我们第一部分所爬取下来的数据，我们只需要将城市编号依次放入后，就能打开不同城市的行程界面，在行程界面下，根据我们之前的分析，使用find方法，找到class属性为m-tags标签，再找到该标签下class属性为bd里面的所有a标签以及em标签，定义一个tag\_count列表，将城市的所有em标签的数据放进去，然后对a标签的属性进行一个切片，切出第二、三位的有用数据，定义一个par列表，放置切片数据。接着，对tag\_count列表的数据进行相加，这就得到了城市的印象标签总数，对tag\_count列表进行遍历相加，这里遍历的过程我们需要设置条件，条件与我们切片的内容有关，分别为jd、cy和gw-yl，对应的分别是景点、餐饮和购物娱乐，这样就得到了城市的景点印象标签总数、餐饮印象标签总数和购物娱乐印象标签总数。接着我们爬取游记总数，游记是URL规律为<http://www.mafengwo.cn/yj/>'+城市编号+'/'，使用find方法，找到class属性为count下的span标签第二个文本内容，就得到了城市游记的总数，最后，将城市名称、城市印象标签总数、城市景点标签总数、城市餐饮标签总数、城市娱乐购物总数和城市游记总数返还。



图 4.2.2.8 城市小吃-景点-娱乐信息获取实现代码

这里实现的是城市小吃、景点、娱乐数据的爬取，第一个函数名为get\_city\_food，它是用来获取城市小吃排行榜里面的数据的，首先，还是url的导入，这里url的规律是http://www.mafengwo.cn/cy/' + 城市编号 + '/gonglve.html，还是使用find的方法，找到class属性为list-rank标签下的所有h3标签，这里爬取的是小吃的名称，由于爬取的只是小吃的名称，可能导致不知道是哪里的小吃，所以我特意在小吃名称前面加入了城市名称，以便区分，同样还是在class属性为list-rank标签下使用find\_all方法，找到所有class属性为trend的文本内容，这里爬取的是小吃的游记推荐数，最后将爬取得到的小吃名称，小吃游记推荐数通过一个二维数组返回。

同样，在城市娱乐中，城市娱乐与城市小吃所使用的爬取代码是一样的，只不过是url有所区别，城市娱乐的url规律是http://www.mafengwo.cn/yl/' + 城市编号 + '/gonglve.html，这里只需要将小吃代码的url改为城市娱乐的url就可以了，返回一个娱乐名称以及娱乐游记推荐数的二维数组。

对比小吃和娱乐，景点的代码还是有所不同的，同样，景点的函数我们命名为get\_city\_jd，景点的url规律为http://www.mafengwo.cn/jd/' + 城市编号 + '/gonglve.html，还是使用find的方法，找到class属性为row-top5标签下的所有h3标签，这里爬取的是景点的名称，由于爬取的只是景点的名称，可能导致不知道是哪里的景点，所以我特意在景点名称前面加入了城市名称，以便区分，使用find\_all方法，找到class属性为rev-total标签的文本内容，由于内容包含中文字符，我们是哦那个replace方法，将文字内容“条点评”替换为空，返回一个景点名称以及景点点评数的二维数组。

最后，我们对这些返回的二维数组导入到excel表中。

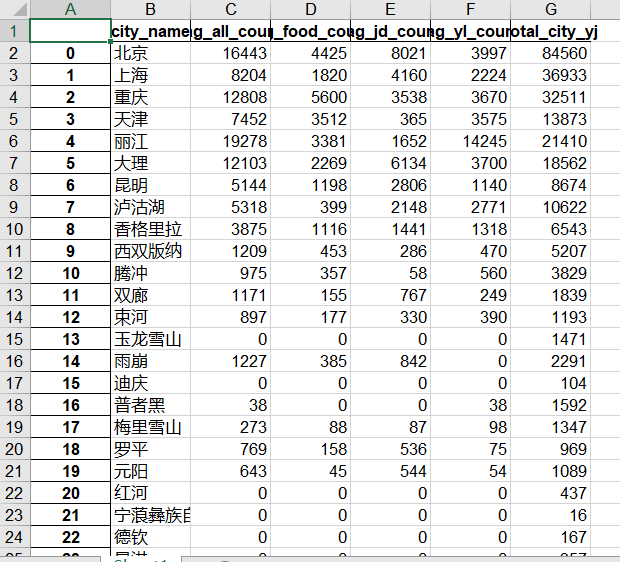


图 4.2.2.9 城市标签数据以及游记数据

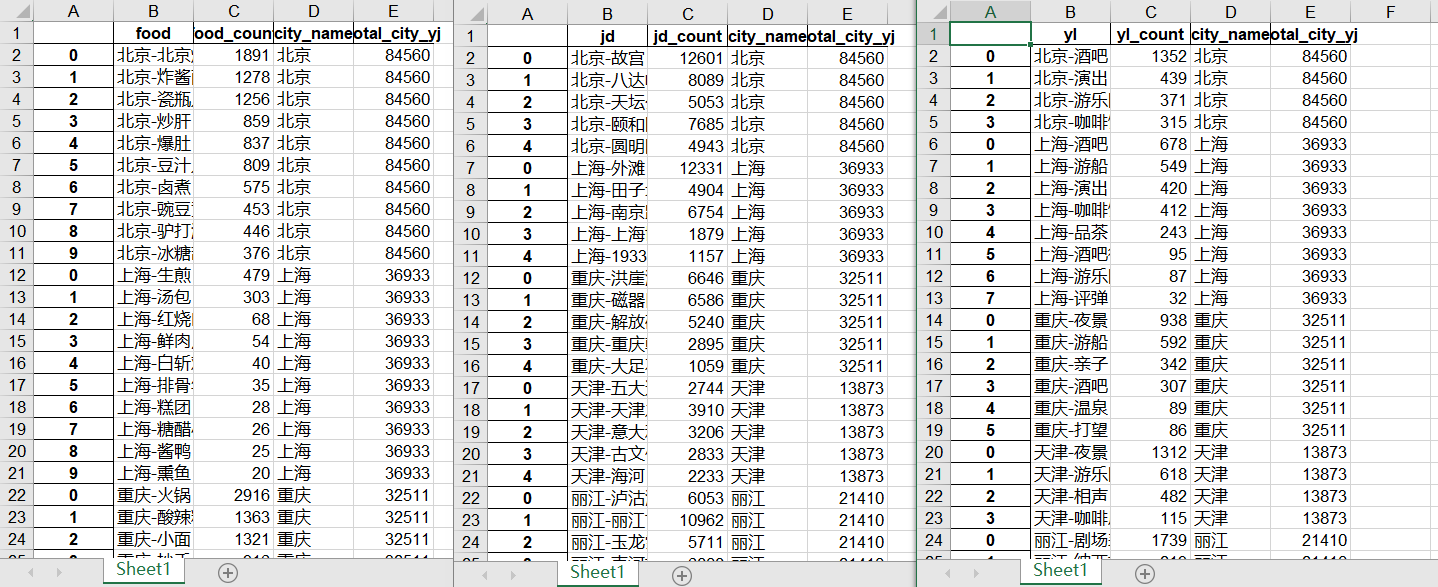


图 4.2.2.10 城市小吃-景点-娱乐信息数据

通过代码，这样我们就能得到四个存储数据的excel表格，最后我们就需要根据这些数据，进行可视化分析。

1. **数据可视化处理**
2. 数据可视化处理所使用的库

Pyecharts库，这里我们使用的Pyecharts来对数据进行可视化处理，Pyecharts用于图表的生成，它可以将数据以图表的形式生成，然后自己在本地生成一个网页用于保存图表信息，我们可视化的图表就是在这些生成的网页里面，Pyecharts里面包含了很多图表，这里我们可视化使用到的有bar，geo以及page。

1. 柱状图数据可视化代码实现



图 4.3.2.1 马蜂窝全国旅游游记-景点-餐饮-娱乐购物信息可视化代码

这里我们使用numpy库读取之前爬虫所保存的excel数据，对其中我们需要的数据进行了一个排序，毕竟我们是需要得出哪些数据是比较靠前来得出热门程度，可视化使用的是bar柱状图，xaxis表示的是x轴坐标，x轴坐标一共有十五个数据，数据来源是city\_base表中tag\_yl\_count排名前15位的城市名称，y轴坐标是city\_base表中tag\_yl\_count排名前15位的数据，还有就是柱状图名称，这里由于某些名称过长导致了x轴无法显示完全，所以这里我们对x轴数据使用了一个45°的倾斜表示，这样x轴数据就能显示全部，这里我一共写了四个bar方法，大致代码都是差不多的，所以这里只截图了一个bar方法出来，这四个bar的区别在于读取city\_base表后，对其进行排序时的数据是不同的，四个bar分别排序的数据分别为total\_city\_yj、tag\_jd\_count、tag\_food\_count、tag\_yl\_count，最后使用了一个page，将四个bar添加到page中，生成为一个命名为马蜂窝全国旅游游记-景点-餐饮-娱乐购物信息可视化的HTML页面中。

1. 饼状图可视化代码实现



图 4.3.3.1 景点人气排名-餐饮人气排名-娱乐购物人气排名信息可视化代码

这里是对城市的小吃、景点、娱乐进行可视化处理，由于这三部分代码构成也是差不多的，所以这里就拿娱乐模块来讲，首先是数据的导入，娱乐模块的数据我们是存储到了一个city\_yl表中，所以第一件事是读取，然后根据表中的yl\_count进行从大到小的排序，使用pie方法，首先设置长宽，接着添加数据，具体使用是[list(z) for z in zip(Faker.choose(), Faker.values())]

，Faker.choose(), Faker.values()就是我们需要添加的数据的列名来写入，接着就是一个饼状图的命名以及界面的简单排版，居中以及左边距离百分之一，顶部距离百分之十五，我们最后使用了一个page，将三个pie添加到page中，生成为一个命名为马蜂窝全国旅游景点人气排名-餐饮人气排名-娱乐购物人气排名信息饼状图可视化的HTML页面中。

1. 热力图数据可视化代码实现

图 4.3.4.1 马蜂窝全国旅游热力图TOP30可视化代码

这里我们同样先导入数据，这里导入的数据已经是提前排序好保存下来的了，然后根据city\_name和total\_city\_yj将前30的数据放置到data中，再使用geo方法，首先设置地图，为中国地图，数据为data，大小为10，颜色为蓝色，主题是热力图，命名为Geo-HeatMap,生成一个名为马蜂窝全国旅游热力图TOP30的HTML界面。

1. **可视化图片展示**

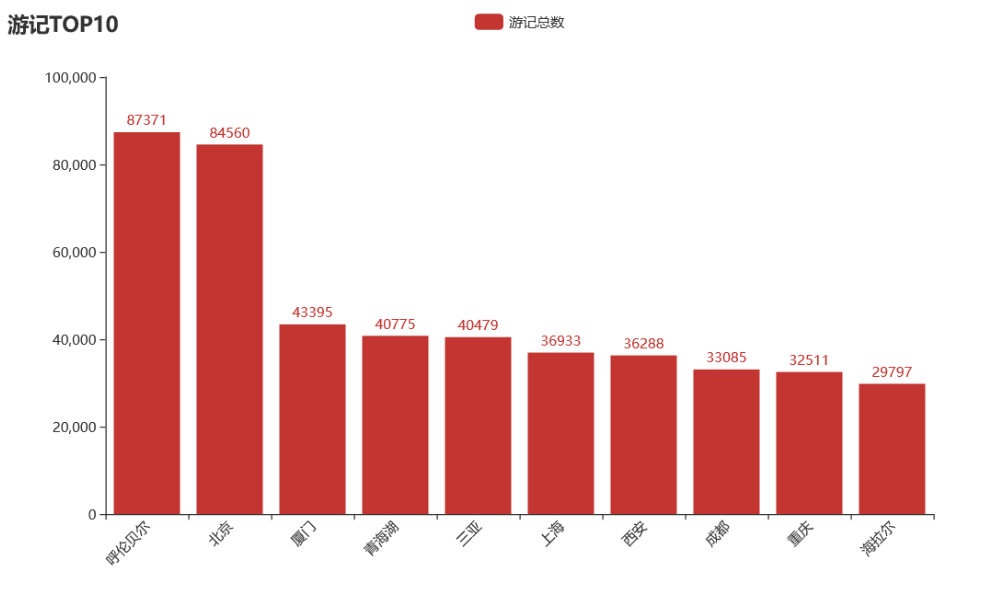


图 4.4.1 马蜂窝全国旅游游记TOP10

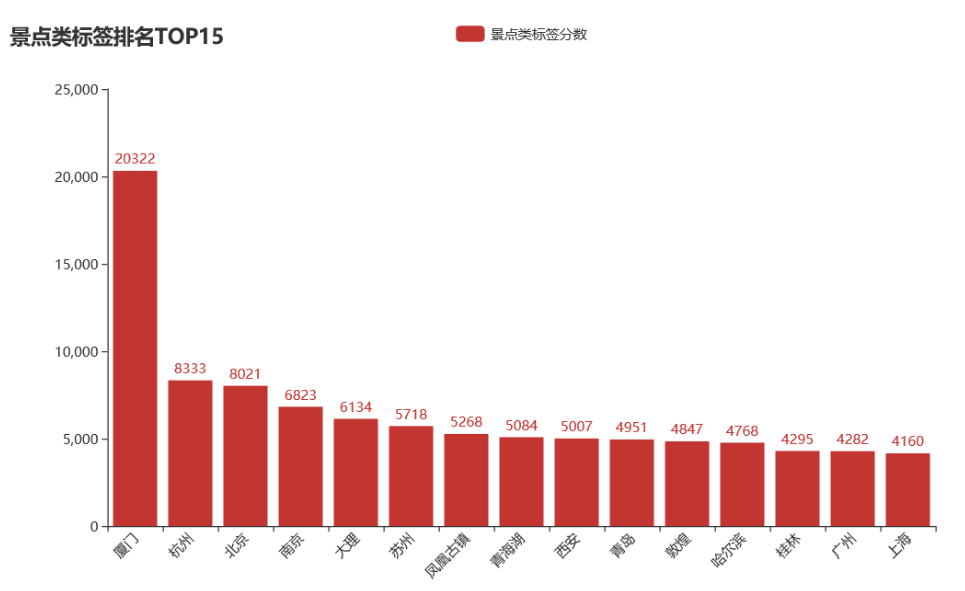


图 4.4.2 马蜂窝全国旅游景点类标签TOP15

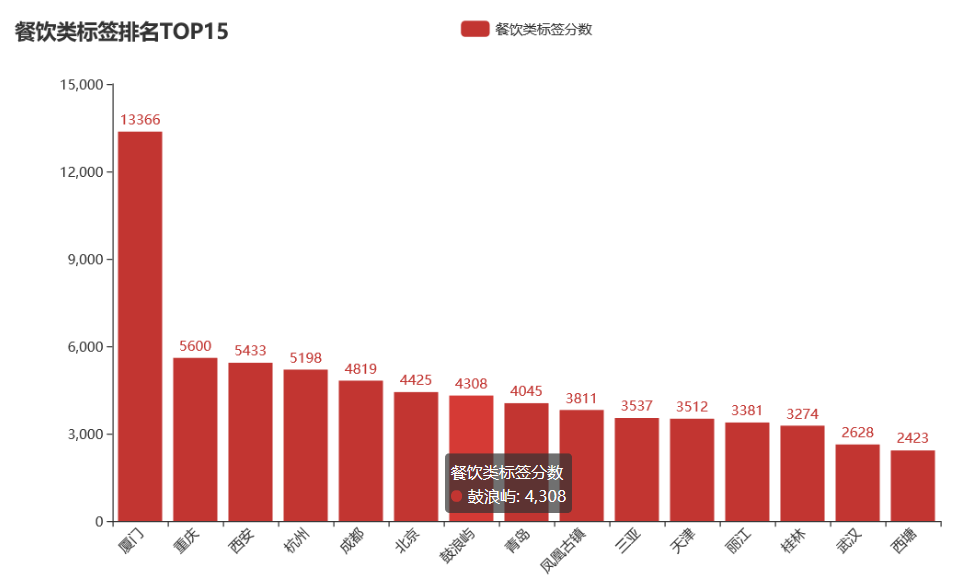


图 4.4.3 马蜂窝全国旅游餐饮类标签TOP15

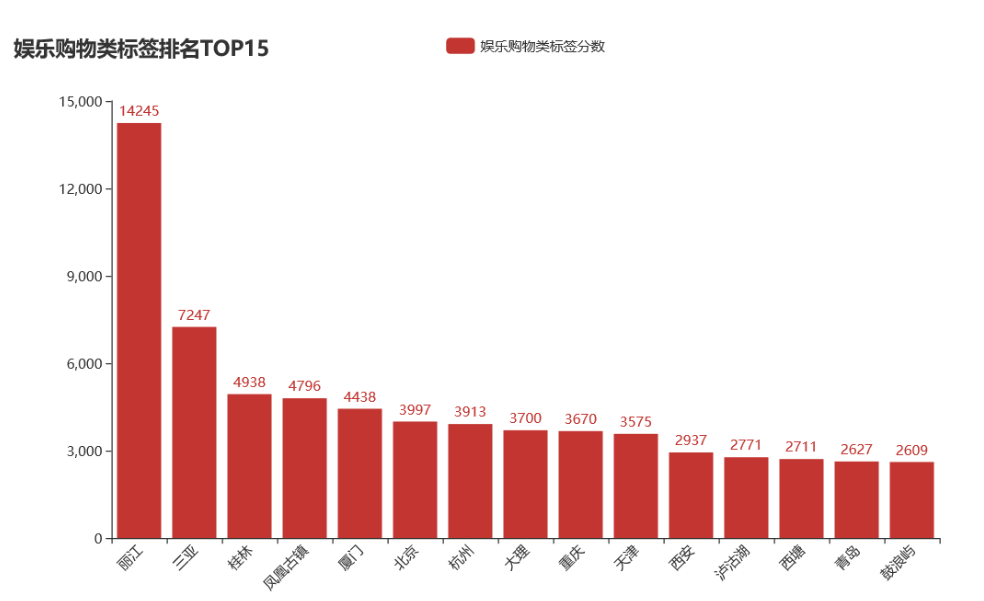


图 4.4.4 马蜂窝全国旅游购物娱乐类标签TOP15

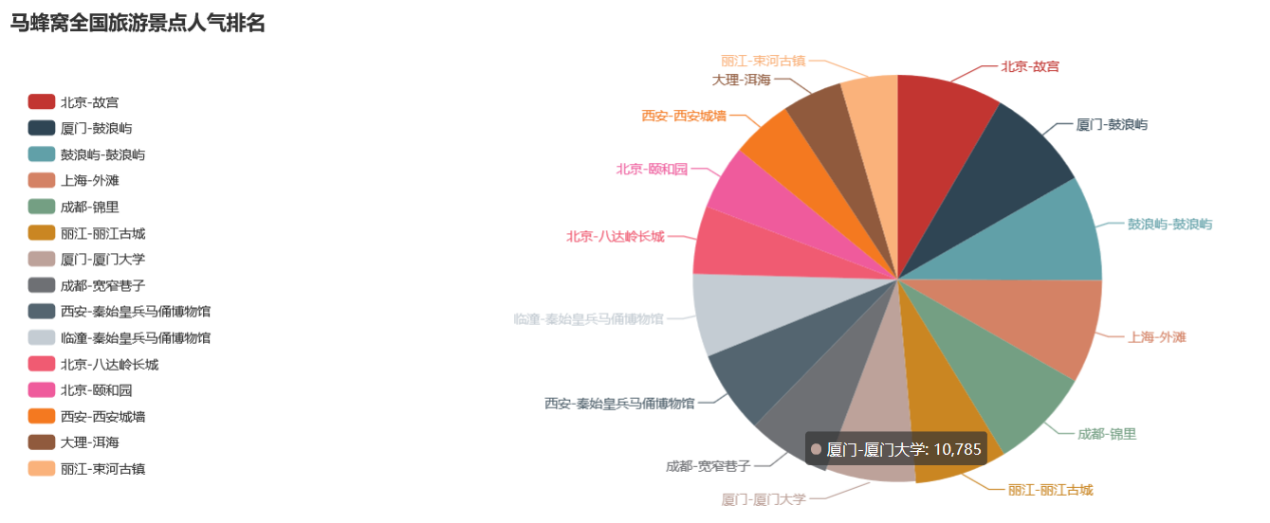


图 4.4.5 马蜂窝全国旅游景点人气排名TOP15

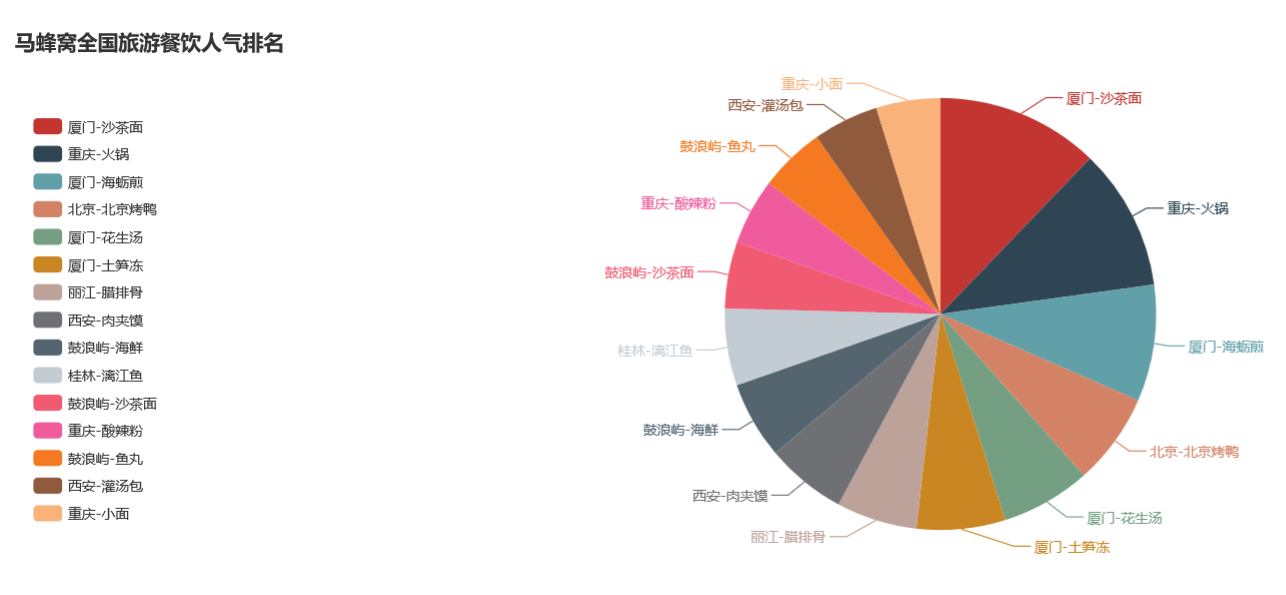


图 4.4.6 马蜂窝全国旅游餐饮人气排名TOP15

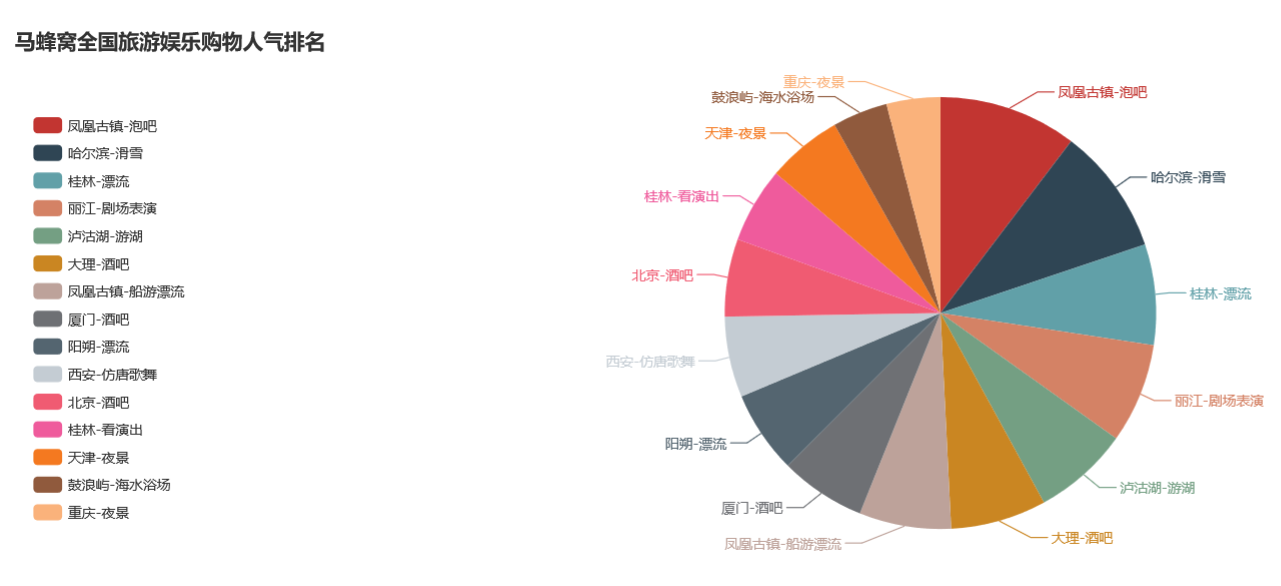


图 4.4.7 马蜂窝全国旅游娱乐购物人气排名TOP15

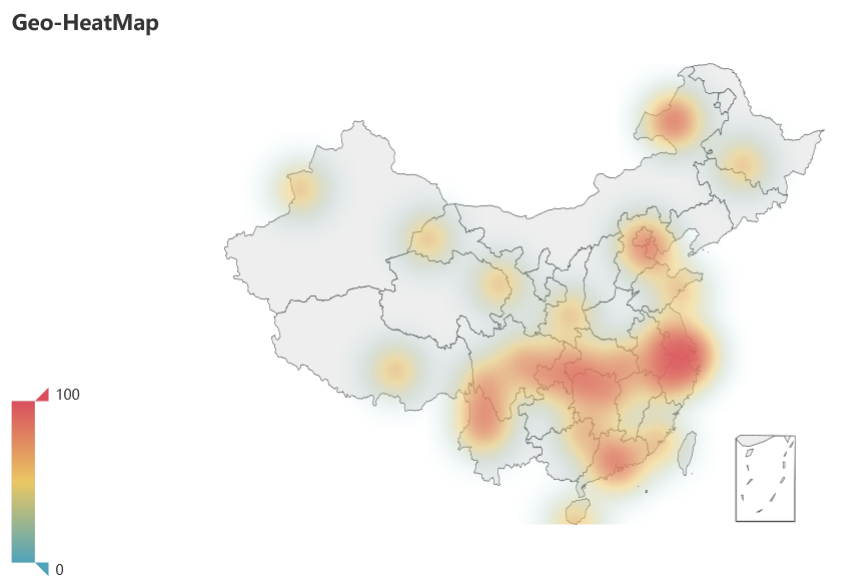


图 4.4.8 马蜂窝全国旅游热力图TOP30

通过可视化图片我们可以知道，在马蜂窝旅游中，呼伦贝尔是最多人去旅游的地方，呼伦贝尔经常被我们称之为大草原，呼伦贝尔位于内蒙古地区，在热力图显示中，我们看可以清楚的看到中国的北部地区有较深颜色的显示，呼伦贝尔也是一个避暑胜地，在夏天的季节，那儿的天气确实十分凉爽，十分适宜人们旅游避暑，骑着马儿欣赏大草原的风光。北京作为首都，当然也是很多人旅游的圣地，拥有众多的旅游景点，故宫、长城、颐和园等等，都是著名的世界遗产，想一睹中华民族辉煌历史的，北京是个不错的旅游地点。

接着，我们再来介绍下厦门这座旅游城市从数据中我们可以看到，景点类的城市最多人去的是厦门，餐饮类的城市最多人去的也是厦门，可见厦门也是许多人心目中的旅游胜地，首先是厦门的位置，厦门位于沿海地区，所以这里冬天天气温和，夏天没有酷暑，这对于旅游来说就是一个很不错的条件，不过由于沿海地区，大家旅游的时候，记得避开夏天的台风天气，其次，沿海的风景，十分令人向往，对于内陆地区人们没有到海边玩过的人，这也是一个很大的吸引点，接着，就是美食了，经济发展到现在，人们生活水平提高的同时，对于吃的要求也越来越高，不仅要吃的饱，还得吃得好，而恰恰好厦门的美食数不胜数，在餐饮TOP15的数据中我们就可以发现，厦门的美食就独占其六，沙茶面、海蛎煎、土笋冻、花生汤等等，无一不诱惑着吃货们的味蕾，最后就是价格方面了，厦门的旅游价格还是较低的，对比之前的呼伦贝尔大草原以及北京来说，厦门属于实惠的旅游地点。

在娱乐标签这快，丽江是最多人旅游的地方，小桥流水人家的风景以及白雪皑皑的玉龙雪山，都是很不错的旅游地点，但是由于近年来丽江酒吧女的事件，导致丽江旅游名声有了污点，所以丽江旅游时候，要辩真假，火眼金睛分清楚酒吧套路，别中招了。

最后，根据热力图的显示，我们可以清楚的看出南方地区以及沿海地区都是比较热门的旅游地点，看来还是很多人喜欢南方的美食，宜人的天气气候以及沿海的风光啊，根据这些数据，你有没有得出你想要去旅游的地方呢？

1. **不足之处**

在做热力图显示的时候，由于Pyecharts库中自带的城市键值对与我们在马蜂窝上爬取下来的城市名称会出现不匹配的现象，所以这里热力图显示只弄了TOP30的数据。

另外，我们爬虫是为了分析马蜂窝网站的旅游数据来解释说明旅游的好去处，但是由于今年的新冠病毒的缘故，对旅游行业的冲击十分巨大，很多旅游业都处于闭业状态，导致了马蜂窝网站旅游数据的没有实时更新，所以我们的数据来源只能建立在新冠病毒爆发之前的的数据，中间夹杂着几个月的空白期，也希望有旅游意向的人们，在新冠病毒完全控制下来之前，取消近期旅游项目，避免中招。当疫情结束的时候，再根据本文的数据分析以及热门城市，选择自己喜好的旅游目的地、景点、小吃或娱乐项目，再去放松自己，散散心。

1. **期望与总结**

这几个月来，面对毕业设计的构思与设计，通过不断的网上寻找资料，阅读有关书籍以及对参考文献的阅读理解，一步一步的走过来，终于是将毕业设计与论文成功的实现并完成了。

通过这次毕业设计，我又一次的感受到了python这门编程语言的魅力所在，它简单易懂的代码以及丰富的库给我留下了深刻的印象，让简单的操作能够发挥出复杂的作用，让人爱不释手。当然，在毕业设计实现的过程中，也遇到过很多的困难，有时候在寻找页面规则的时候，往往卡在那里好久，久久没有进展，让人无从下手，大大的减缓了毕业设计完成进度，这个时候，我的同学以及导师吴瑞然老师都会帮我指明方向，同学之间的互相讨论，不同的人有不同的思考方式，拥有不同的看法意见，大多时候能够帮助我换种方法去实现目的，让我受益匪浅。吴瑞然老师则会引导我如何去思考和解决这个困难，在这里我要感谢吴瑞然老师对我的帮助，感谢老师给予的资料参考以及建议。

这次的毕业设计也让我学到了很多之前不懂的知识，比如python库的运用，有些库是我第一次使用，让我的代码知识储量又一次增加了，也培养了我独立完成任务的能力，树立了自己的自信心。相信自己在以后的编程道路上，能够披荆斩棘，走得更远，学的更多。

**参考文献**

[1]郭丽蓉;;基于Python的网络爬虫程序设计[J];电子技术与软件工程;2017年23期

[2]刘杰;葛晓玢;闻顺杰;;基于Python的网络爬虫系统的设计与实现[J];信息与电脑(理论版);2018年12期

[3]唐琳;董依萌;何天宇;;基于Python的网络爬虫技术的关键性问题探索[J];电子世界;2018年14期

[4]彭智鑫;;基于Python的深度网络爬虫的设计与实现[J];信息记录材料;2018年08期

[5]潘巧智;张磊;;浅谈大数据环境下基于python的网络爬虫技术[J];网络安全技术与应用;2018年05期

[6]魏程程;;基于Python的数据信息爬虫技术[J];电子世界;2018年11期

[7]杨国志;江业峰;;基于python的聚焦网络爬虫数据采集系统设计与实现[J];科学技术创新;2018年27期

[8]夏天琦;;Python爬虫获取网络图片[J];电子世界;2018年10期

[9]郭二强;李博;;大数据环境下基于python的网络爬虫技术[J];计算机产品与流通;2017年12期

[10]涂辉;王锋;商庆伟;;Python3编程实现网络图片爬虫[J];电脑编程技巧与维护;2017年23期