**题 目** 基于Hadoop框架的

Python旅游大数据分析

**摘要：**

随着经疫情影响下的旅游业慢慢复苏，全国旅游景点都在紧锣密鼓的吸引游客前来游玩，本着为大众提供出行热门旅游地以及路线，了解游客的喜爱偏好、旅游目的地现状、旅游产品发展等旅游现状，为进一步为游客提供更好的服务、完善旅游目的地管理营销、创新旅游产品等提供方向和建议的目的。此份报告通过携程平台公开数据，利用Python爬取以全国主要省会、热门城市（例如北京、南京、青岛、济南、成都、大理等几十个城市）进行景点、评论风向等各种数据为基础，并将近百万数据储存在HDFS中，接着利用MapReduce对人们的各种出行目的地进行分析，并对景点值不值得去与评论之间建立情感分析，最后还建立了相对应的搜索引擎，并在最后，进行可视化数据展示，例如词云、热力图、雷达图等。

**关键词：**

Python爬虫；全国主要城市大数据；MapReduce数据分析； PyeChart；可视化页面； ZincSearch搜索引擎

目录

[一、引言 4](#_Toc105948423)

[二、相关技术与工具前提 6](#_Toc105948424)

[Python爬虫： 6](#_Toc105948425)

[HDFS： 6](#_Toc105948426)

[MapReduce： 7](#_Toc105948427)

[WordCloud： 7](#_Toc105948428)

[PyeCharts： 7](#_Toc105948429)

[SnowNLP： 7](#_Toc105948430)

[ZincSearch： 8](#_Toc105948431)

[HTML 8](#_Toc105948432)

[三、数据爬取 9](#_Toc105948433)

[3.1 爬取策略： 9](#_Toc105948434)

[3.2 爬取方式 9](#_Toc105948435)

[3.3 更新爬取数据集 12](#_Toc105948436)

[3.4 清洗数据 14](#_Toc105948437)

[四、 Hadoop处理： 15](#_Toc105948438)

[4.1 Docker搭建环境 15](#_Toc105948439)

[4.2 数据MapReduce处理： 15](#_Toc105948440)

[4.3 处理结果 16](#_Toc105948441)

[五、可视化 17](#_Toc105948442)

[5.1 ZincSearch搜索引擎： 17](#_Toc105948443)

[5.2 WordCloud生成： 18](#_Toc105948444)

[5.3 HeatMap与VisualMap 19](#_Toc105948445)

[5.4 SnowNLP 20](#_Toc105948446)

[六、总结 21](#_Toc105948447)

[参考文献： 22](#_Toc105948448)

# 一、引言

大数据主要有现状描述、预测分析、决策支持这3个方面的功能，将这3个功能与旅游情景相结合，学术界对于旅游大数据的应用也可以细化为以下3个层次[1][2]。

1.现状描述。李军轶等人通过构建社会感知计算模型，基于新浪微博提取西安国内游客的签到数据进行分析，结果表明游客的关系比较松散，并揭示了游客在西安的出游偏好与习惯，同时发现西安区域间存在不平衡现象[4]。高慧君等研究发现游客的情感与旅游目的地的其后舒适度呈正向相关关系，气温、风度、相对湿度及其不同组合对游客消极情感和游客综合情感会产生一定影响[4]。刘逸等通过赴澳中国游客发布的评论为素材进行分析，研究发现中国游客与国际游客的偏好存在差异。中国游客对目的地的地标性景观和多样化特色更为喜欢和重视。阮冬茹等对河北省文化旅游网站的内容进行抓取分析，研究发现河北省的旅游宣传与“文化”“历史”相关度较高，与河北丰富的历史文化底蕴相匹配[5]。

研究者通过历史数据、网络数据、GIS数据等资料进行分析，了解游客的喜爱偏好、旅游目的地现状、旅游产品发展等旅游现状，为进一步为游客提供更好的服务、完善旅游目的地管理营销、创新旅游产品等提供方向和建议。

2.预测分析。周效东等通过提前预测景区热度、游客饱和度构建景区安全预警机制，保证景区安全、正常运转[6]。张诗雨通过采集平台数据、实时数据、舆情数据预测景区客流量及相关情况，避免景区拥堵情况的出现，为景区的畅通提供保障[7]。黄先开等研究发现利用与北京故宫旅游相关的关键词的百度指数可以预测故宫当天的游客量[8]。

学者基于大数据进行的预测分析研究，主要集中在客流量的分析上，在旅游路线、旅游商品的推荐上也有所应用。研究方法主要有线性模拟、人工神经网络模型、网络分析等方法，研究模型选择上趋向于选择更加准确、更具有时效性的研究模型。未来的旅游预测应在更加丰富的互联网数据、物联网数据基础之上，在旅游的交通、预警、旅游产品等更多方面进行应用。

3.决策支持。大数据的决策支持是在现状解读、预测分析上的进一步应用。目前的文献中没有单独研究表明大数据在旅游决策中的支持作用，大多在对于现状和预测分析研究后，提到可以对景区、政府、游客的决策提供支持。而大数据技术在决策支持方面的应用较多，主要是目的地政府和景区利用旅游行业数据库进行分析，建立纵向和横向的维度进行分析建模，形成基于大数据的公共服务体系，提高游客满意度。并且在现在的智慧旅游研究中，旅游决策支持系统是其中的关键环节，是“智慧”所在，而大数据是其中的关键技术。部分学者对旅游决策支持系统进行了研究，王军伟提出了利用层次分析模型建立景区决策支持系统，帮助景区解决旅游景点优先开发方面的问题[9]。郑天翔等通过对游客个性化需求信息、景区设施的运营数据和其他游客的排队位置等环境信息的基础上进行实验模拟和比较分析，为游客提供更加个性化的游玩路线，从而实现分流，帮助景区解决拥堵问题[10]。

这些都是此次报告所进行分析的基础研究结果。

# 二、相关技术与工具前提

本报告总体以此流程进行：使用 Python爬取目标网站的元数据，以CSV的格式进行保存。将元数据上传到Hadoop平台中的HDFS分布式文件系统。之后，执行MapReduce程序完成数据分析，分析完成的数据存数据库。我主要负责以上部分。

接着可视化程序读取数据，基于PyeCharts进行数据可视化处理：

## Python爬虫：

一段自动抓取互联网信息的程序，从互联网上抓取对于我们有价值的信息。很多语言都可以进行爬虫，但我们使用更为简单的Python进行爬取。Python 爬虫架构主要由五个部分组成，分别是调度器、URL管理器、网页下载器、网页解析器、应用程序

调度器：相当于一台电脑的CPU。

URL管理器：包括待爬取的URL地址和已爬取的URL地址，防止重复抓取URL和循环抓取URL。

网页下载器：通过传入一个URL地址来下载网页，将网页转换成一个字符串，网页下载器有urllib2包括需要登录、代理、和cookie，requests。这里主要使用requests。后面都会用到。

网页解析器：将一个网页字符串进行解析，可以按照我们的要求来提取出我们有用的信息， html.parser（Python自带）、beautifulsoup（第三方插件），也可以使用lxml进行解析，相对于其他几种来说要强大一些。这里我们主要使用beautifulsoup4 以及lxml。后面都会用到。

应用程序：就是从网页中提取的有用数据组成的一个应用。

## HDFS：

HDFS是Hadoop项目的核心子项目之一，另一个是[MapReduce](#_MapReduce：)，是分布式计算中数据存储管理的基础，是基于流数据模式访问和处理超大文件的需求而开发的，可以运行于廉价的商用服务器上。它所具有的高容错、高可靠性、高可扩展性、高获得性、高吞吐率等特征为海量数据提供了不怕故障的存储，为超大数据集（Large Data Set）的应用处理带来了很多便利。

HDFS特点：高容错性、可构建在廉价机器上、适合批处理、适合大数据处理、流式文件访问

HDFS局限：不支持低延迟访问、不适合小文件存储、不支持并发写入、不支持修改

本次实验主要使用单机模式，并没有涉及分布式或伪分布式，而且仅仅是通过命令行或诸如WinSCP直接上传至其中，并在接下来的过程中使用MapReduce进行数据分析。这些之后都会用到。

## MapReduce：

MapReduce是Hadoop的另一个核心子项目有。是一种分布式计算框架 ，以一种可靠的，具有容错能力的方式并行地处理上TB级别的海量数据集。主要用于搜索领域，解决海量数据的计算问题。

MapReduce有两个阶段组成：Map和Reduce，用户只需实现map()和reduce()两个函数，即可实现分布式计算。本报告使用其对爬取的近百万数据集做数据处理。

## WordCloud：

可视化的一种常用手段，可以非常直观的统计文本中的词汇出现频率并以图形化形式展现。wordcloud 库把词云当作一个WordCloud对象可以根据文本中词语出现的频率等参数绘制词云、也绘制词云的形状、尺寸和颜色均可设定。

## PyeCharts：

Echarts是一个由百度开源的商业级数据图表，它是一个纯JavaScript的图表库，可以为用户提供直观生动，可交互，可高度个性化定制的数据可视化图表，赋予了用户对数据进行挖掘整合的能力。而当数据分析遇上数据可视化时，PyeCharts 诞生了。我们即可定义：PyeCharts库是一个用于生成 Echarts 图表的类库。

## SnowNLP：

SnowNLP是一个python写的类库，可以方便的处理中文文本内容，是受到了TextBlob的启发而写的，由于现在大部分的自然语言处理库基本都是针对英文的，于是写了一个方便处理中文的类库，并且和TextBlob不同的是，这里没有用NLTK，所有的算法都是自己实现的，并且自带了一些训练好的字典，而这也是我选择使用它的原因。

## ZincSearch：

ZincSearch 是一个可用于任何类型文本数据的搜索引擎。它可用于日志、指标、事件等。它允许您进行全文搜索等。例如将服务器日志发送到 ZincSearch，或者您可以推送应用程序数据并提供全文搜索，或者您可以使用 ZincSearch 在应用程序中构建搜索栏。而我之所以使用ZincSearch而不是solar 或者ElasticSearch主要是因为它非常轻量化，可以写个小脚本直接输入已生成的csv文件（甚至无需转成json），并且检索速度较快。

## HTML

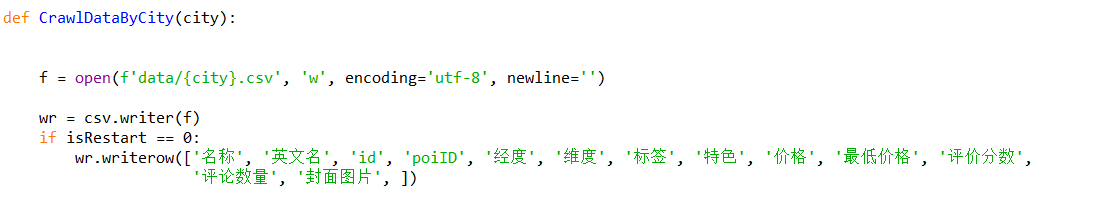
HTML的全称为超文本标记语言，是一种标记语言。它包括一系列标签．通过这些标签可以将网络上的文档格式统一，使分散的Internet资源连接为一个逻辑整体。HTML文本是由HTML命令组成的描述性文本，HTML命令可以说明文字，图形、动画、声音、表格、链接等。在本报告中，主要运用PyeCharts生成html页面，其生成可视化页面，

# 三、数据爬取

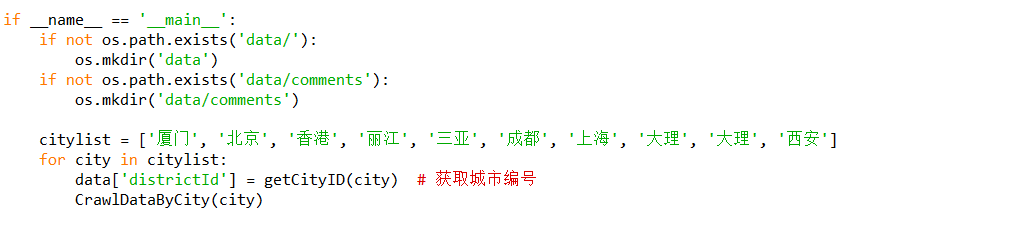
## 3.1 爬取策略：

数据爬取的策略是：在携程网站平台上，以省、市级的旅游信息咨询为起点，根据网页页面，按照经典排名连接层层抓取，每二十个景点为一页，直到该城市没有新的景点，并在每个景点数据内爬取如下数据：

## 3.2 爬取方式

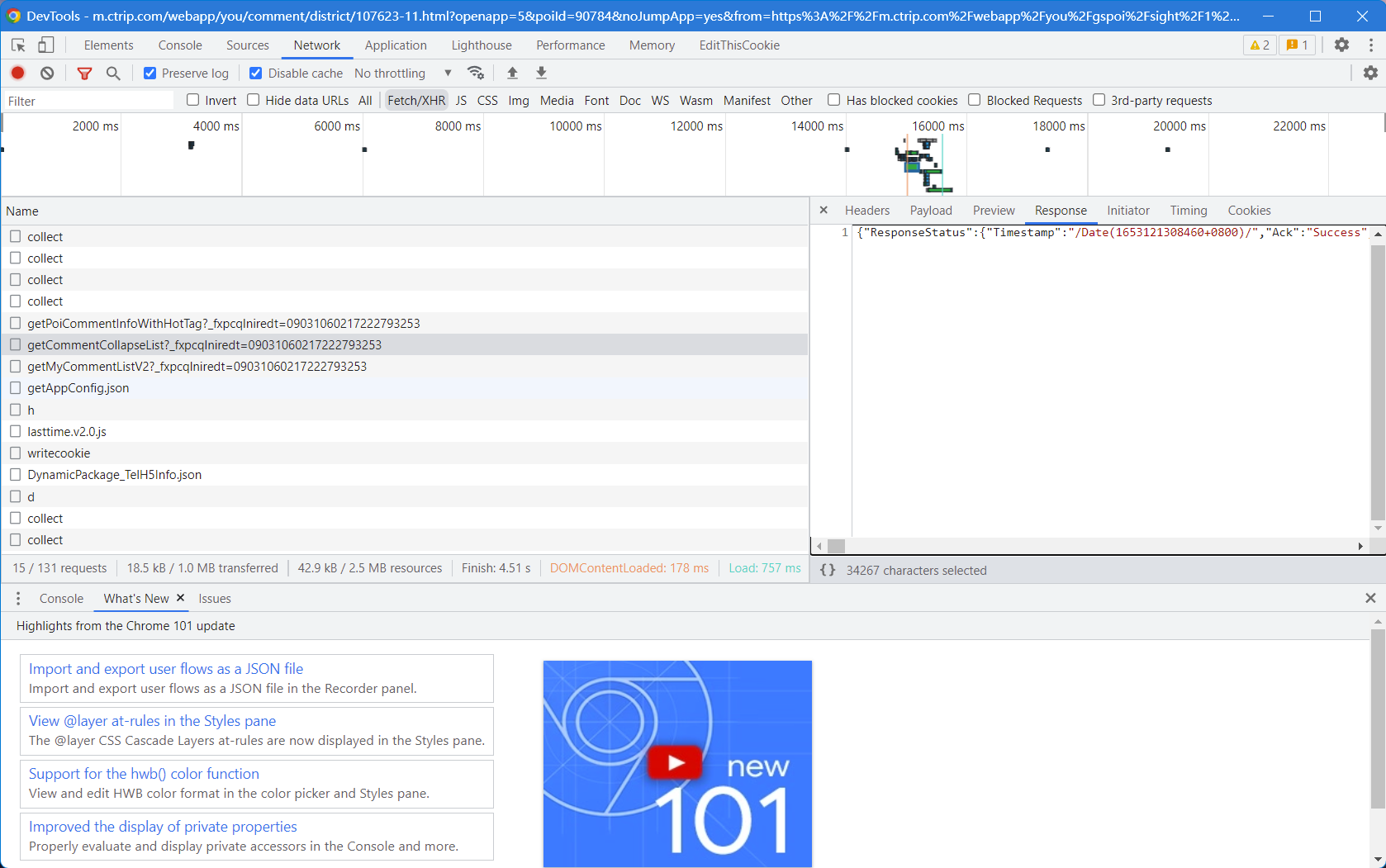


而爬取的城市暂且只有此九个（后面会有增加）：

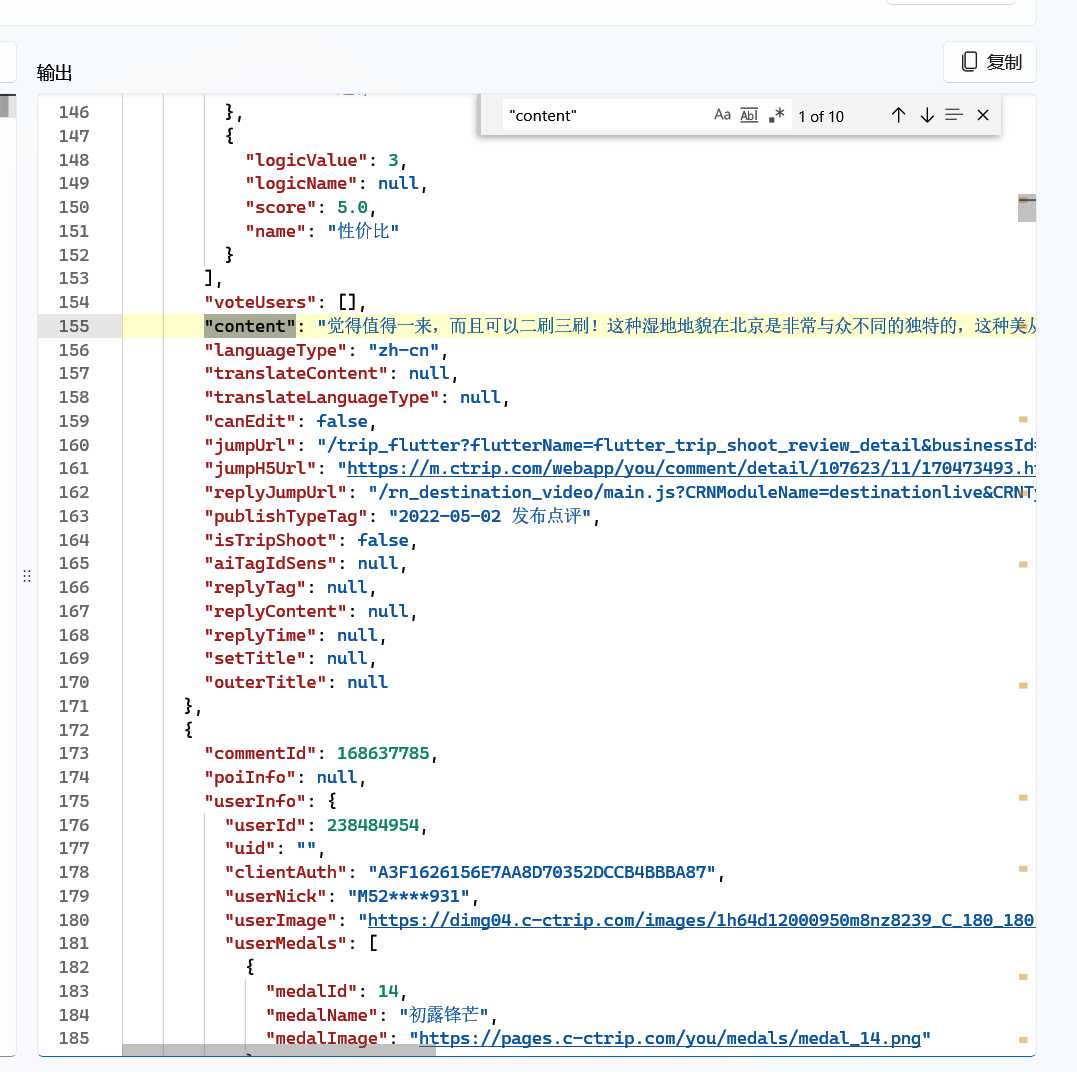


**（详细代码请看文档附的文件）**

而获取这些数据的方式是：在分析携程平台后发现，爬取网页的异步json数据作为数据源。而如果要处理大量 JavaScript 对象，那么 JSON 几乎肯定是一个好选择，这样就可以轻松地将数据转换为可以在请求中发送给服务器端程序的格式。进入招聘页面，打开Chrome dev tools，选中Network页面，重新刷新页面。从结果中找到传递数据的JSON文件，并分析其构成：

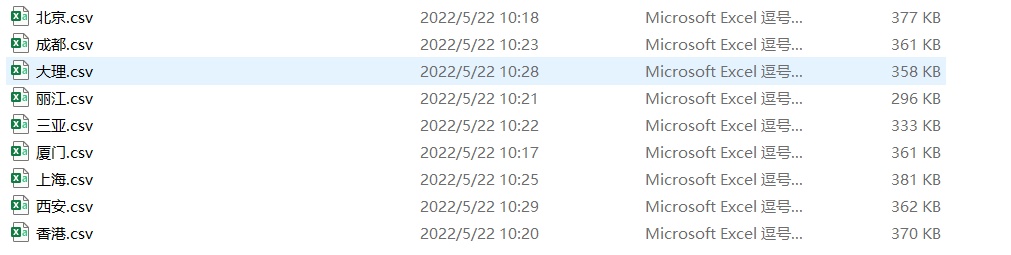


分析JSON文件中的各字段后，编写python爬虫程序，开始爬取数据：



成功实现后，对多关键词进行爬取，获得没有经过处理的原始数据。

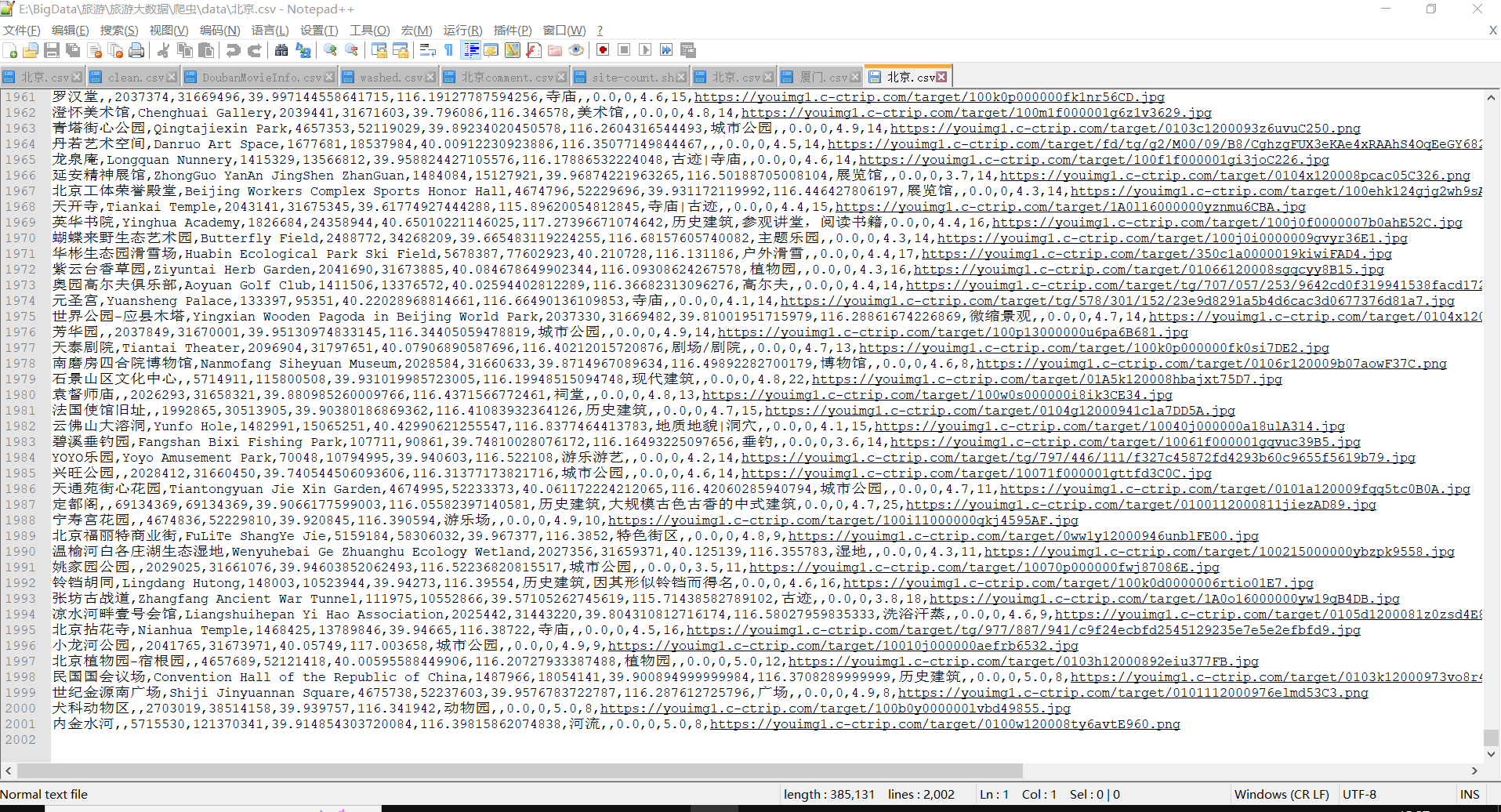
最终景区地点数据爬取如下：



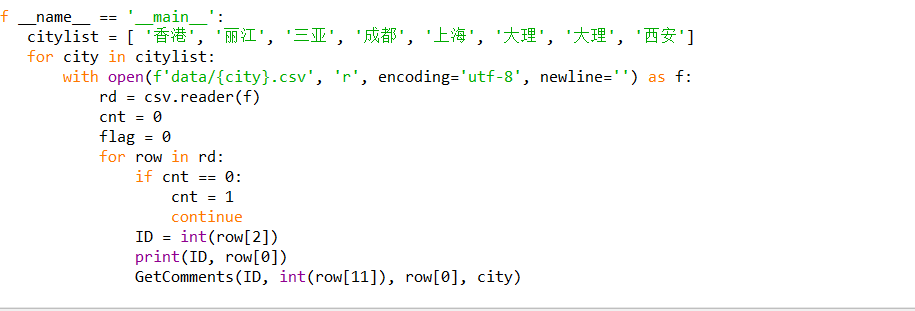
（当然后面会爬更多）



以北京为例，大约数据量为每条2k：

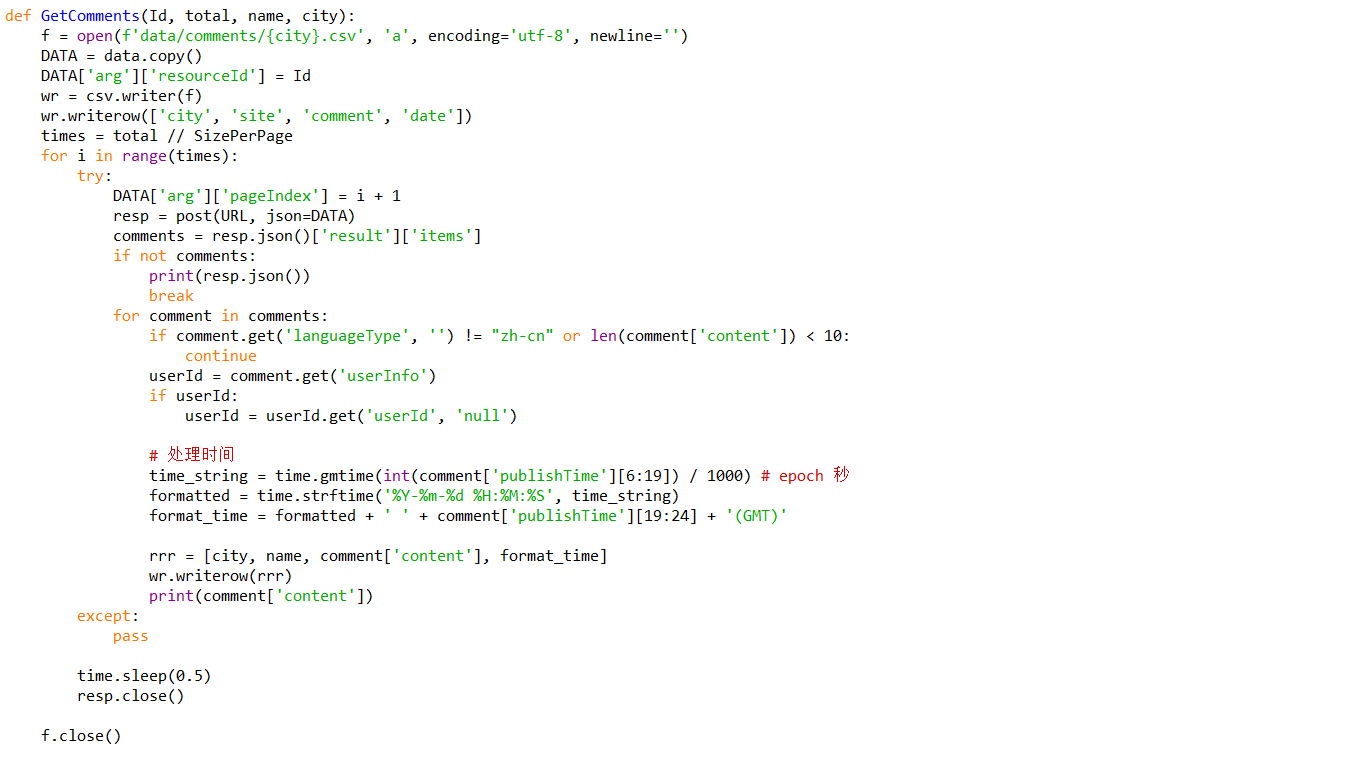


然后在读取的城市中读取每个景点的评论：

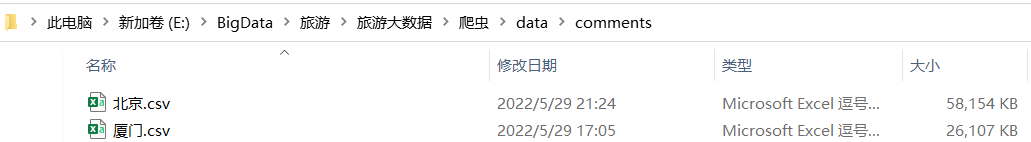


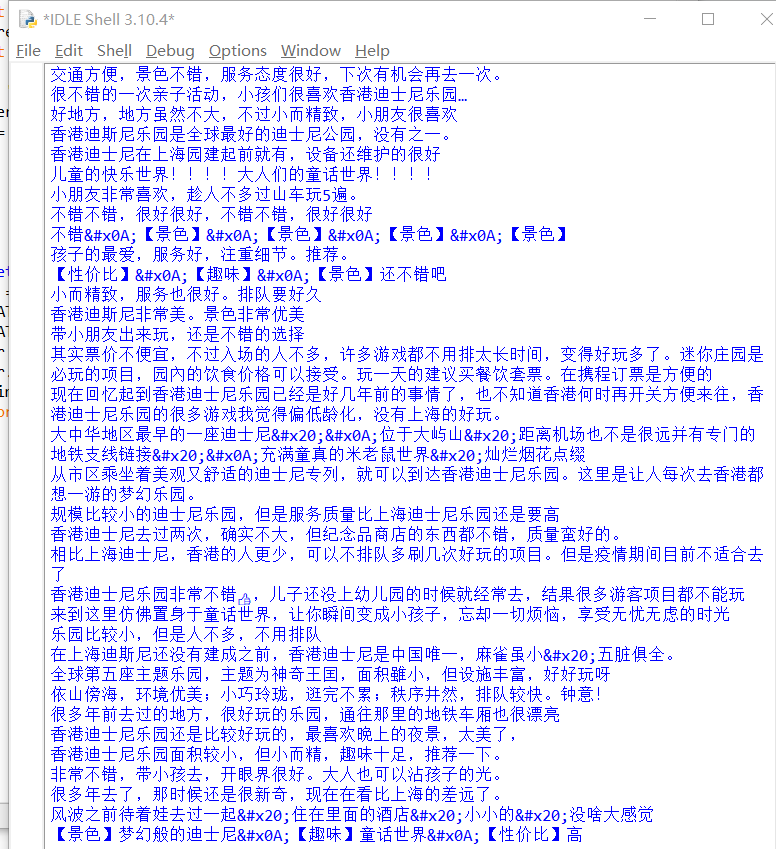
还需要在运行时注意的是不要弄错data文件夹的相对位置。

以及获取每个景点的Id、total、name、city：（这里的爬取速度真的让我操碎心，IP被封多次）

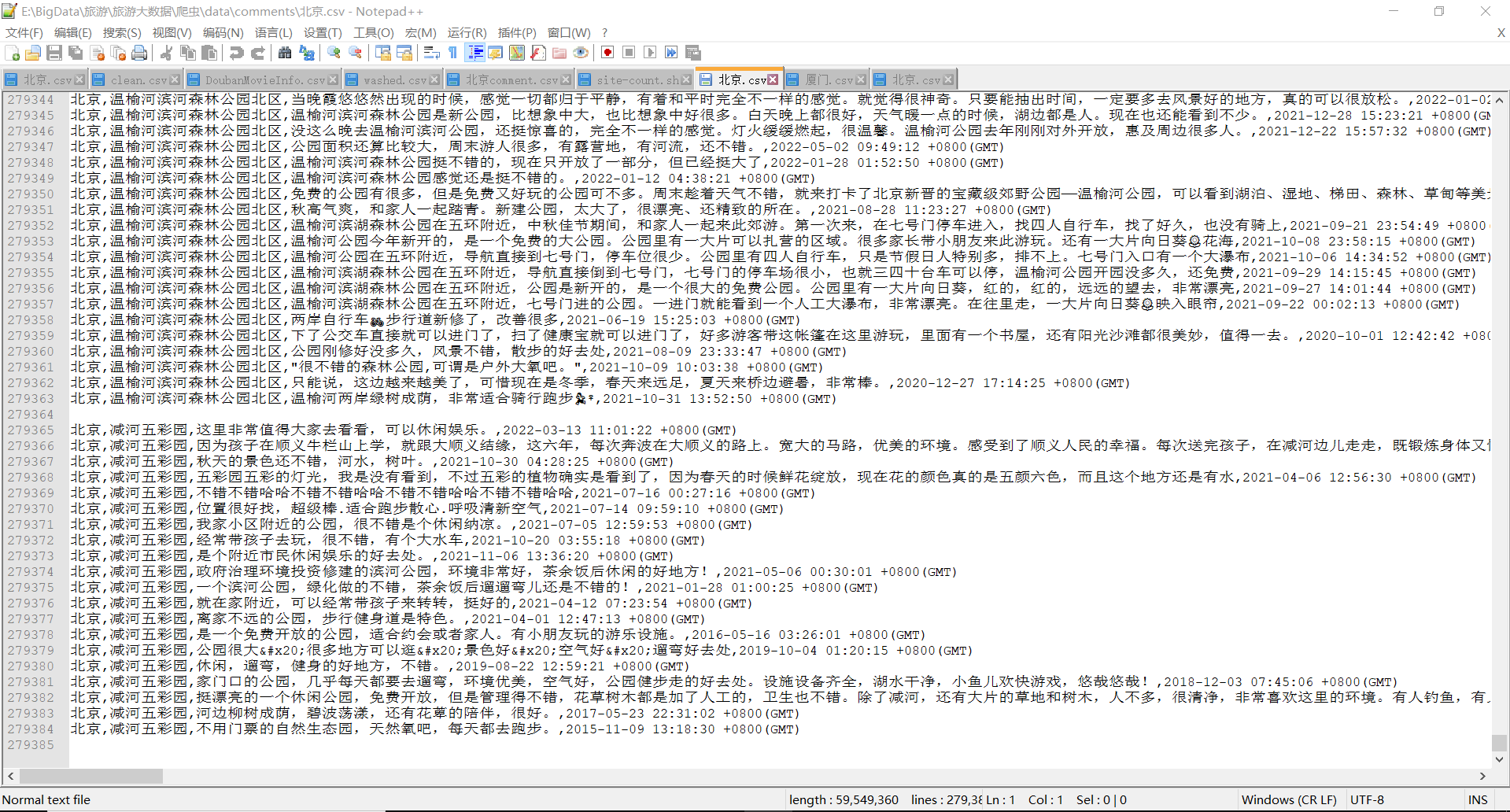


由于数据量过大，这里仅爬取两个城市的景点评论：

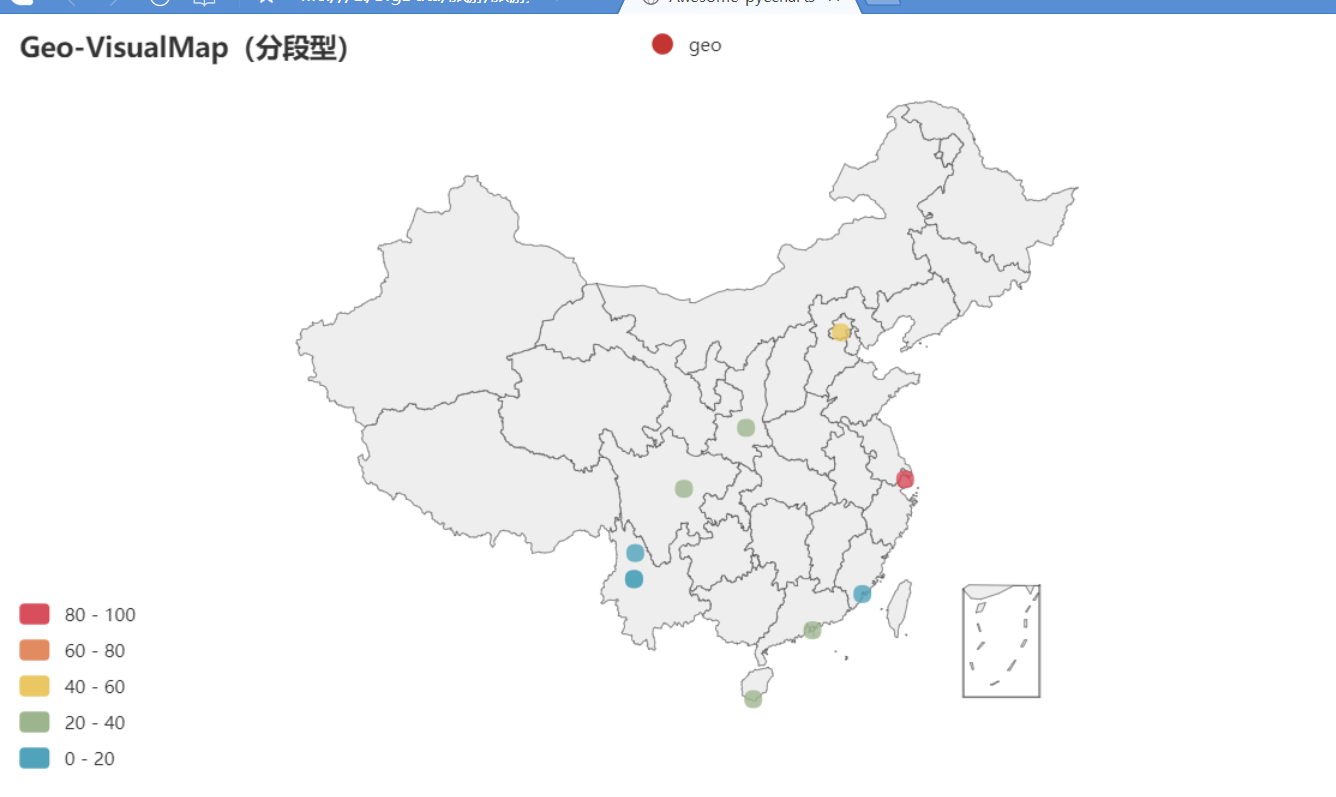




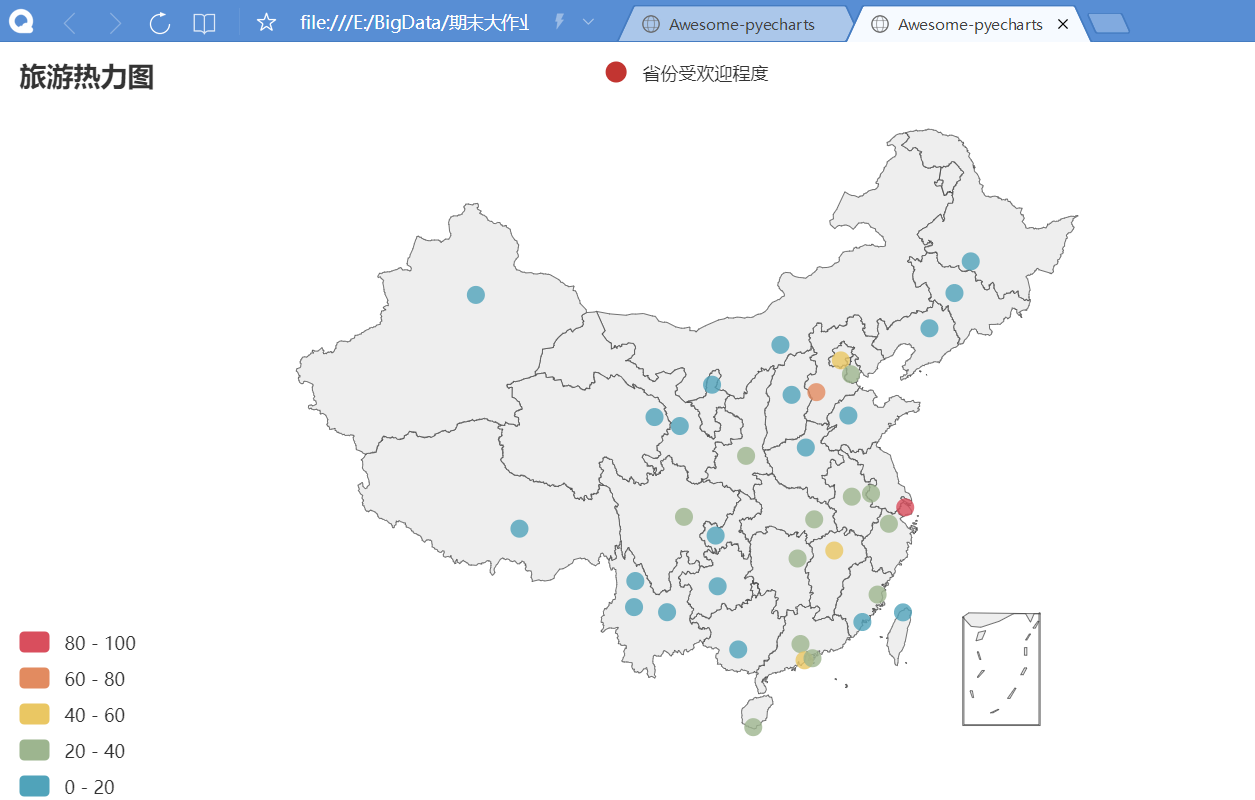
还是以北京的数据为例，大约有27w条：（注意这里是清理完的）



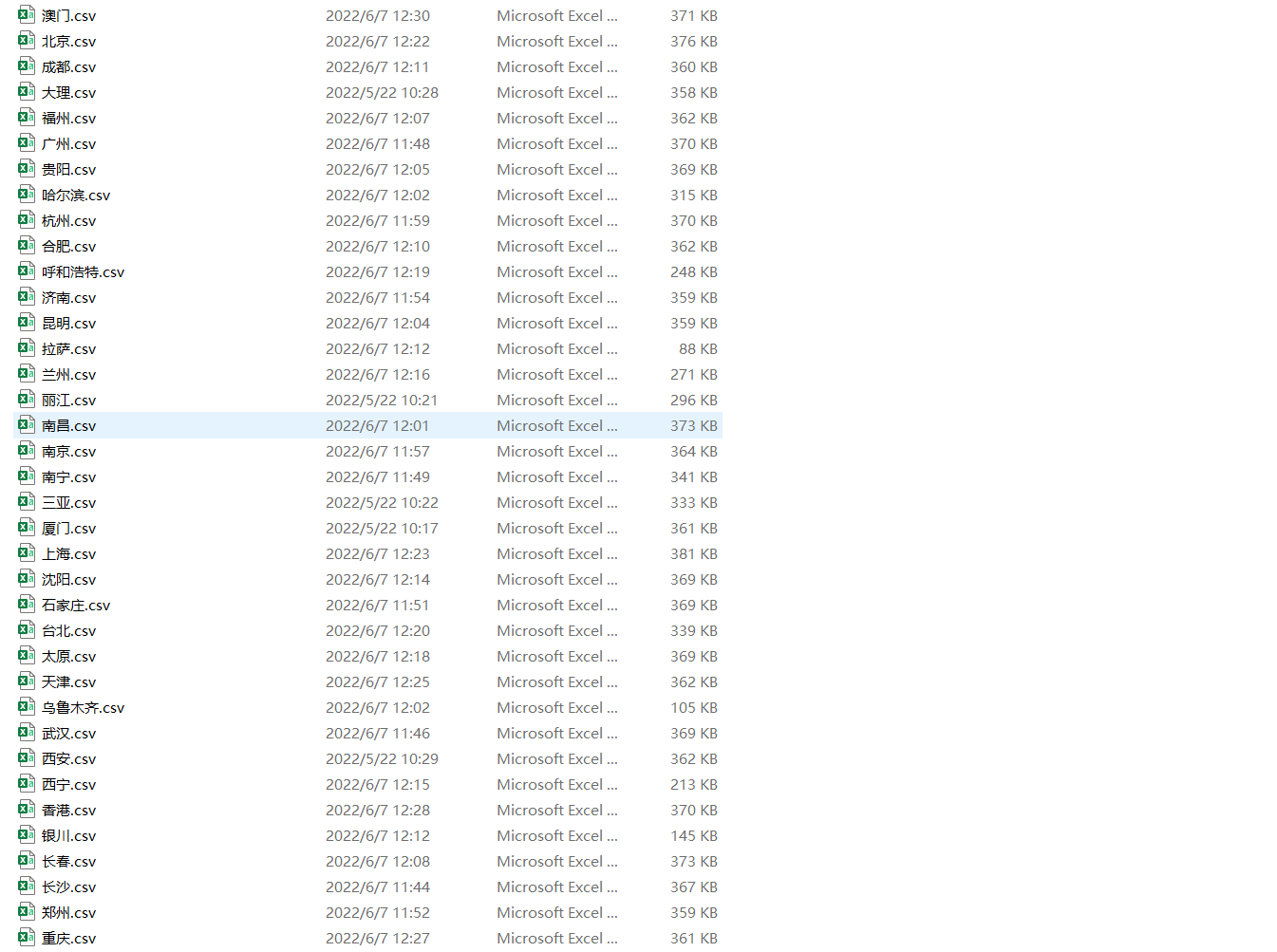
## 3.3 更新爬取数据集

而在后来的可视化过程中，在生成VisualMap的过程中突然觉得，需要爬取更多的城市以形成更好的Geo地图：

所以我决定爬取剩余主要城市经典数据并经过清洗后形成VisualMap：



清洗过的数据集如下：

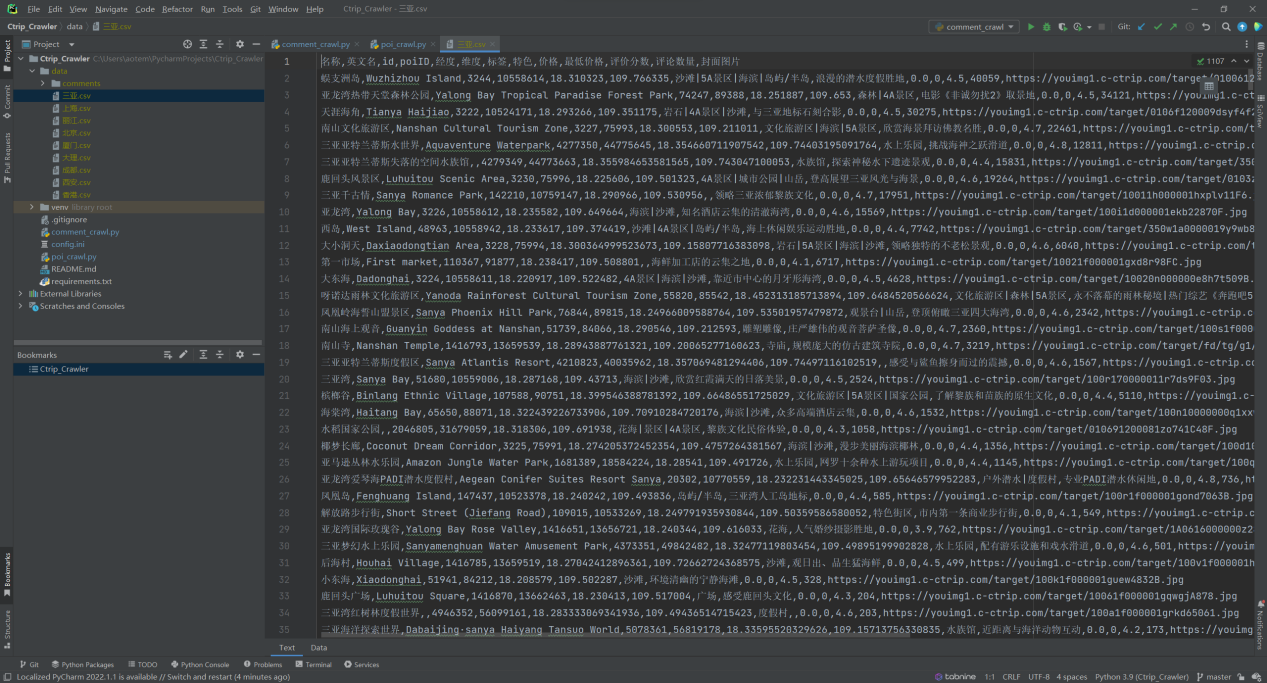


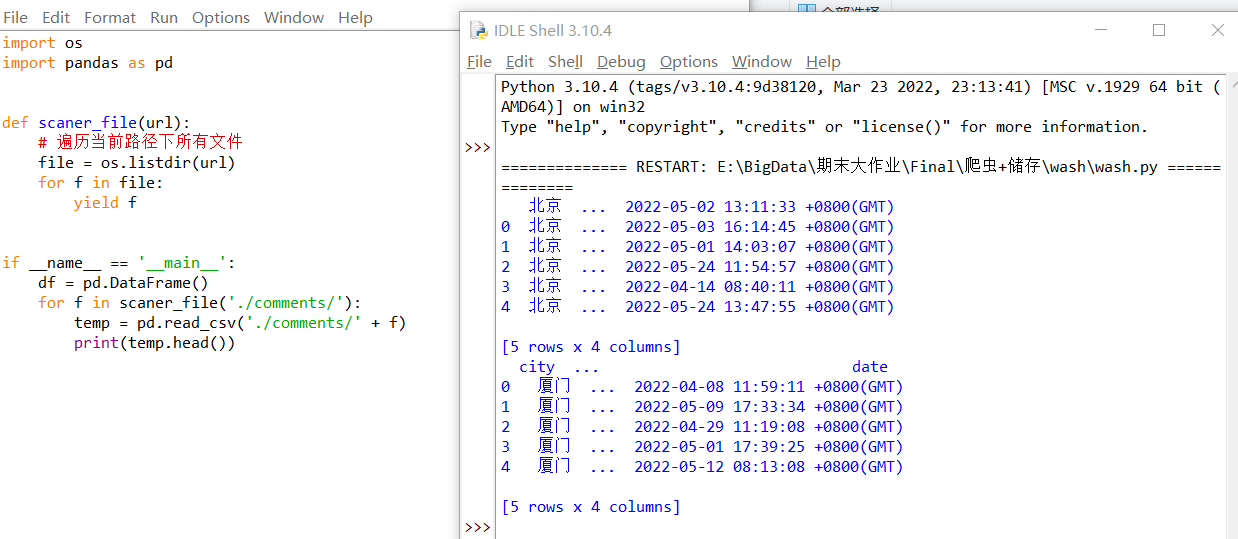
当然这些都是后话。

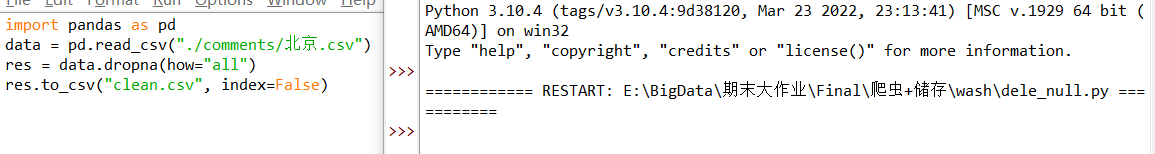
## 3.4 清洗数据

接着需要对数据进行初步处理，以便进行后续的操作。首先将无用字段删去，并将时间转换为正确的格式。

使用pandas库对csv文件进行数据清洗，得到处理后的数据，以便输入到Hadoop中进行操作：



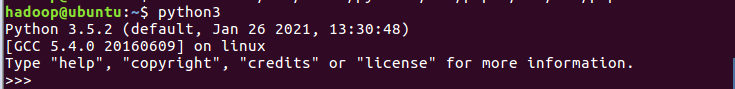
清洗：

删除空值：

进而得到清洁的clean.csv:



同样这些步骤也都可以在Hadoop中运行，只需要配置下：



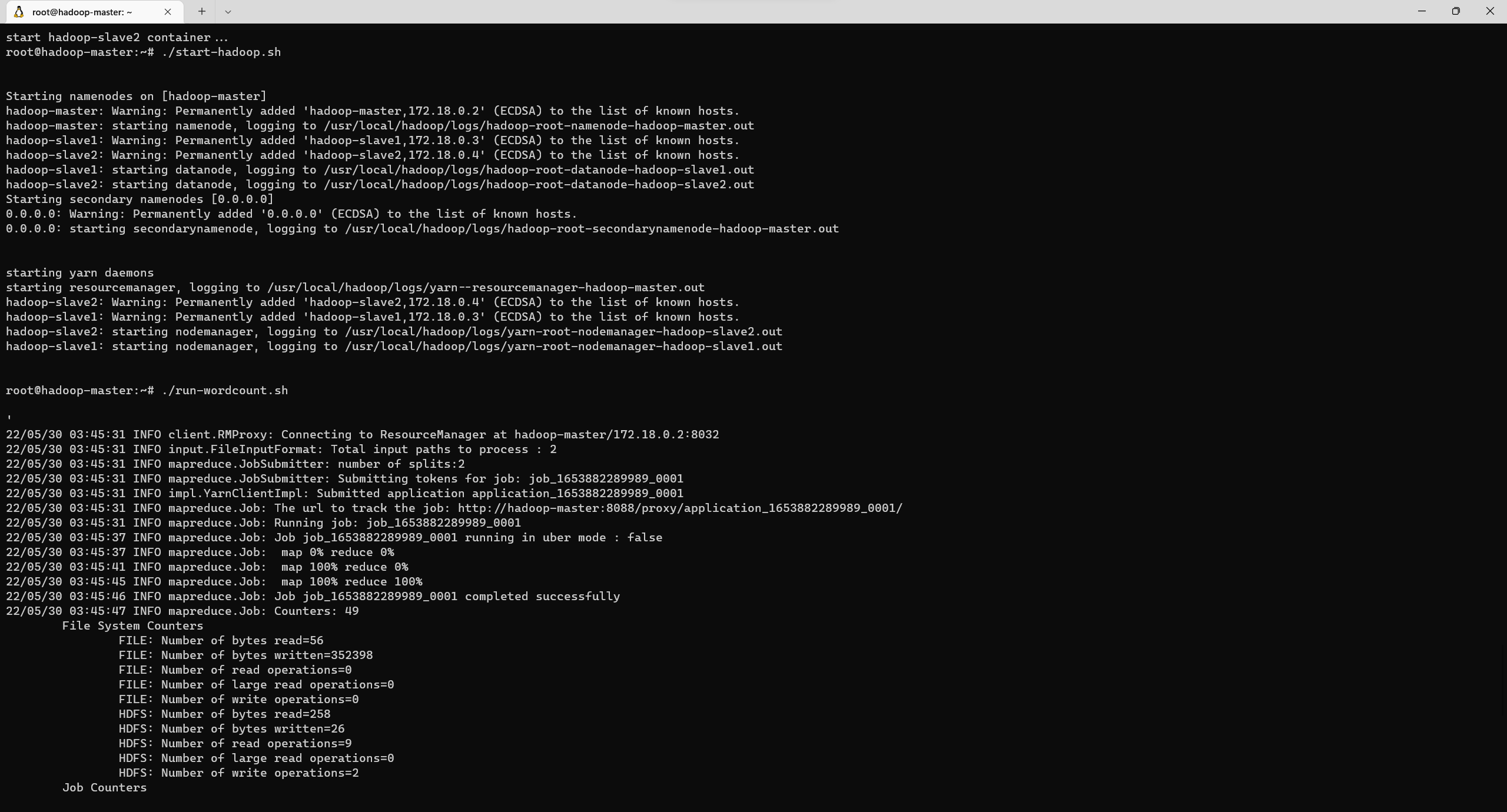
# 四、 Hadoop处理：

## 4.1 Docker搭建环境

这里我是用了别人搭建好的Docker,其中运行了诸多程序（这样快很多）：



然后移植到我的虚拟机上，进而将程序及csv文件打包，上传运行基本程序：



## 4.2 数据MapReduce处理：

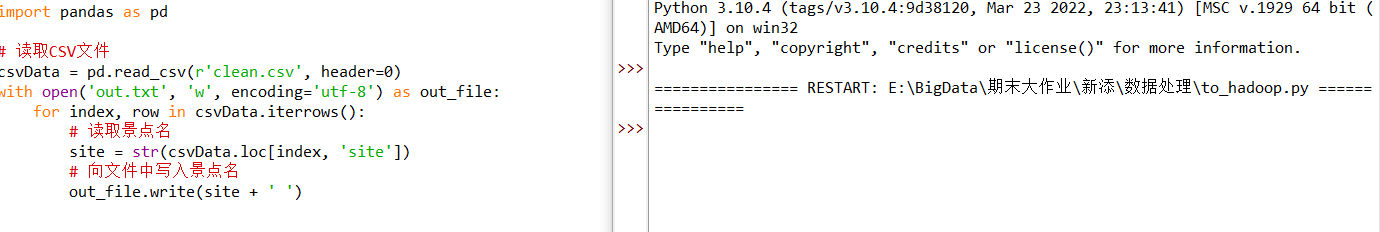
1. #!/bin/bash
3. # 计算出现次数

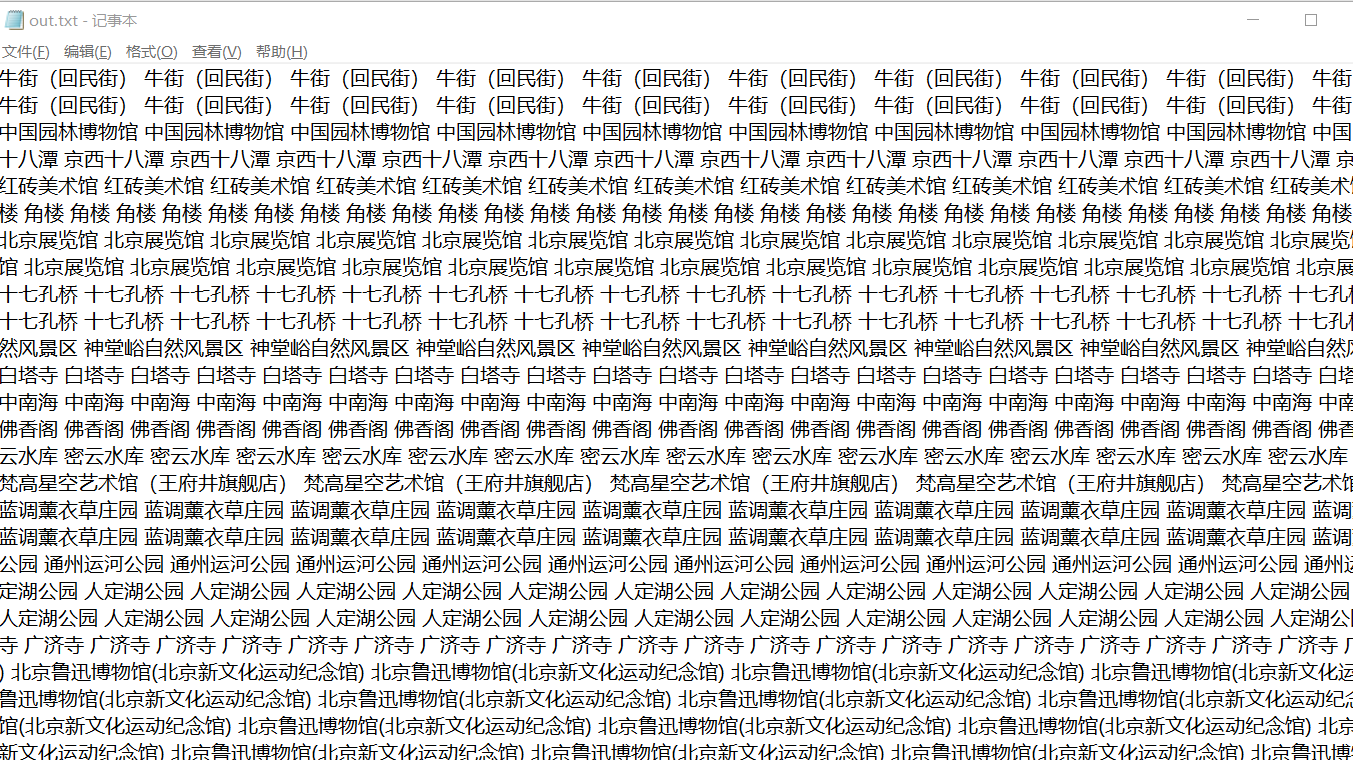
6. # create input directory on HDFS
7. hadoop fs -mkdir -p input
9. # put input files to HDFS
10. hdfs dfs -put ./input/\* input
12. # run wordcount
13. hadoop jar $HADOOP\_HOME/share/hadoop/mapreduce/sources/hadoop-mapreduce-examples-2.7.2-sources.jar org.apache.hadoop.examples.WordCount input output

16. # print the output of wordcount
17. echo -e "\nwordcount output:"
18. hdfs dfs -cat output/part-r-00000

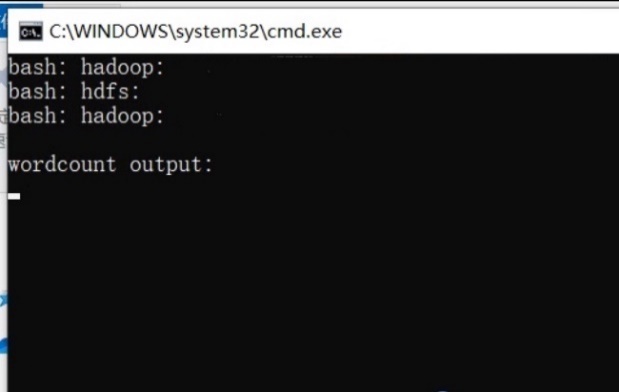
## 4.3 处理结果

运行生成只包含site的out.txt文件：





也可以直接运行：



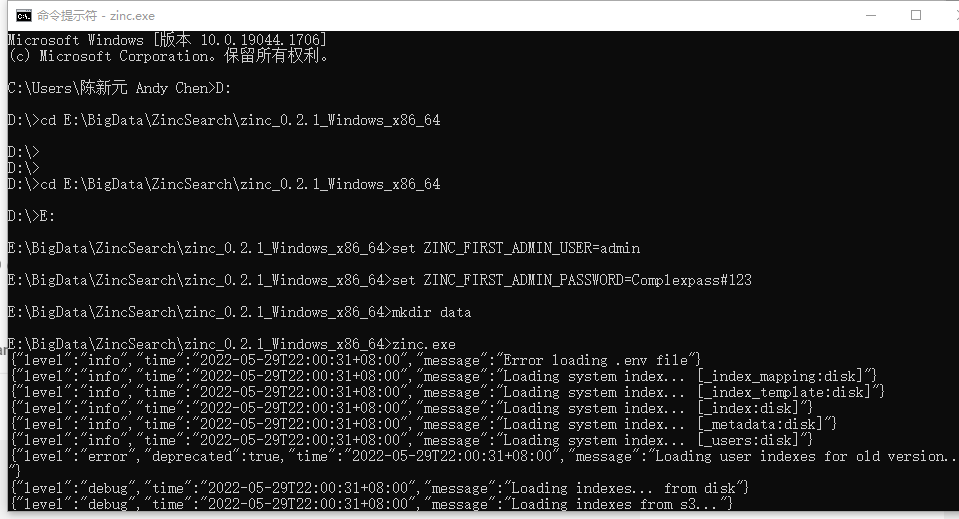
接下来我们也可以将数据从HDFS中调取出来，进而在Windows系统中进行一系列操作。此后将数据交给接下来的其他成员进行搭建搜索引擎、可视化等操作。

# 五、可视化

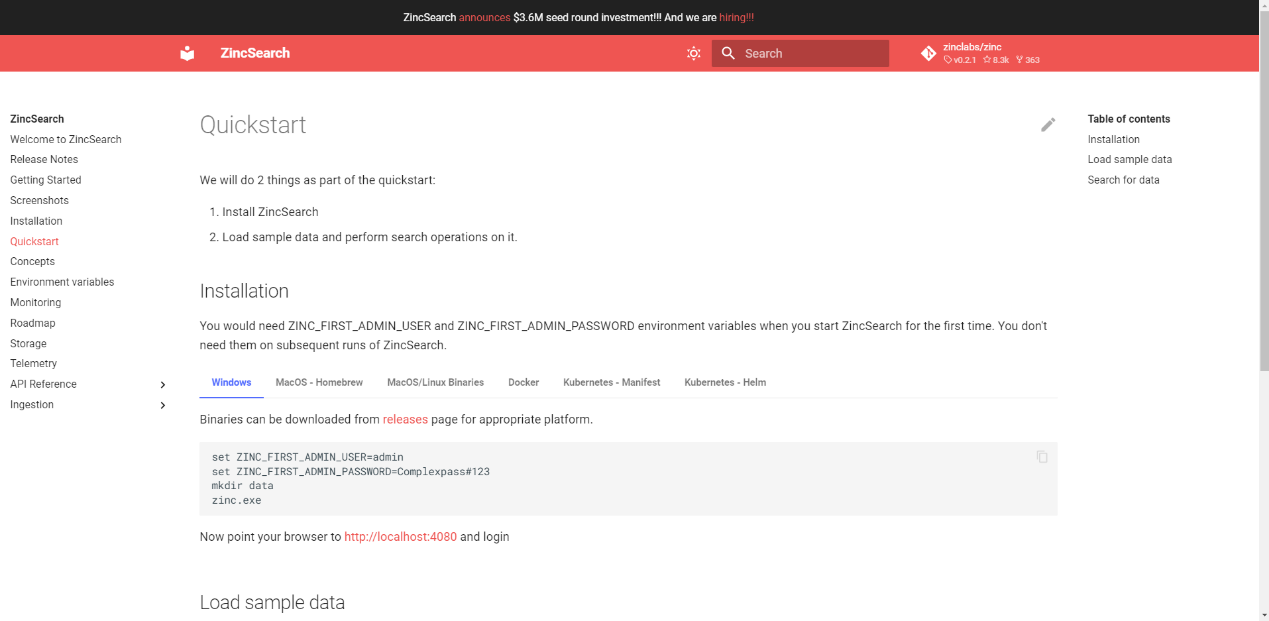
## 5.1 ZincSearch搜索引擎：

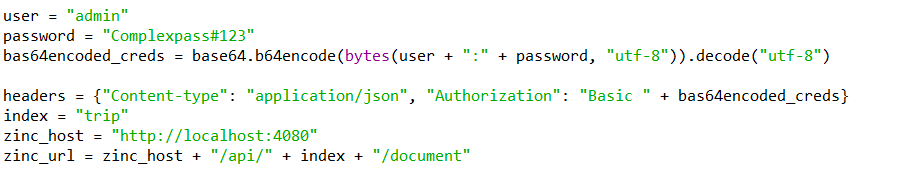
正如在第二部分：[ZincSearch：](#_ZincSearch：)提到的，之所以使用ZincSearch而不是solar 或者ElasticSearch主要是因为它非常轻量化，搭建非常简单，可以写个小脚本直接输入已生成的csv文件（甚至无需转成json），并且检索速度较快。

直接搭建：



通过默认账号、密码运行脚本输入数据：





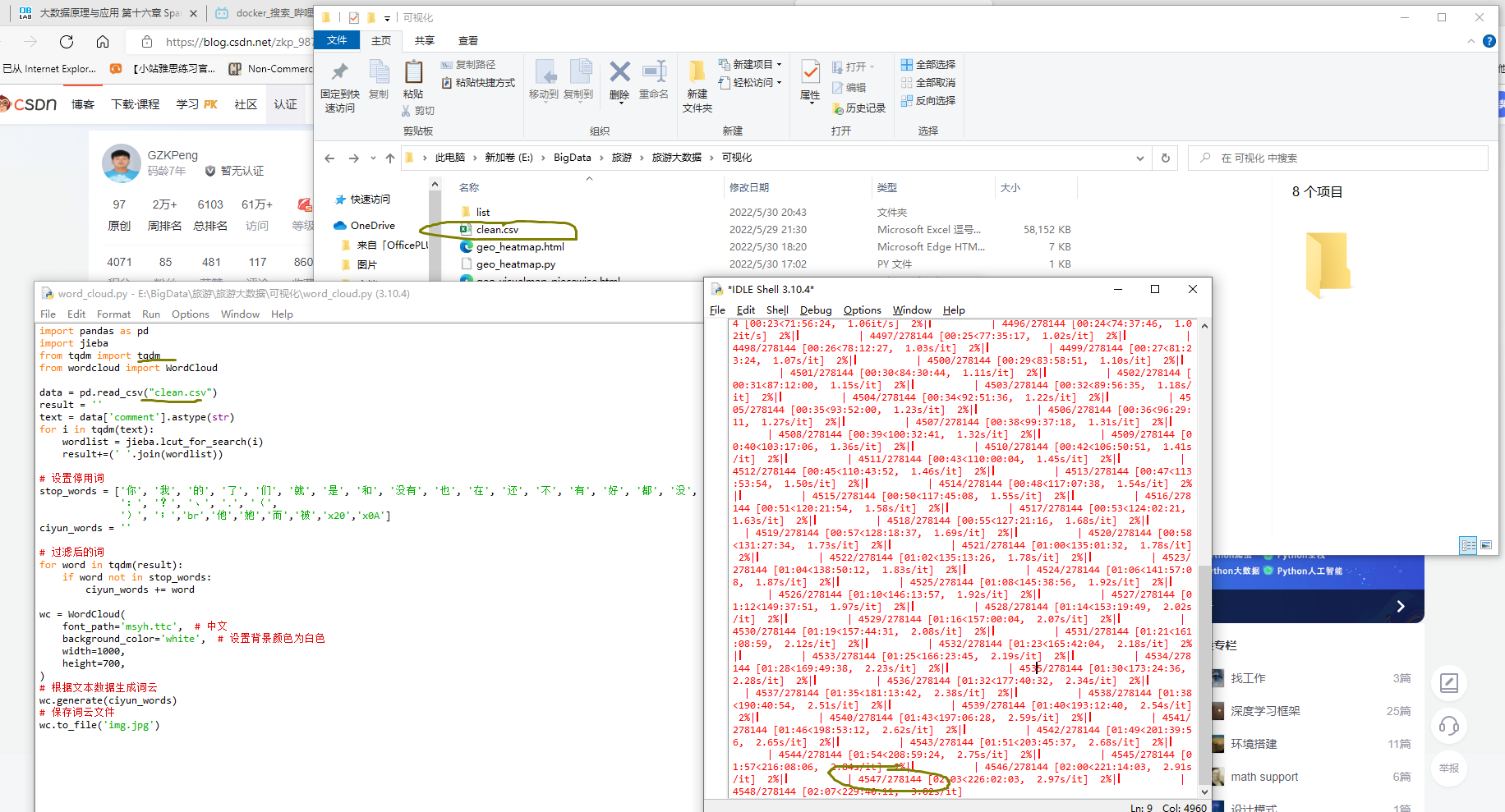


进而可以搜索文本信息：



## 5.2 WordCloud生成：

直接使用插件慢慢生成：



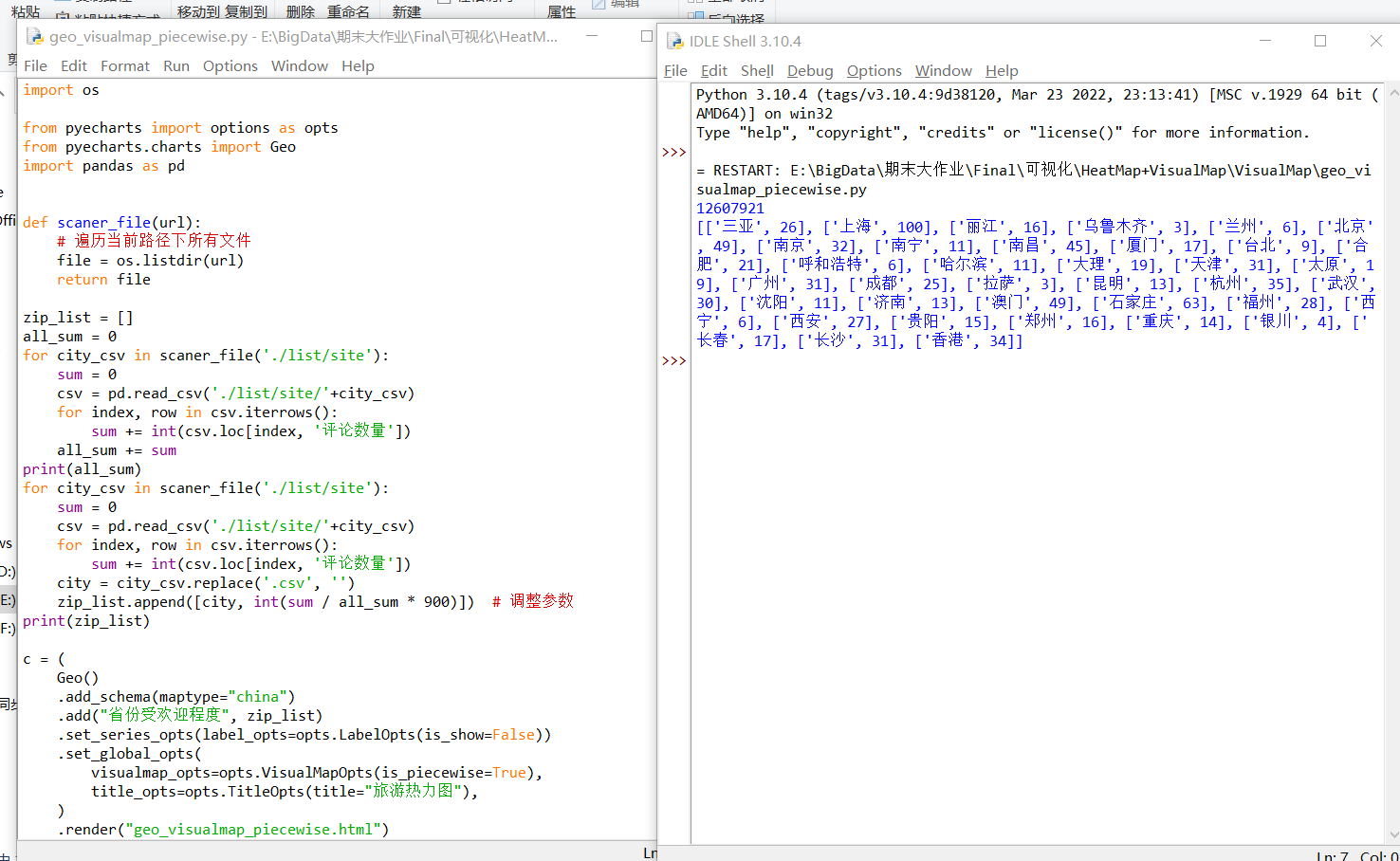
第一次生成包含没有转换中文字符导致的诸如X20、x0A等问题：转换后问题解决：

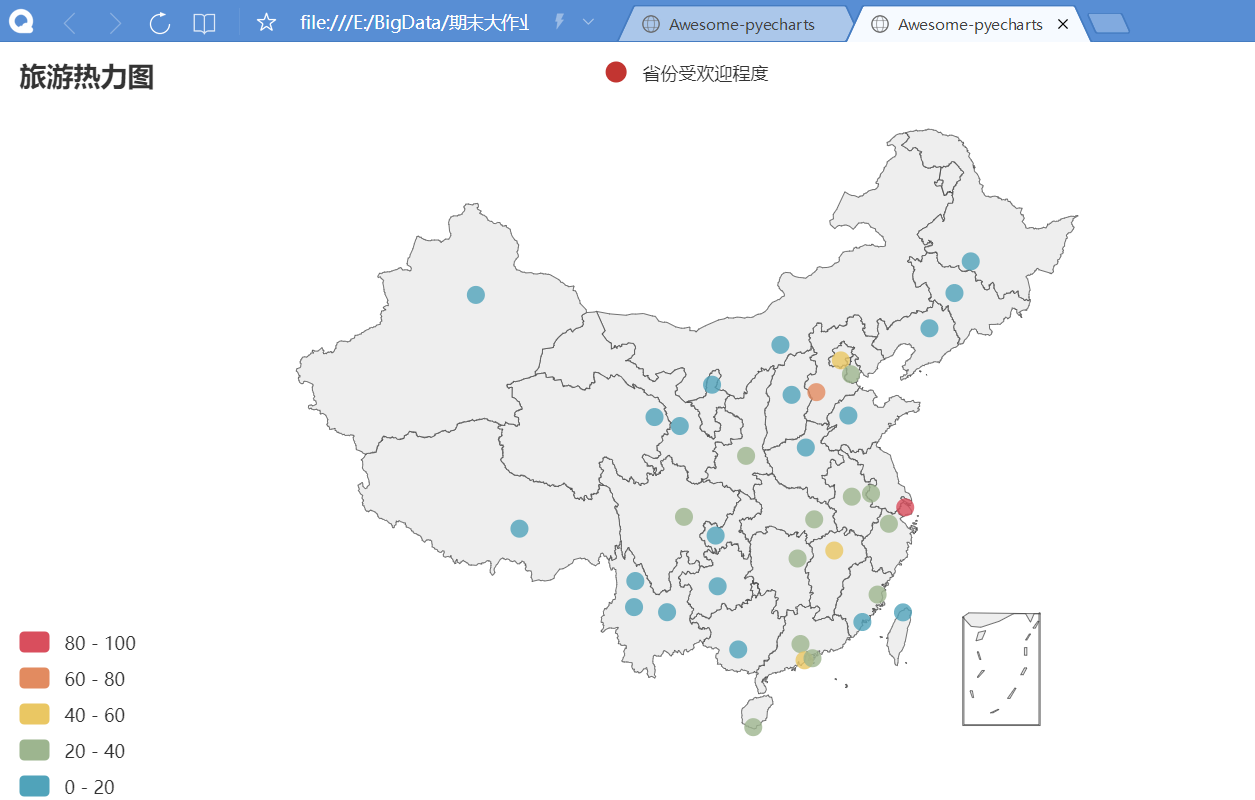
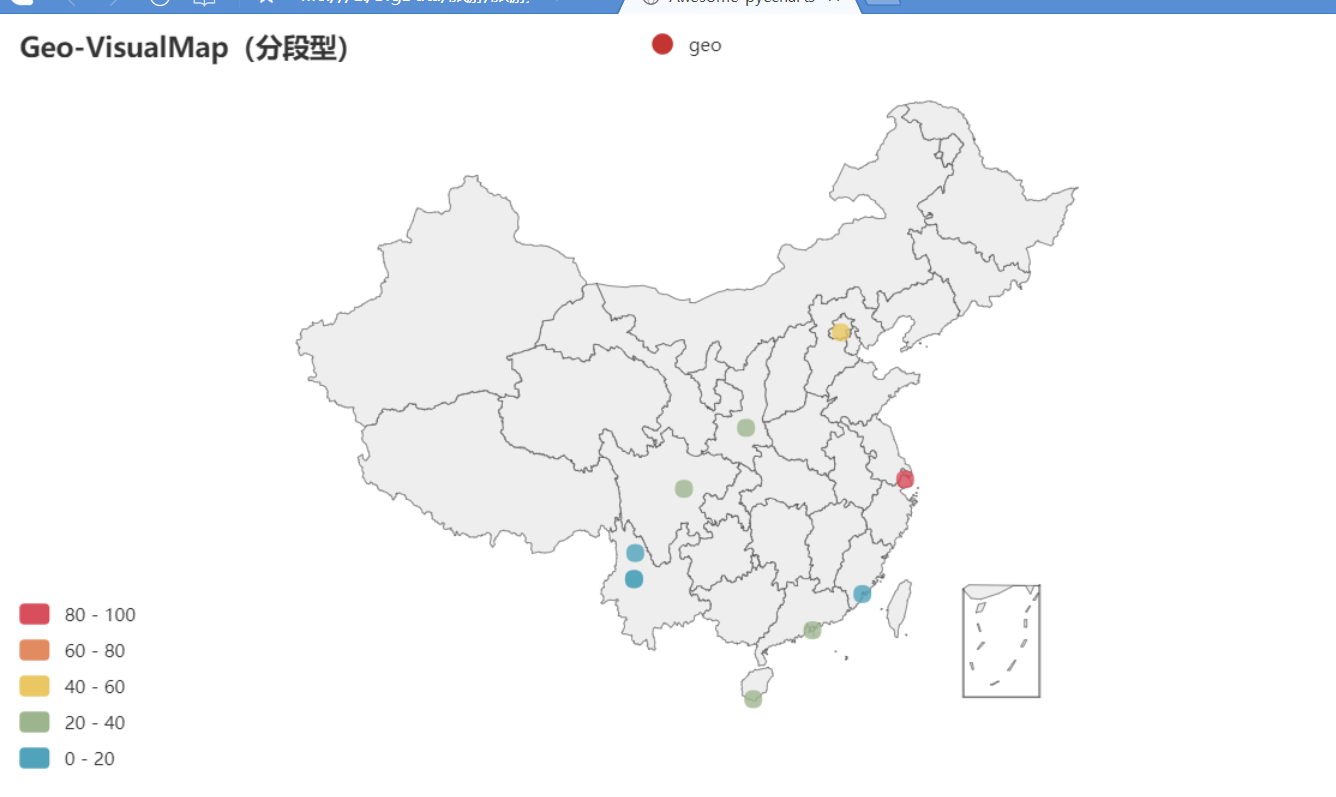
改进前 改进后

## 5.3 HeatMap与VisualMap

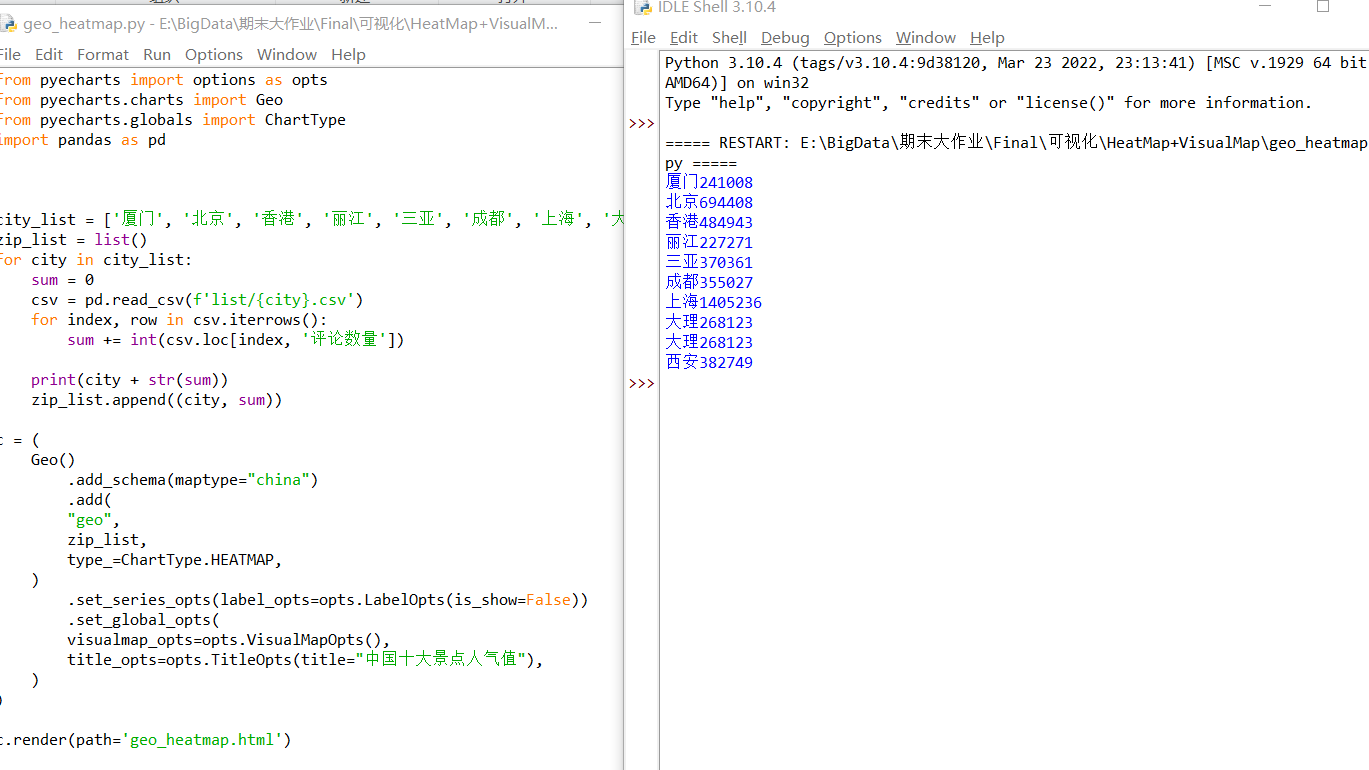
直接对于site进行统计，并生成：

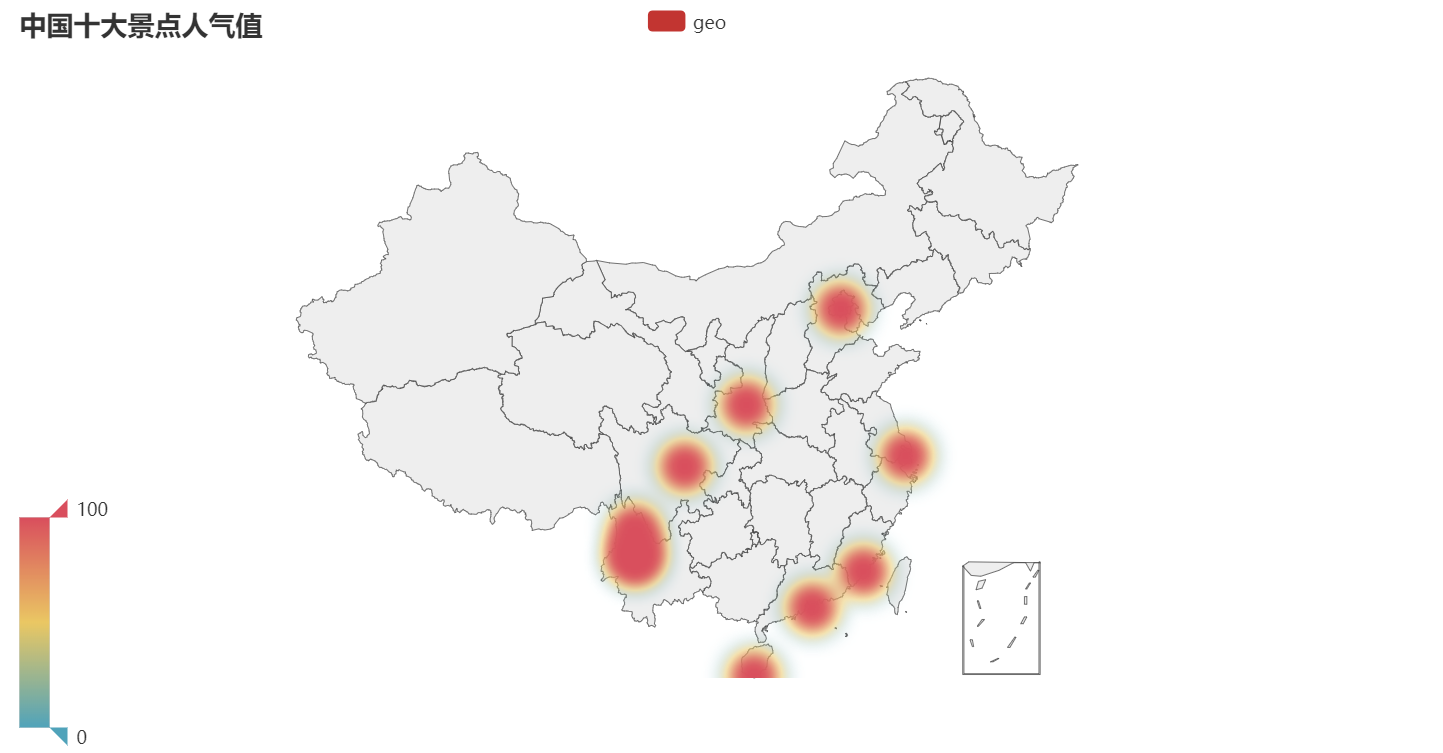


也正如在之前提到过[更新爬取数据集](#_3.3_更新爬取数据集)，这里爬取了更多城市的数据：



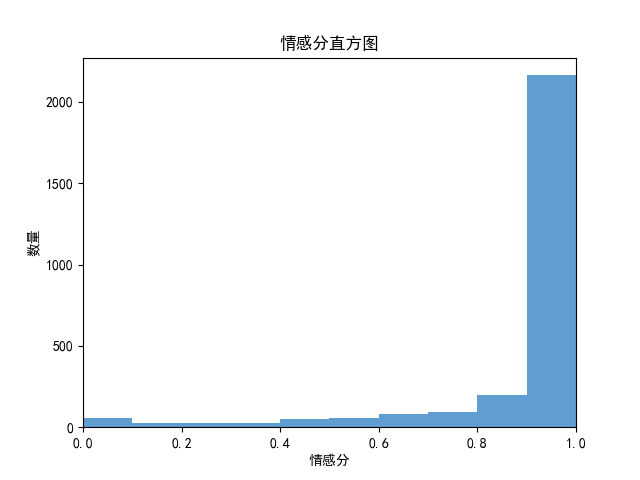
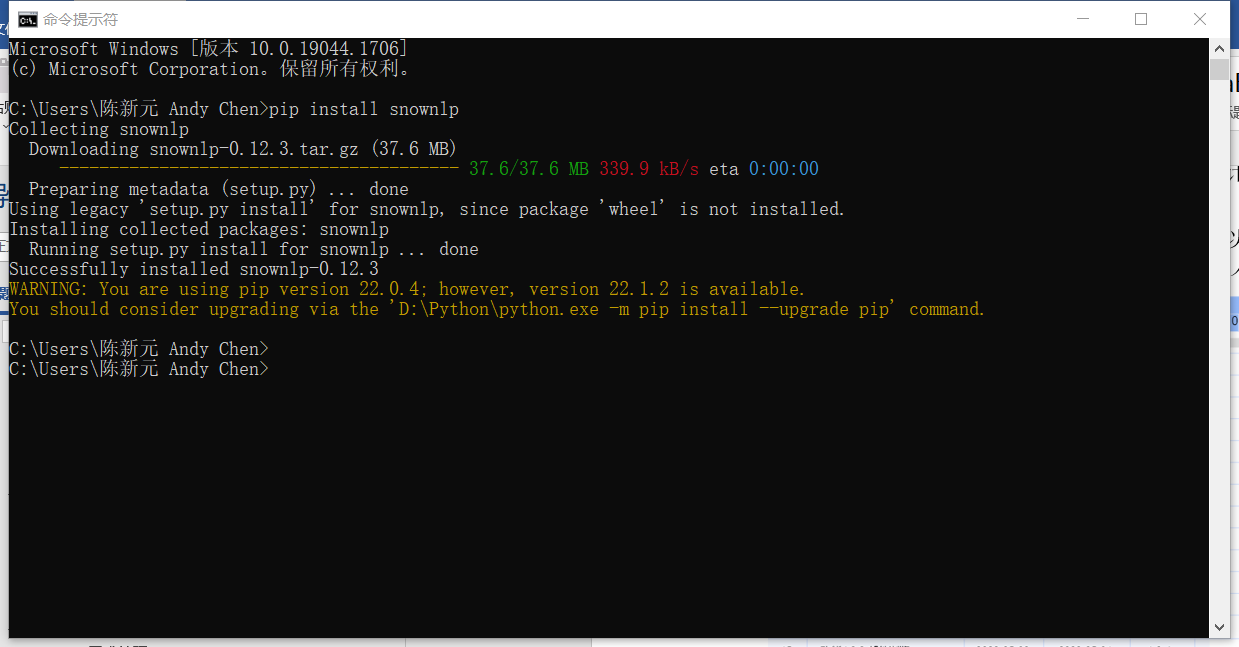
运行heatmap也是一样，只不过这里没有更新更多数据：





## 5.4 SnowNLP

本模块都是处理的Unicode编码，所以使用时需要decode成Unicode。因为本身就带词典库，所以可以很简易的操控而不需要训练数据：



# 六、总结

本次大数据的应用让我受益匪浅。

从一开始只是用python爬虫就要分析半天web结构，到之后直接用json快速定位；

从一开始parser一步一坑的解析，到之后直接beautifulsoup快速解析；

从不会使用shell命令操作Hadoop，到之后连各种命令闭眼都能熟练使用；

从WordCloud能够生成第一张测试图片，到之后能够调整各种参数使得图形呈现成更为美观、合适的词云；

从配置不出来环境，到之后能够熟练运用成熟的Docker直接下载；

从一开始连MapReduce都搞不懂为什么要分开，到之后可以熟练使用scala操控spark；

从不会建立搜索引擎，到可以写个脚本注入数据并且运行；

从不懂什么叫机器学习，到可以使用成熟的训练过的库进行词汇分析；

从不会使用Echarts，到可以熟练使用PyeCharts生成heatmap与VisualMap；

……

可以说通过这段时期的学习，让我更加坚信自己有着强大的自学能力，而且潘老师也对我莫大的指导，培养了我对于大数据分析的兴趣，对我之后的毕业设计，甚至研究生进修都有着非常大的引导作用。

# 参考文献：

1. 金赛.近10年国内旅游大数据文献综述[J].旅游纵览,2020(17):27-29.
2. 于超越.大数据分析在旅游领域中的应用探讨[J].江西电力职业技术学院学报,2020,33(06):166-168.
3. 李茜燕.大数据背景下旅游信息与区域旅游合作的耦合研究[J].情报科学,2016,34(04):129-132.
4. 高慧君,李君轶.基于微博大数据的游客情感与气候舒适度关系研究——以西安市国内游客为例[J].陕西师范大学学报(自然科学版),2017,45(01):110-117.DOI:10.15983/j.cnki.jsnu.2017.01.414.
5. 阮冬茹,平晓丽,赵刚,高凯.基于大数据的河北省文化旅游信息化现状分析[J].河北工业科技,2015,32(04):335-341.
6. 周效东,金梅.基于旅游大数据构建景区安全预警机制[J].四川旅游学院学报,2017(01):78-80.
7. 张诗雨,罗文佳,李化.基于大数据的旅游景区拥堵预警机制研究[J].信息技术与信息化,2017(12):161-162.
8. 黄先开,张丽峰,丁于思.百度指数与旅游景区游客量的关系及预测研究——以北京故宫为例[J].旅游学刊,2013,28(11):93-100.
9. 王军伟,才书训.决策支持系统在旅游景区优先开发中的应用[J].东北大学学报,2002(07):632-635.
10. 郑天翔,吴蓉,罗海媛.国内景区旅游流调控研究综述[J].地理与地理信息科学,2015,31(05):90-96.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **教师评分(满分100分)：**   |  |  |  |  | | --- | --- | --- | --- | | **项 目** | | **分 值** | **得分** | | 态度与整体表现 | | 10 |  | | 程序开发 | 数据量 | 10 |  | | 功能完整性 | 20 |  | | 技术难度 | 10 |  | | 代码质量 | 10 |  | | 创新能力 | 10 |  | | 报告质量 | 格式规范 | 10 |  | | 内容完整性 | 20 |  | | 总 计 | | |  | |