东 莞 理 工 学 院

本 科 毕 业 设 计

**（2023届）**

**题目：基于python的旅游数据分析平台设计与实现**

**学生姓名：麦志桓**

**学 号： 201941417128**

**院（系）： 计算机科学与技术学院**

**专业班级：数据科学与大数据2班**

**指导教师：李广明**

**起止时间：2023年3月—2023年4月**

**基于python的旅游数据分析平台设计与实现**

**摘 要：**在疫情防控期间，全国的旅游业几乎是在停摆的状态。随着如今我国疫情的政策放开，旅游业也逐步复苏，在全国各地的旅游优惠政策支持下，掀起了一波旅游热潮。基于目前的环境下，本人也因此设计一个旅游数据分析平台，把我国的旅游景点信息通过各种图表分析的展现出来，可以更直观的查看出景点的热度等各种信息，可以更好的供人选择旅游城市与景点。经过本人的深思熟虑，决定以python为主体语言构造一个数据分析平台，运用Scrapy、scrapy-redis进行数据抓取，crontab进行定时爬取，Gerapy进行爬虫的部署，mysql数据库进行数据存储，redis数据库进行数据缓存，Pandas和Numpy进行数据处理，echarts进行数据可视化展示，flex布局进行数据大屏,Flask框架进行前后端交互。

**关键词：** 数据挖掘；网络爬虫；数据分析；数据可视化

**Design and implementation of tourism data analysis platform based on python**

**Abstract:** During the epidemic prevention and control period, the country's tourism industry almost shut down. With the release of the epidemic policy, the tourism industry has gradually recovered. With the support of the tourism preferential policies all over the country, a wave of tourism boom has been set off. Based on the current environment, I also design a tourism data analysis platform, the information of our tourist attractions through a variety of chart analysis show, can be more intuitive view of the heat of scenic spots and other information, can be better for people to choose tourist cities and attractions. After careful consideration, I have decided to construct a data analysis platform using python as the main language, scrapy and Scrapy-Redis are used for data scraping, crontab is used for timing crawling, Gerapy is used for crawler deployment, and mysql database is used for data storage. redis database is used for data caching, Pandas and Numpy are used for data processing, echarts is used for data visualization, flex layout is used for data screen, and Flask framework is used for front-end and back-end interaction.

**Keywords:** Data mining; Web crawler; Data analysis; Data visualization

**目 录**

[第1章 绪论 6](#_Toc75360562)

[1.1 研究背景与意义 6](#_Toc1717097451)

[1.2 国内外发展现状 6](#_Toc1371610571)

[1.3 研究内容 7](#_Toc1569399899)

[1.4 论文结构 8](#_Toc1509950039)

[第2章 相关技术介绍 10](#_Toc916048874)

[2.1 网络爬虫 10](#_Toc723159975)

[2.1.1网络爬虫基本概念 10](#_Toc1539741452)

[2.1.2爬虫分类 11](#_Toc1256637414)

[2.1.3常见反爬和应对方法 11](#_Toc1950832500)

[2.1.4爬虫框架 13](#_Toc2008988751)

[2.1 数据库 15](#_Toc188556276)

[2.2 数据处理 16](#_Toc1526951407)

[2.3 数据可视化 16](#_Toc1042715799)

[2.4 Flask框架 17](#_Toc1457874273)

[第3章 数据采集 18](#_Toc1851977688)

[3.1旅游景点采集 18](#_Toc561022598)

[3.2旅游景点评论采集 21](#_Toc1653594256)

[3.3城市交通数据采集 21](#_Toc1372784765)

[3.4 爬虫部署 21](#_Toc1976725634)

[第4章 数据库设计 21](#_Toc1255711548)

[4.1 mysql数据库的设计 21](#_Toc1422188167)

[4.2 redis数据库 24](#_Toc1223531659)

[第5章 平台设计 26](#_Toc1740672788)

[5.1 功能组成 26](#_Toc317824835)

[5.2 结构分析 26](#_Toc890171756)

[第6章 数据可视化 26](#_Toc1745618090)

[6.1数据大屏 26](#_Toc1829136963)

[6.2数据分析 26](#_Toc1076530336)

[第7 结论 26](#_Toc695631177)

[7.1论文总结 26](#_Toc572217571)

[7.2存在问题 26](#_Toc828944531)

[7.3未来展望 26](#_Toc1344314428)

# 第1章 绪论

## 研究背景与意义

随着旅游业的发展和智能化、信息化的进步，数据分析已经成为了旅游业发展的重要手段。旅游景点数据分析可以帮助旅游企业了解游客的需求和兴趣，优化旅游产品，提高服务质量，提升企业的竞争力。此外，旅游景点数据分析还可以为政府部门提供决策依据，制定合理的旅游政策，推动旅游业的可持续发展计划。

精神文明日益发展的今天, 出行旅游成为人们的主要休闲方式和社会经济活动, 旅游业不仅能推动多个产业的发展, 调节产业结构, 同时也满足了人们日益增长的文化需要。其越来越成为国民经济的重要支撑。相关数据显示, 疫情放开后的旅游人数出现急剧增长的趋势。而与之相对应的是海量的无序化旅游数据日益激增。针对有效旅游数据进行准确分析、 合理预测有助于制定与旅游发展相关的规划, 同时能有效地帮助后续在旅游设施分配和资源调度等方面上制定明智的决策, 发挥旅游信息价值的最大化。由此可知, 旅游经济的发展与对数据的合理分析是密不可分的。

旅游景点数据分析的研究意义在于帮助我们了解旅游产业的市场情况、旅游者的旅游偏好、旅游资源的分布情况等，为旅游企业和政府决策提供科学的依据，更好的、更方便的选择旅游城市和旅游景点。通过数据分析可以优化旅游路线、提高旅游体验、促进旅游业的发展，同时也有助于保护旅游资源、提升旅游景点的品质和服务水平。

## 国内外发展现状

由于2020年爆发了全球性的新型冠状病毒疫情，为了防止疫情的传播，政府也实施了一系列的措施来应对，减少人员外出，旅游景点关闭等，国内外旅游人数急剧下降，旅游业的发展进入了停摆状态。根据中国文旅部数据统计，2019-2020年我国国内游客数据从60.06亿下降到28.79亿。此外，随着国外疫情的发展，出入境旅游市场受到较大冲击。2020年以来，全球和中国旅游市场面临着较为严峻的挑战。直至2022年末，随着疫情的好转，我国以及全国疫情政策放开，旅游行业复苏，迎来了全面的发展。旅游市场正在迅速回暖，各地纷纷出台优惠措施大力支持旅游市场的发展，在优惠政策的加持下，各地的人们纷纷响应，加入到旅游的热潮行动中。

在这种旅游热潮的环境下，选择一个好的旅游城市和旅游景点就形成为一个难点。此时，就需要通过大数据分析结合图表的形势供人们更好的选择，与文字相比，图表更有表现力，所要呈现的内容更直观。旅游景点数据分析在国内外都有相应的发展，随着旅游行业的不断发展和数据技术的不断进步，更多的企业和机构开始利用数据分析来提高旅游景区的管理和推广效果。目前国内优秀的旅游景点数据分析系统有携程的"猫眼大数据"以及途牛的"大数据实验室"等。在国外，像美国的"TripAdvisor"以及"Expedia"等都是比较著名的旅游景点分析平台。

## 研究内容

本文研究方向是设计一个旅游数据分析的平台。主要分为数据采集，数据清洗，数据储存，数据分析，数据可视化这几部分

数据采集：

1. 爬取全国各地的旅游景点，目标网站是同城旅行网站，技术采用的是Scrapy框架，Scrapy是一个基于Twisted的异步处理框架，可以更方便、更快的采集完数据。其中该网站业涉及到数据反爬。请求参数加密，cookie反爬等，通过js逆向解决。
2. 爬取全国各地旅景点的评论，由于目标网站的评论数目很庞大，达到200w的数据，采用Scarpy-redis来进行分布式爬取。分布式爬虫用一个共同的爬虫程序，同时部署到多台电脑上运行，这样可以提高爬虫速度，实现分布式爬虫。为了保证代码的可执行性和稳定性，构造了一个ip代理池，随机headers池。每请求一次切换不同的ip和headers。
3. 爬取高德地图的交通数据，查看该旅游城市的出行情况。由于城市交通是在实时变化的，采用了crontab进行定时爬取，每小时爬取一次数据。
4. 爬虫阿里云地图数据。本平台设计的数据大屏模块需要地图数据的支持。

找到阿里云地图的接口数据，根据我国各个城市的ID进行采集每一个城市的经纬度信息。

数据清洗：

主要运用pandas库进行处理，对数据进行检查清洗。包括数据缺失、数据填充、重复数据去重、无效数据删除、异常值删除

数据储存：

根据爬虫部分的需求把数据存储的方式主要是mysql数据库。数据量大的评论数据切换mysql的引擎，由InnoDB切换为MyISAM引擎。

数据分析：

主要通过echarts进行数据分析。平台中的运用的echarts图表有:折线图、柱状图、面积图、饼图、玫瑰图、漏斗图、雷达图、热力图、词云图等图表。

数据可视化：

通过flask轻度框架，首先将原本的地址，重列想到我们所创建的网页，前端通过点击按钮，往后端发送请求，发送请求向前端返回数据，前端将返回的数据传递给js，最后呈现在网页上。

## 论文结构

本文共分为七章，其组织结构如下（图1.1所示）：

第一章 绪论主要介绍该课题的研究背景与意义、分析国内外发展现状、叙述该论文研究的内容，为论文提供内容并指明方向。

第二章 相关技术的介绍。主要介绍论文使用到的四种技术：数据采集、redis与mysql数据库、数据处理、数据分析、flask框架。并逐一介绍其优点以及在论文中发挥的作用。整片论文是以这几种技术为支撑完成系统搭建过程。

第三章 数据采集过程。对平台的设计提供数据的支持。完成数据的爬取与预处理。

第四章 数据库的设计。根据爬取的数据量大小以及需求构造一种合适数据库的模型，涉及表结构的创建、字段的设计、索引、以及分表等。

第五章 平台的设计，flask框架进行前后端交互，以及平台一些注册、登录等操作功能的实现。

第六章 数据可视化展示。对可视化操作的描述并通过flask结合echarts对爬取并预处理过后的数据进行可视化。

第七章 总结与展望。对论文进行总结，并指出数据挖掘与可视化系统的不足不断更新完善，尽可能解决系统存在或出现的问题。

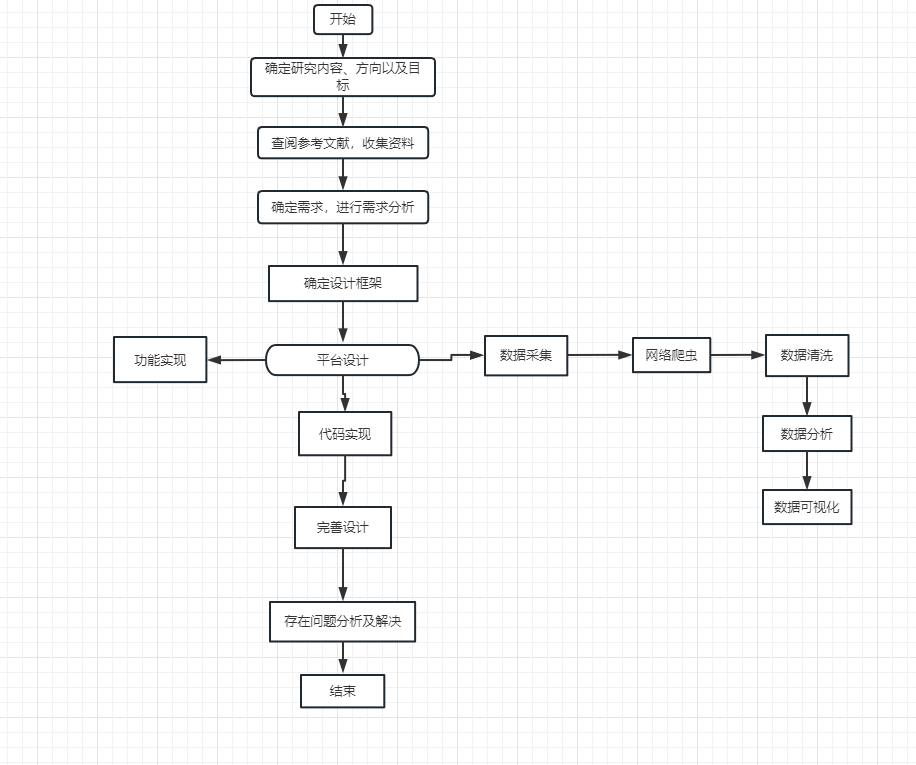


图 1.1论文组织结构

# 第2章 相关技术介绍

## 网络爬虫

### 2.1.1网络爬虫基本概念

网络爬虫的价值其实就是数据的价值，在互联网社会中，数据是无价之宝，一切皆为数据，谁拥有了大量有用的数据，谁就拥有了决策的主动权。

网络爬虫是能够自动抓取网页中各类数据的一段程序。网络爬虫通过网页的链接地址来查找网页内容，并直接返回给用户所需要的数据，不需要人工操纵浏览器获取。爬虫是搜索引擎中的重要组成部分，为搜索引擎抓取互联网中的数据。

设计一个爬虫需要解决以下几个问题：

1. 确定目标网站。获取数据所在的url链接或者api接口。
2. 获取网页。这里是获取网页的源代码。源代码里包含了网页的部分有用信息，所以只要把源代码获取下来，就可以从中提取想要的信息了。
3. 提取信息。获取网页源代码后，接下来就是分析网页源代码，从中提取我们想要的数据。首先，最通用的方法便是采用正则表达式提取，这是一个万能的方法，但是在构造正则表达式时比较复杂且容易出错。另外，由于网页的结构有一定的规则，所以还有一些根据网页节点属性、CSS 选择器或 XPath 来提取网页信息的库，如 Beautiful Soup、pyquery、lxml 等。使用这些库，我们可以高效快速地从中提取网页信息，如节点的属性、文本值等。提取信息是爬虫非常重要的部分，它可以使杂乱的数据变得条理清晰，以便我们后续处理和分析数据。
4. 保存数据。提取信息后，我们一般会将提取到的数据保存到某处以便后续使用。这里保存形式有多种多样，如可以简单保存为 TXT 文本或 JSON 文本，也可以保存到数据库，如 MySQL 和 MongoDB 等，也可保存至远程服务器，如借助 SFTP 进行操作等。

爬虫流程图如下(图2.1所示):

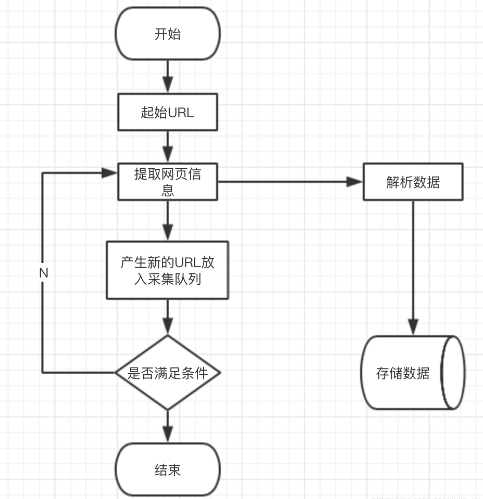


图 2.1论文组织结构

### 2.1.2爬虫分类

爬虫分为：通用爬虫和聚焦爬虫

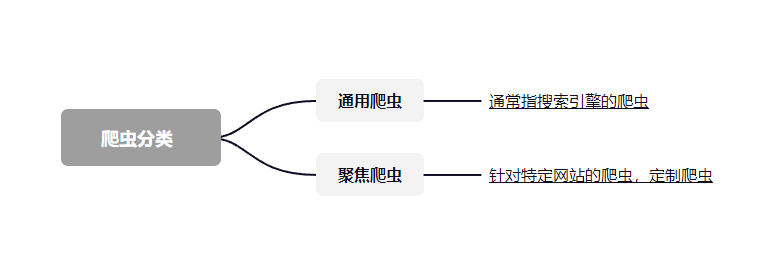


图 2.2爬虫分类

### 2.1.3常见反爬和应对方法

1. Headers反爬虫：

从用户请求的Headers反爬虫是目前最为常见的一种反爬虫手段。大多网站是对headers中的user-agent进行检测，还有是对referer进行检测（也就是我们所说的防盗链检测）。如果遇到这些类似的反爬，只需在发送请求的headers加上ua和referer。而有时也会遇到检测同一ua多次发送请求而导致反爬的，则需要动态切换ua（自己构造一个ua池）。

1. 基于用户行为的反爬虫：

主要是检测的是同一时间同一个IP多次访问同一页面，或者同一个账户短时间内进行多次相同操作。面对这种情况，大多是通过增加代理IP来解决，可以通过购买IP代理或者自己爬取网上开源的代理IP，构造自己的IP池的途径。

1. Cookie反爬：

遇到cookie反爬本提供两种解决方案。第一种，观察找到cookie的加密参数，在源代码中搜索关键字，找到生成cookie加密参数的函数扣出自己生成即可，而当搜索关键字的结果为空时，则需要通过hook脚本来debug到生成cookie的位置。第二种，制造刷cookie的脚本，构造cookie池，每一次请求都动态获取一个cookie。

1. 参数加密：

遇到参数加密，与cookie反爬的第一种方法雷同

1. 出现验证码

图形验证码：干扰、杂色不是特别多的图片可以使用开源库 Tesseract 进行识别，太过复杂的需要借助第三方打码平台。

点击和拖动滑块验证码：可以借助 selenium、无图形界面浏览器（chromedirver 或者 phantomjs） 和pillow 包来模拟人的点击和滑动操作， pillow 可以根据色差识别需要滑动的位置。

### 2.1.4爬虫框架

在本平台设计中涉及到两种爬虫框架：scrapy与scrapy-redis

1. Scrapy：scrapy 是一个为了爬取网站数据，提取结构性数据而编写的应用框架，我们只需要实现少量代码，就能够快速地抓取到数据内容。 Scrapy 使用了 Twisted 异步网络框架来处理网络通讯，可以加快我们的下载速度，不用自己去实现异步框架，并且包含了各种中间件接口，可以灵活地完成各种需求。 scrapy 框架的工作流程（如图下2.3）：
2. 首先 Spiders（爬虫） 将需要发送请求的 url(requests)经 ScrapyEngine（引擎） 交给 Scheduler（调度器）。
3. Scheduler（排序，入队） 处理后，经 ScrapyEngine， DownloaderMiddlewares(可选，主要有User\_Agent， Proxy 代理)交给 Downloader。
4. Downloader 向互联网发送请求，并接收下载响应（response）。将响应（response）经ScrapyEngine， SpiderMiddlewares(可选)交给 Spiders。
5. Spiders 处理 response，提取数据并将数据经 ScrapyEngine 交给 ItemPipeline 保存（可以是本地，可以是数据库）。提取 url 重新经 ScrapyEngine 交给 Scheduler 进行下一个循环。直到无 Url 请求程序停止结束。

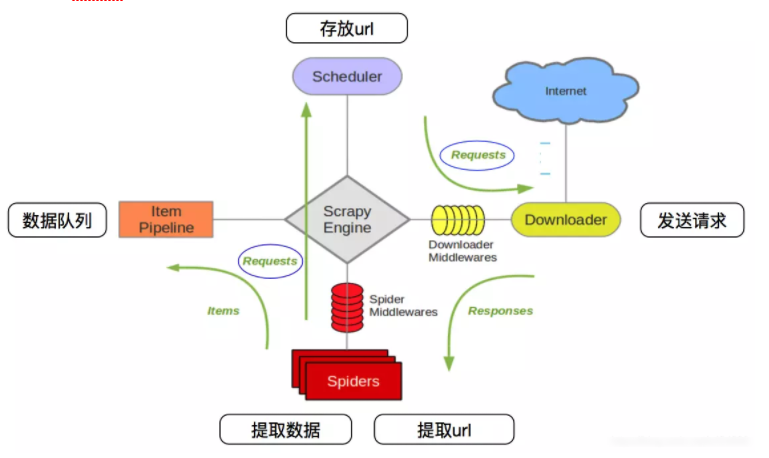


图 2.3 scrapy工作流程图

1. Scrapy-Redis: scrapy-redis是分布式爬虫框架。

分布式爬虫：用一个共同的爬虫程序，同时部署到多台电脑上运行，这样可以 提高爬虫速度，实现分布式爬虫。

Scrapy-redis是scrapy的一个组件，它使用了Redis数据库做为基础，目的为了更方便地让Scrapy实现分布式爬取Scrapy能做的事情很多，但是要做到大规模的分布式应用则捉襟见肘。有能人改变了Scrapy的队列调度，将起始的网址从start\_urls里分离出来，改为从redis读取，多个客户端可以同时读取同一个redis，从而实现了分布式的爬虫。

Scrapy-redis工作流程图（如图下2.4）：

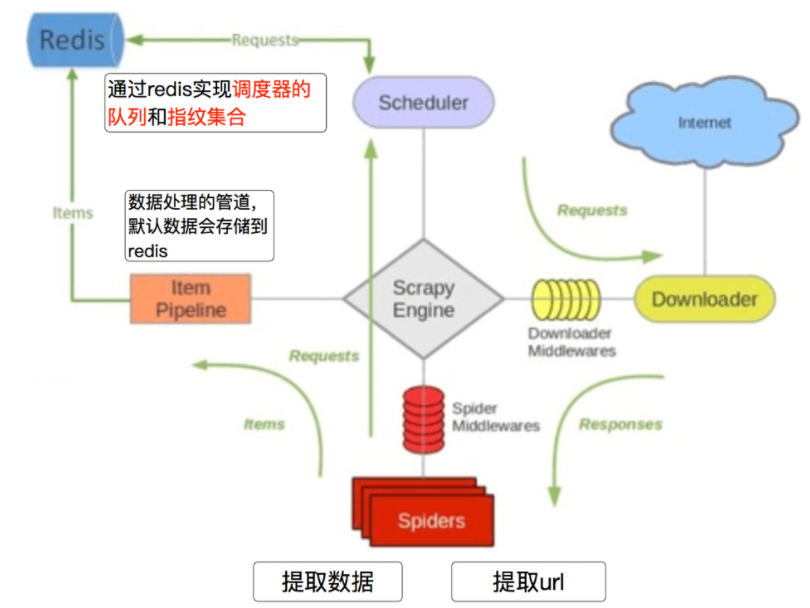


图 2.4 scrapy-redis工作流程图

## 数据库

数据库分为关系型数据库和非关系型数据库。

本平台设计中运用到的是关系型数据库mysql和非关系型数据库redis。

Mysql：

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB公司开发，后来被Sun公司收购，Sun公司后来又被Oracle公司收购，目前属于Oracle旗下产品。Mysql是开源免费的关系型数据库，是基于关系模型建立，用二维表进行数据存储的数据库，需要实现创建数据库、数据表和表的字段，表与表之间可以进行关联（一对多、多对多），是持久化存储，将数据存储在硬盘中，读取速度较慢。

Mysql具备以下特点：支持多线程，充分利用CPU资源；优化的SQL查询算法，有效地提高查询速度；提供多语言支持，常见的编码如GB2312、BIG5、UTF8；大型的数据库，可以处理拥有上千万条记录的大型数据库；支持事务特征，支持多种存储引擎；MySQL 软件采用了双授权政策，它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择MySQL作为网站数据库。

Redis：

非关系型数据库，使用前可以不用设置格式，以键值对的方式保存，文件格式相对自由，主要用与缓存数据库，也可以进行持久化存储。

redis存在5中数据结构：String（字符串），Hash（哈希），List（列表），Set（集合），zset（有序集合）。

redis 具备以下特点：读写性能优异，数据存在内存中；支持数据持久化，支持 AOF 和 RDB 两种持久化方式；支持主从复制，主机会自动将数据同步到从机，可以进行读写分离。Redis 较难支持在线扩容，在集群容量达到上限时在线扩容会变得很复杂。

## 数据处理

数据处理在数据分析中是至关重要的，做好数据清理的工作可以大大提高在数据分析中的效率。而在本平台设计中，主要采用python代码编写处理代码对数据进行处理步骤，其中使用到两个库：Numpy和Pandas

Numpy:是 Python 语言的一个扩展程序库，支持量大的纬度数组与矩阵运算，也提供了大量的数学函数库。是一个运行效率非常高的库,很适合做数据处理方面的工作。

Pandas:是python的一个扩展程序库，是一个基于Numpy创建的库，是专门为实现数据分析任务而创建的库。而pandas提供两种数据结构Series和DataFrame。在pandas也经常被使用与数据清洗起着重要的作用。数据集中存在数据缺失、数据格式错误、数据错误、重复数据、填充数据的情况都是用pandas中来处理的。

## 数据可视化

数据可视化：将数据转化为交互的图案或图像，把数据以图表的形式来展示。

在本平台设计中主要使用Ecahrts来实现数据可视化。Echarts是一个使用JavaScript实现的开源可视化库，可以在pc和移动设备上展示，目前兼容绝大部分浏览器（firefox，chrome，safari等），可随时随地任性展示。支持多种数据格式无需转换，丰富的可视化类型图表（折线图、柱状图、散点图、饼图、K线图、热力图、地图等）。

Echarts使用步骤：

步骤1：创建html页面，下载并引入echarts.js文件

步骤2：准备一个具备大小的DOM容器

步骤3：初始化echarts实例对象

步骤4：指定配置项和数据（option）

步骤5：将配置项设置给echarts的实例对象

## Flask框架

python中存在众多的web开发框架，但在近几年中较为流行是Flask和Django。而在本平台设计中用的是Flask框架。

Flask是一个基于python编写的轻量型框架，较其他同类型框架更为灵活、轻便且容易上手。Flask基于Werkzeug WSGI工具包和Jinja2模板引擎。WSGI是Web服务器和Web应用程序之间通用接口的规范。Werkzeug 是一个WSGI工具包，它实现了请求，响应对象和实用函数。jinja2是一个流行的模板引擎。使用jinja2提高了前端代码的复用率。这样可以提高开发效率和有利于后期开发与维护。且是基于结合[MVC模式](https://baike.baidu.com/item/MVC%E6%A8%A1%E5%BC%8F/713147?fromModule=lemma_inlink)进行开发，开发人员可以在短时间内就可以完成功能丰富的中小型网站或[Web服务](https://baike.baidu.com/item/Web%E6%9C%8D%E5%8A%A1/2837593?fromModule=lemma_inlink)的实现。

Flask作为一个web开发微框架，并不代表Flask 功能比较简陋、有所欠缺。

而是flask旨在保持核心简单而易于扩展，不会默认选择某种数据库和模板引擎

等对象，帮用户做过多的决策。且具备很强的定制性，可以根据自己的需求来添加相应的功能，开发出功能强大的网站。

Flask工作流程如下（图2.5）:

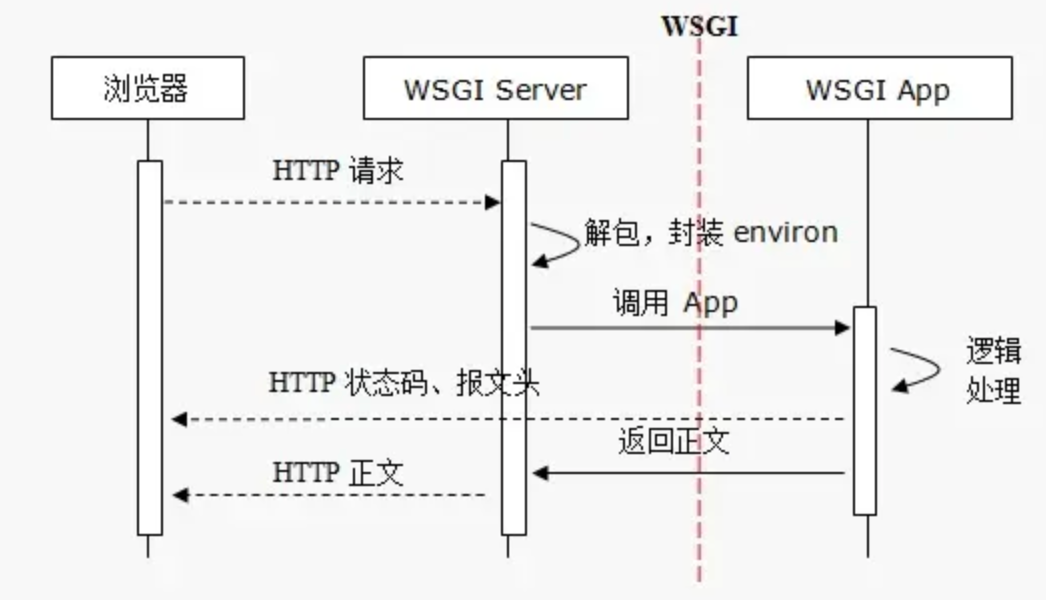


图 2.5 flask工作流程图

# 第3章 数据采集

## 3.1旅游景点采集

旅游景点数据采集目标网站：同程旅行

用scrapy进行多任务采集数据。分为两部分:menu列表页和detail详情页。

1. menu分析:

分析网页获取到网页数据接口:

<https://so.ly.com/scenery/newsearchlist_hot.aspx>

分析接口url，存在参数加密，page为传入的页数，iid是0-1的随机小数，dctrack则需要来逆向。（如图3.1）



图 3.1

分析dctrack参数：通过debuger定位到js中生成的位置，可知dctrack是收集用户的行为轨迹（如图3.2）。



图 3.2

跟栈进去分析可知，dctrack是由a.ct生成（图3.3）。

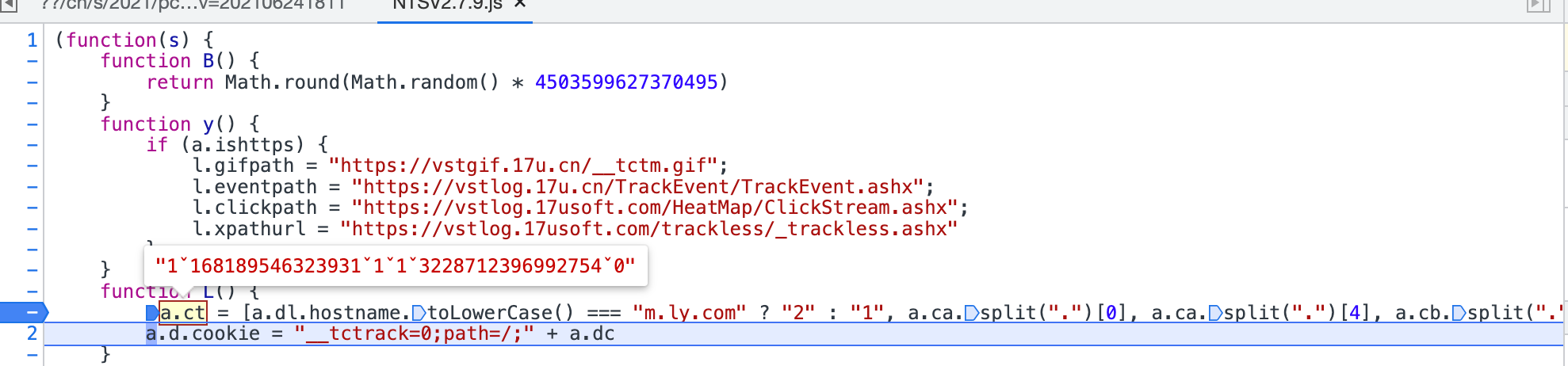


图 3.3

继续跟栈分析可以知道，a.ct是由a.ca与a.cb生成。ca与cb是由t函数生成，而t函数生成也需要传入cookie的值（图3.4）：

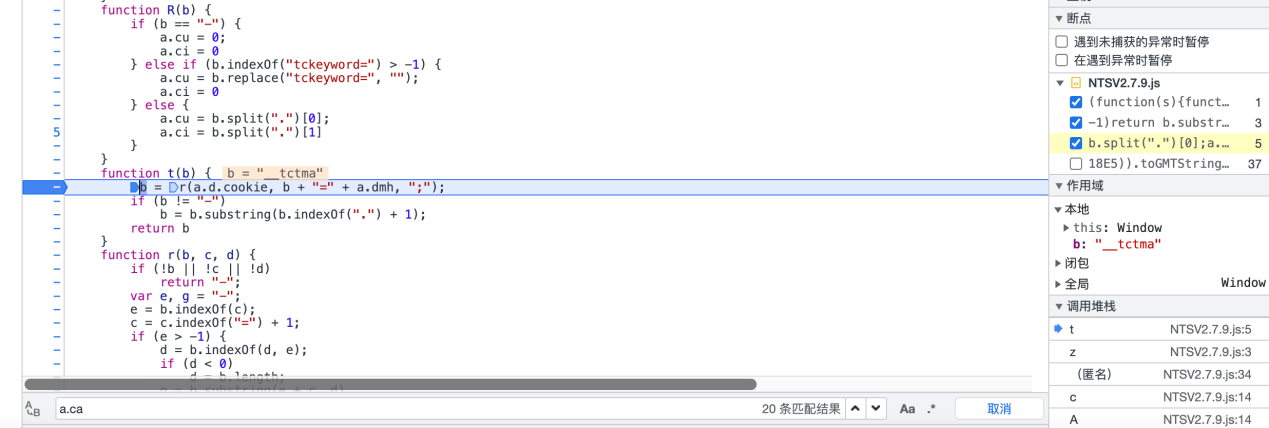


图 3.4

通过分析扣出对应的js函数，可知生成dctrack需要传入cookie（图3.5），而cookie不是固定不变的，就需要编写一个刷cookie的脚本。

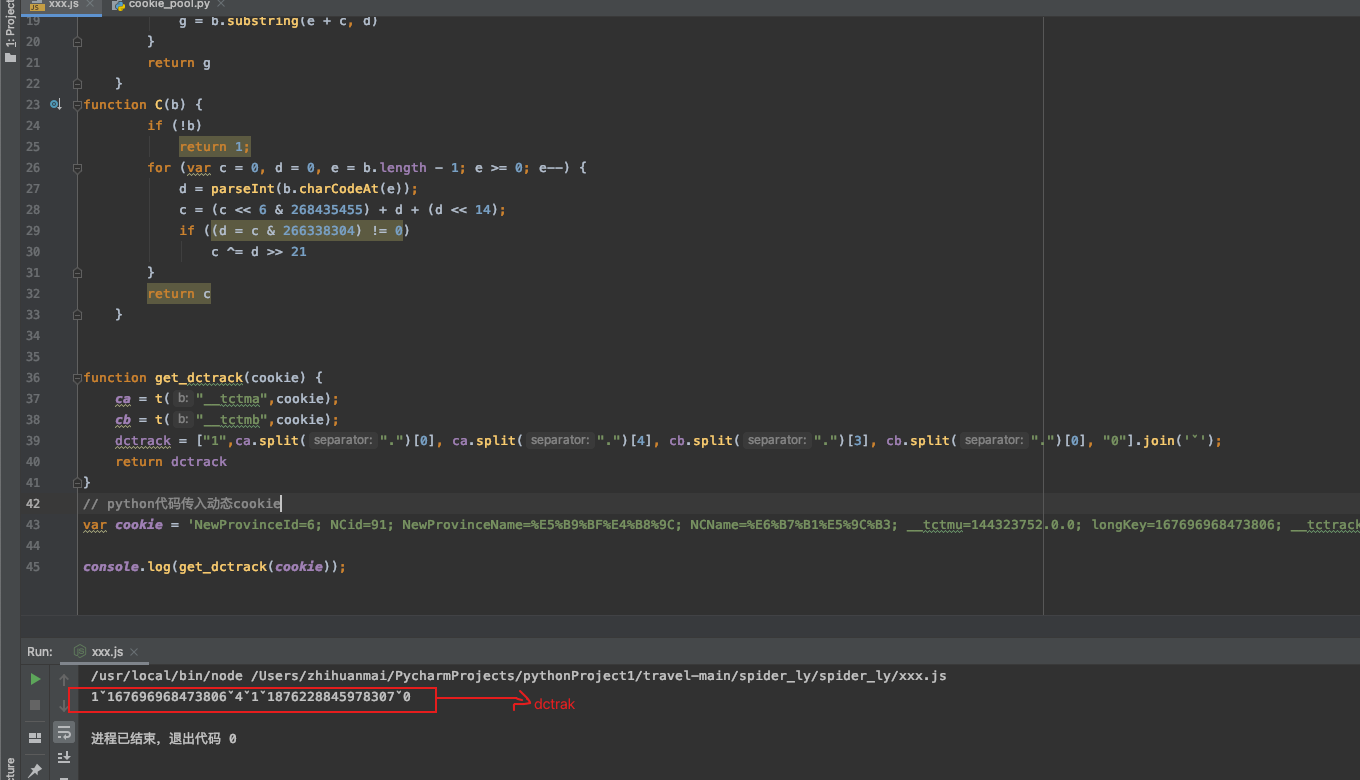


图 3.5

刷cookie脚本：使用PhantomJS和多任务进行自动化获取cookie，获取获的cookie存放到redis中，设置的容量为100个，当数量达到100个时，就等待消耗。数据小于100时，继续执行脚本保证cookie池始终保持有100个cookie（图3.6）。

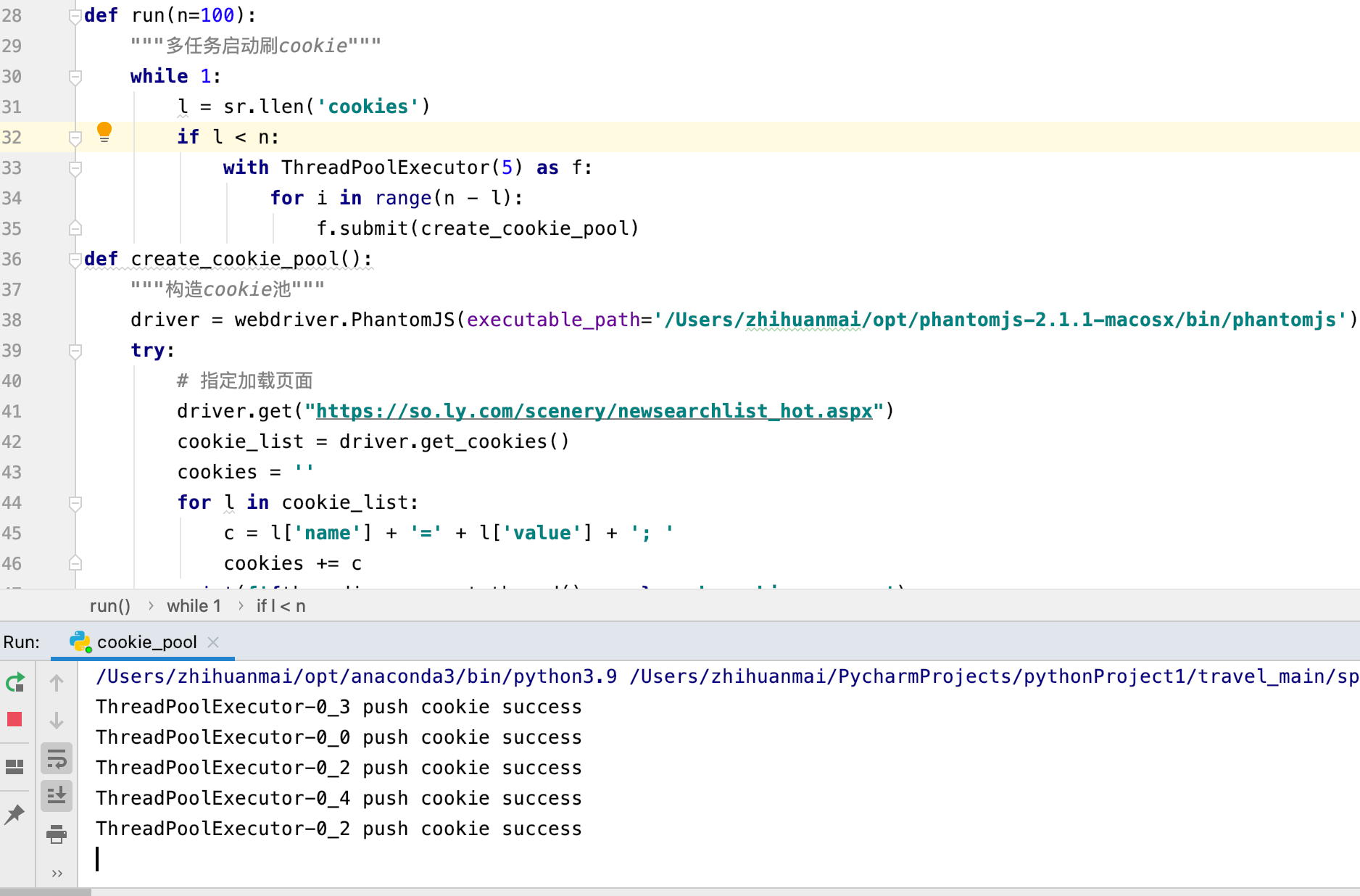


图 3.6

1. detail分析：

F12查看网络和源代码会进入的无限debug状态（图3.7）。此时，可以通过抓包工具charles进行抓包。

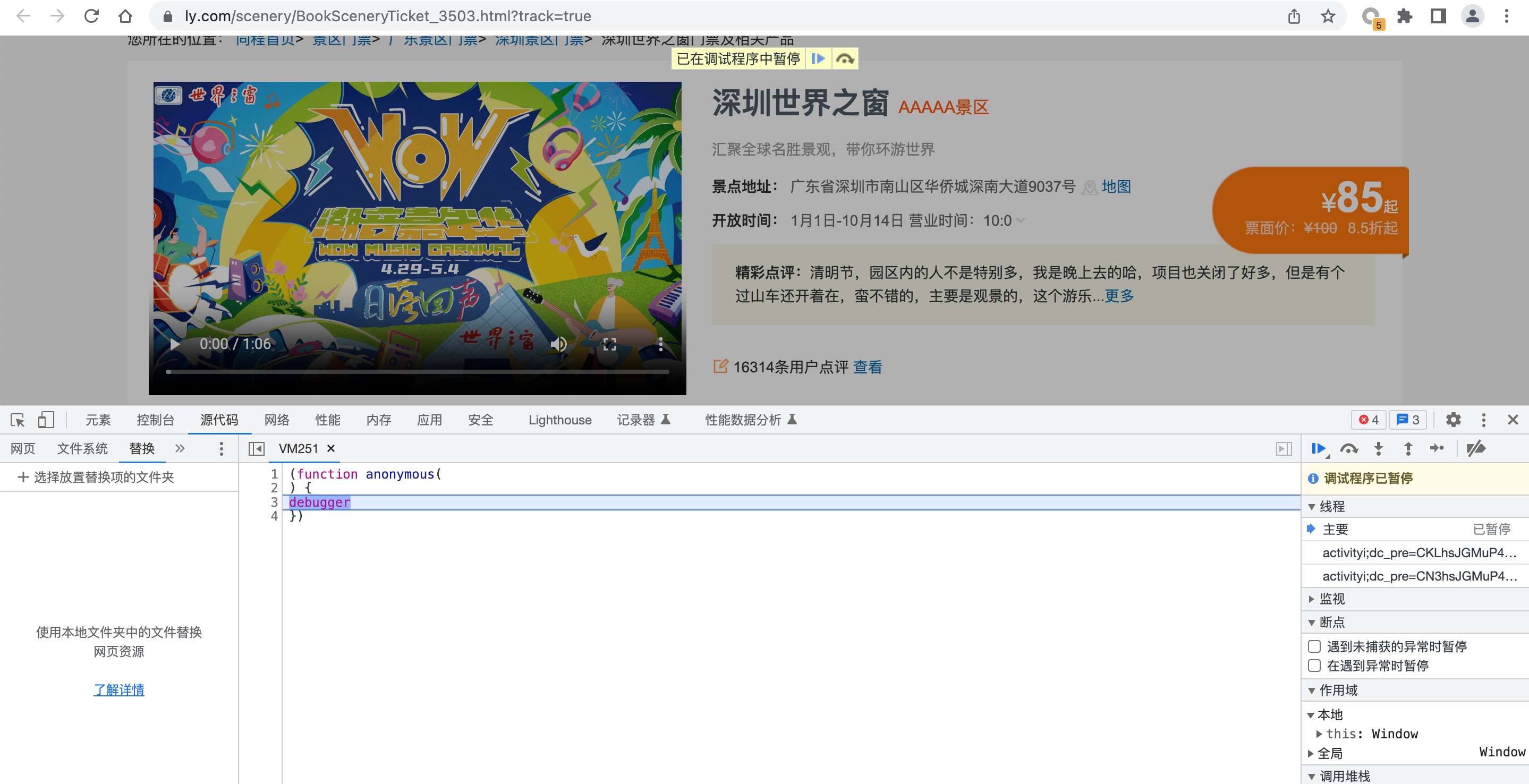


图 3.7

charles抓包（图3.8），根据抓包结果分析，可以直接获取结果。

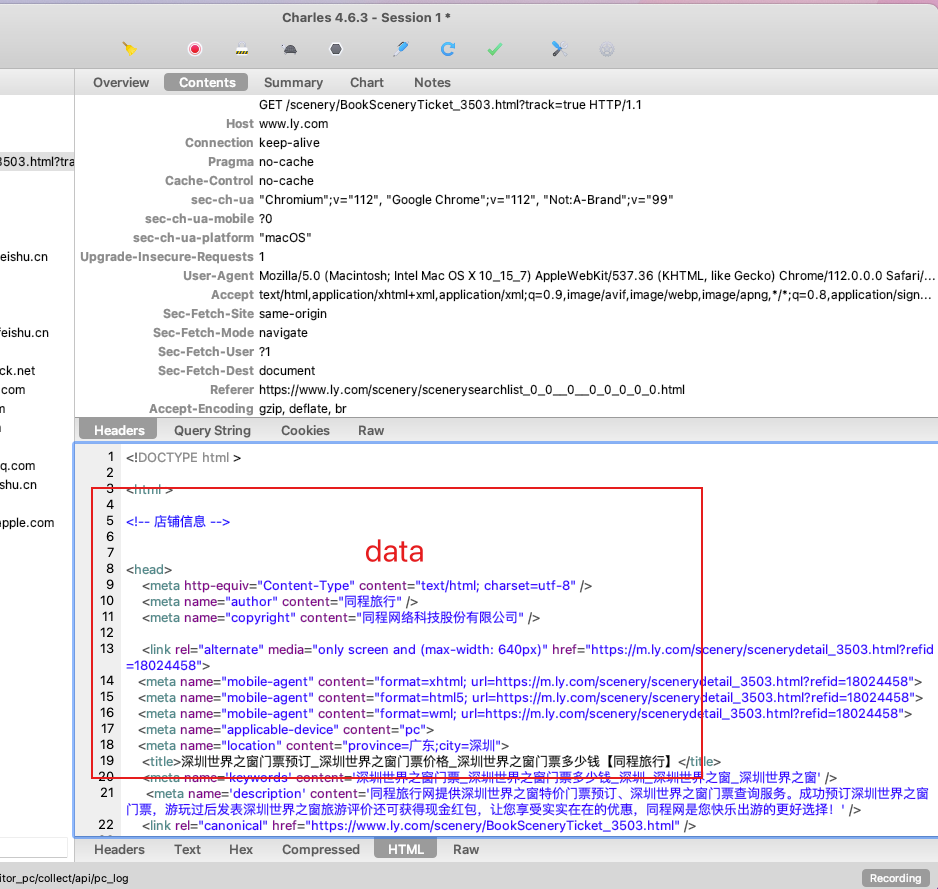


图 3.8

1. scrapy抓取景点数据。在启动crawl\_scenic抓取前需要启动刷cookie和刷ip的脚本。抓取的数据保存在csv文件和mysql数据库。控制台进入当前目录，输入 scrapy crawl crawl\_scenic 即可启动scrapy抓取景点数据（图3.9）。

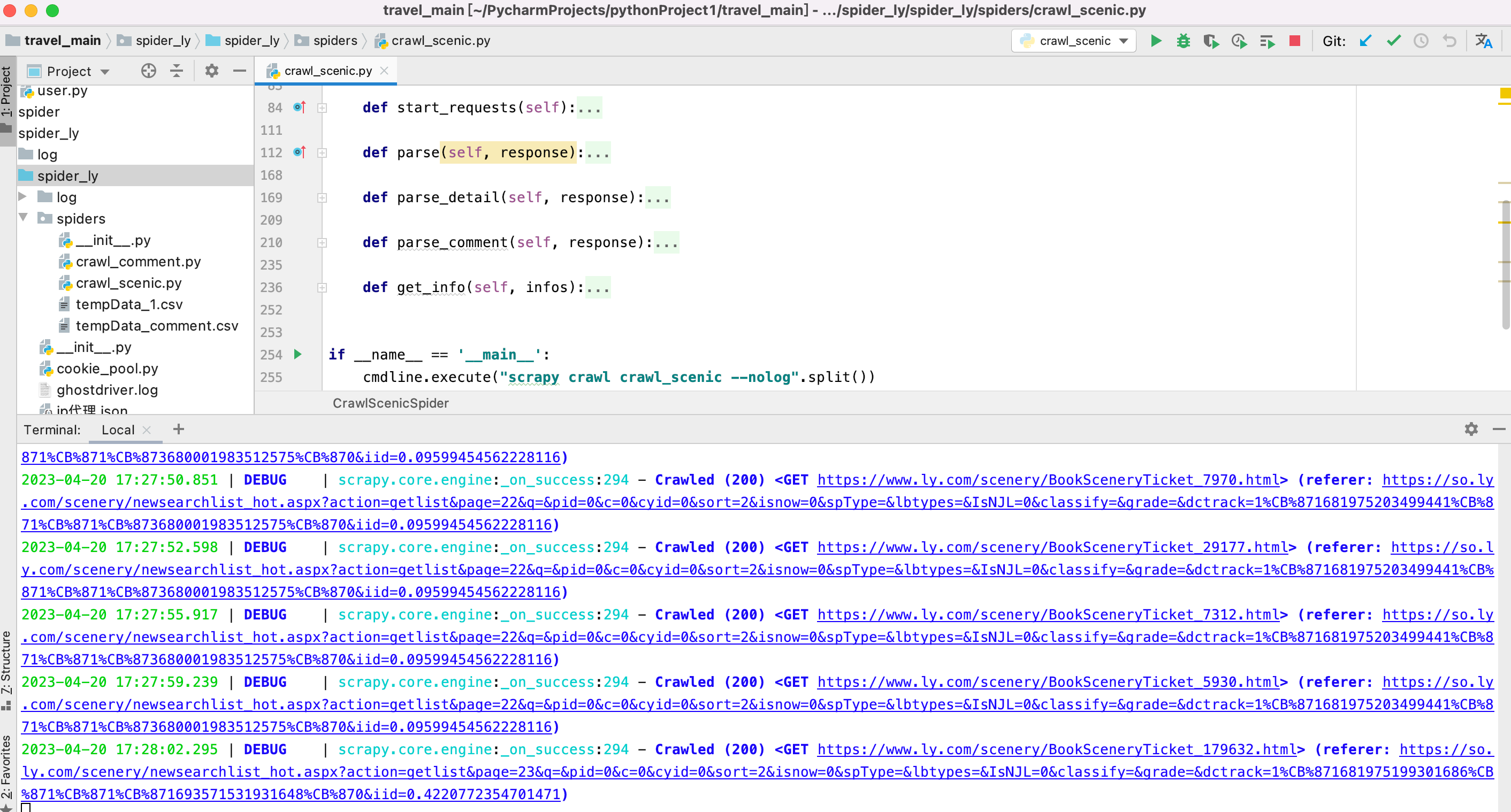


图 3.9

## 3.2旅游景点评论采集

网页存在无限debuger模型不好抓到评论接口，通过charles抓包得出评论的数据接口为：<https://www.ly.com/scenery/AjaxHelper/DianPingAjax.aspx> get请求，params参数(如图3.10)。经过调试得出，sid是景区id，page为当前的页数，id是0-1的随机小数，其他都是固定值。



图 3.10

评论数据量大，选择分布式爬取（scrapy-redis）。crawl\_comment.py脚本进行分布式的爬取，启动后，会监听redis中的comment\_url队列中的url，当有url时就会自动爬取，直到队列为空时，又会进入等待状态。也可以在其他机器中部署（同一份代码，不同的机器），redis服务器是同一个（如图3.11）。

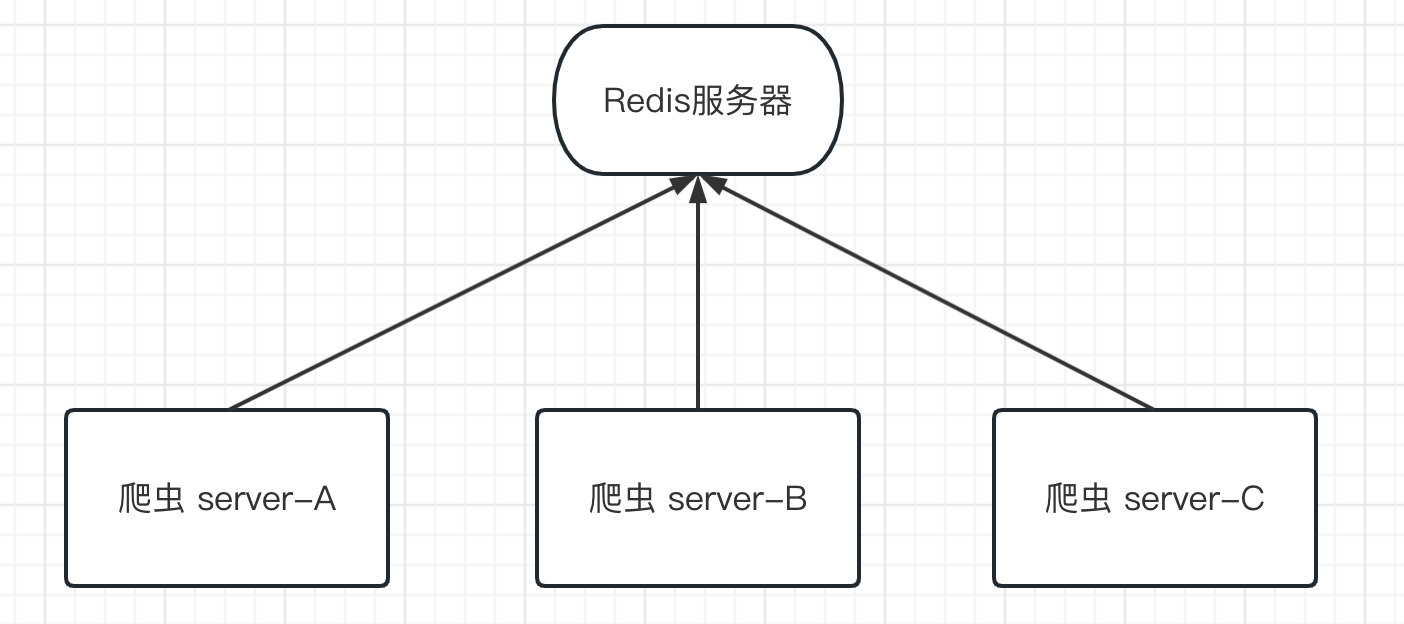


图 3.11

发布任务的脚本（add\_comment\_task\_url.py），运行此脚本会往redis中添加rpush 评论的url（如图3.12）。当爬虫机器监听到redis服务器有任务来了，会自动获取url，且不同的机器不会重复执行同一任务的url。



图 3.12

## 3.3城市交通数据采集

目标网站：<https://report.amap.com/detail.do?city=110000>

对目标网站采集三种类型数据,分别为交通健康情况、区域拥堵排名、道路拥堵情况

数据来源都是通过ajax接口返回的json格式。api接口链接不一致，参数需要根据城市的code和所要抓取的数据类型dataType决定。采用多进程的方式抓取数据（如图3.13）。因交通情况是实时更新的，采用crontab部署为定时执行爬虫脚本，实时抓取更新数据。基于学习用途设置间隔时间为12h。



图 3.13

## 3.4 爬虫部署

# 数据库设计

## 4.1 mysql数据库的设计

在数据储存方面选择的是mysql数据库。数据库设计在数据分析中起着重要作用，一个好的数据库设计可以加快数据分析的效率，可以较少在数据清洗和数据分析过程中冗余操作。其数据表设计的好坏直接影响数据库访问的快慢，影响系统的性能。在本数据分析平台中包含着

用户表、中国旅游数据表、国内景点表、评论表、交通数据方面的三张表。其中景点表常用于查询，评论表数据庞大（200w+）,因此做了以下的数据库方面的优化设计：sid字段设置为索引，且索引方法改为btree；表引擎改为MyISAM。

设计表的结构如下：

1. 用户表：用户登录、注册等功能

表4-1 user表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 小数位数 | 是否是主键 | 是否为空 |
| email | varchar(255) | 255 |  |  | YES |
| id | int |  | 0 | PRI | NO |
| name | varchar(255) | 255 |  |  | YES |
| password | varchar(255) | 255 |  |  | YES |

1. 中国旅游数据表：记录着我国从1994年-2022年的旅游数据的情况

表4-2 china\_tourism\_data表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 小数位数 | 是否是主键 | 是否为空 |
| id | int |  | 0 | PRI | NO |
| title | varchar(255) | 255 |  |  | YES |
| 1994 | float |  |  |  | YES |
| 1995 | float |  |  |  | YES |
| … | … |  |  |  | … |
| … | … |  |  |  | … |
| 2021 | float |  |  |  | YES |
| 2022 | float |  |  |  | YES |

1. 国内景点表:保存国内景点数据包括sid，价格，名字等数据

其中sid字段经常用于查询，设置sid为索引，索引方法为btree类型，并设计表的引擎为MyISAM。可以提高查询的速率。

表4-3 travel表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 小数位数 | 是否是主键 | 是否为空 |
| address | varchar(255) | 255 |  |  | YES |
| badNum | varchar(255) | 255 |  |  | YES |
| contents | text | 65535 |  |  | YES |
| degreeLevel | varchar(255) | 255 |  |  | YES |
| detail\_link | varchar(255) | 255 |  |  | YES |
| feature | varchar(1000) | 1000 |  |  | YES |
| full\_address | varchar(255) | 255 |  |  | YES |
| goodNum | varchar(255) | 255 |  |  | YES |
| hasImgNum | varchar(255) | 255 |  |  | YES |
| id | int |  | 0 | PRI | NO |
| images | text | 65535 |  |  | YES |
| level | varchar(255) | 255 |  |  | YES |
| main\_img | varchar(1000) | 1000 |  |  | YES |
| midNum | varchar(255) | 255 |  |  | YES |
| open\_time | varchar(255) | 255 |  |  | YES |
| price | varchar(255) | 255 |  |  | YES |
| serviceScoreAvgList | text | 65535 |  |  | YES |
| sid | int |  | 0 | MUL | YES |
| starNum | varchar(255) | 255 |  |  | YES |
| title | varchar(255) | 255 |  |  | YES |
| totalNum | varchar(255) | 255 |  |  | YES |
| video | varchar(255) | 255 |  |  | YES |

1. 评论表，储存景点对应的评论。因存在200w+评论，所有把sid设置为索引，索引方法为btree，表引擎改为MyISAM。

表4-4 travel表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 小数位数 | 是否是主键 | 是否为空 |
| comment\_cotent | text | 65535 |  |  | YES |
| comment\_stat | text | 65535 |  |  | YES |
| comment\_time | varchar(255) | 255 |  |  | YES |
| comment\_user | varchar(255) | 255 |  |  | YES |
| dpId | varchar(100) | 100 |  |  | YES |
| id | int unsigned |  | 0 | PRI | NO |
| page | int |  | 0 |  | YES |
| reply\_content | text | 65535 |  |  | YES |
| sid | int |  | 0 | MUL | YES |

1. 交通数据表。分为traffic\_district\_rank、traffic\_roadrank\_rank、traffic\_congestion\_index三张表，表的数据结构一致，但储存着不同的数据。

表4-4 traffic\_district\_rank表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 小数位数 | 是否是主键 | 是否为空 |
| code | int |  | 0 | MUL | YES |
| data | text | 65535 |  |  | YES |
| id | varchar(50) | 50 |  | PRI | NO |
| name | varchar(30) | 30 |  |  | YES |
| time | datetime |  |  |  | YES |
| typyid | Int |  | 0 |  | YES |

## 4.2 redis数据库

redis数据库是基于内存的存储，读取效率快，通常用于做缓存。在本平台设计中redis的作用是：数据采集时需要构造的cookie池和ip池、数据可视化中做缓存。

1. 构造cookie池：

根据数据采集章节中的分析结果是需要在采集景点信息中，构造一个cookie池用来获取动态ip。

选择redis中的list的来储存cookie，每新来一个cookie就从rpush进list，获取cookie使用lpop方法，保证每次获取的cookie都是最新的。也保证cookie池始终保持100个cookie(流程图4.1如下)。



图4.1 cookie池流程图

1. 构造ip池：

在评论采集过程中，数据量太大，需要切换ip，避免出现封ip的情况。因此需要构造一个动态的ip池。

选择的是zset来存在ip。

1. 缓存：

在数据可视化中，当遇到数据量大的时候，需要用redis做一个缓存。第一次访问时，把数据分析的结果存入到redis中，当再次访问时，就从redis中获取数据给到前端，就不需要再进行一次数据分析的流程，大大缩短了访问的时间(流程图4.3如下)。

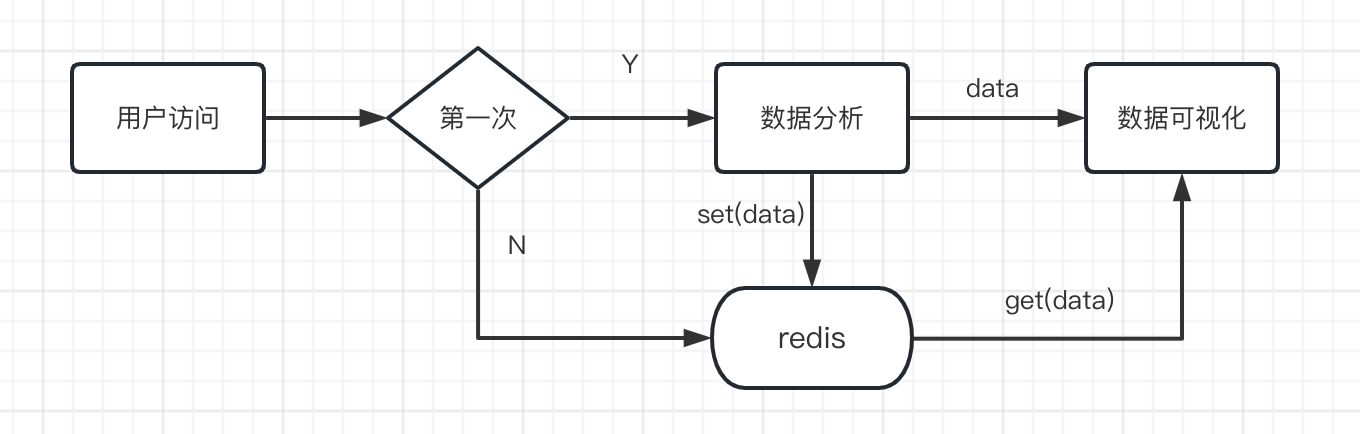


图4.3 redis缓存流程图

# 第5章 平台设计

## 5.1 功能组成

## 5.2 结构分析

# 第6章 数据可视化

## 6.1数据大屏

## 6.2数据分析

# 第7 结论

## 7.1论文总结

## 7.2存在问题

## 7.3未来展望