**个人作品报告**

|  |  |
| --- | --- |
| **课程名称：** | 数据采集 |
| **项目名称：** | 携程旅行酒店数据采集 |
| **学 号：** | 2021220440 |
| **姓 名：** | 阳健 |
| **所属学院：** | 理工农学院 |
| **专业年级：** | 21级软件工程四班 |

目 录

[**一、项目需求分析和设计** 2](#_Toc164878242)

[1.1项目背景 2](#_Toc164878243)

[1.2数据来源 2](#_Toc164878244)

[1.3数据范围 2](#_Toc164878245)

[1.4采集方式 3](#_Toc164878246)

[1.5数据存储 3](#_Toc164878247)

[1.6数据处理 3](#_Toc164878248)

[**二、项目功能和实现** 3](#_Toc164878249)

[2.1爬虫模块 3](#_Toc164878250)

[2.2数据解析模块 3](#_Toc164878251)

[2.3数据存储模块 3](#_Toc164878252)

[2.4数据处理模块 3](#_Toc164878253)

[2.5代码实现 4](#_Toc164878254)

[2.6 运行截图 6](#_Toc164878255)

[**三、使用技术** 6](#_Toc164878256)

[**四、项目发布或运行情况** 7](#_Toc164878257)

[4.1运行状态 7](#_Toc164878258)

[4.2问题和改进 7](#_Toc164878259)

[**五、项目总结** 7](#_Toc164878260)

[5.1 项目成果 7](#_Toc164878261)

[5.2 经验教训 7](#_Toc164878262)

[5.3 未来展望 7](#_Toc164878263)

# **一、项目需求分析和设计**

## 1.1项目背景

随着旅游业的蓬勃发展，酒店数据分析的需求日益增长。本项目旨在开发一个能够从携程旅行网站采集酒店数据的系统，为酒店行业提供数据支持，帮助企业进行市场分析、竞争分析和价格策略制定。

## 1.2数据来源

本项目数据采集目标网站为携程旅行（[https://www.ctrip.com/）。](https://www.ctrip.com/%EF%BC%89%E3%80%82)

## 1.3数据范围

主要采集以下酒店数据：

1. 酒店名称
2. 酒店价格
3. 酒店评分
4. 酒店区域
5. 酒店地理位置
6. 酒店设施服务
7. 酒店标签

## 1.4采集方式

本项目采用爬虫技术进行数据采集，通过模拟浏览器访问携程网站，解析网页内容提取所需信息。

## 1.5数据存储

采集到的数据将存储在csv文件中，方便后续的数据分析和处理。

## 1.6数据处理

对采集到的数据进行清洗、去重和格式转换，确保数据的准确性和一致性。

# **二、项目功能和实现**

## 2.1爬虫模块

使用Python编程语言开发爬虫程序。

使用DrissionPage 库模拟浏览器行为，绕过网站的反爬虫机制。

## 2.2数据解析模块

代码使用 Python 字典和列表操作来解析 JSON 数据，提取所需信息。

## 2.3数据存储模块

使用csv文件进行数据的存储。

## 2.4数据处理模块

使用Pandas库进行数据分析和处理。

开发数据清洗算法，去除重复数据和异常数据。

## 2.5代码实现

*"""  
[模块使用]:  
 DrissionPage  
 csv  
 time  
"""*import time  
  
# 导入自动化模块  
from DrissionPage import ChromiumPage  
# 导入csv模块  
import csv  
  
# 创建文件对象  
f = open('data.csv', mode='w', encoding='utf-8', newline='')  
csv\_writer = csv.DictWriter(f, fieldnames=[  
 '酒店',  
 '评论',  
 '价格',  
 '城市',  
 '区域',  
 '地址',  
 '距离',  
 '纬度',  
 '经度',  
 '评分',  
 '评价',  
 '环境',  
 '卫生',  
 '服务',  
 '设施',  
 '标签',  
])  
csv\_writer.writeheader()  
  
# 打开浏览器  
dp = ChromiumPage()  
# 监听数据包  
dp.listen.start('json/HotelSearch')  
# 访问网站  
dp.get('https://hotels.ctrip.com/hotels/list?countryId=1&city=2&checkin=2024/05/01&checkout=2024/05/06&optionId=2&optionType=City&directSearch=0&display=%E4%B8%8A%E6%B5%B7&crn=1&adult=1&children=0&searchBoxArg=t&travelPurpose=0&ctm\_ref=ix\_sb\_dl&domestic=1&')  
  
# for循环下滑页面  
for page in range(1, 15):  
 print(f'正在采集第{page}页的数据内容')  
 if page >= 4:  
 # 获取所有匹配的元素  
 next\_page = dp.ele('css:.btn-box span')  
 # 判断是否出现点击按钮  
 if next\_page.text == '搜索更多酒店':  
 dp.wait.load\_start(2)  
 # 点击搜索更多酒店  
 next\_page.click('js')  
 # 等待数据包加载  
 resp = dp.listen.wait()  
 # 获取响应内容  
 json\_data = resp.response.body  
 # 解析数据，提取酒店信息所在列表  
 hotelList = json\_data['Response']['hotelList']['list']  
 # for循环遍历，提取列表里面的元素  
 for index in hotelList:  
 """ 提取具体数据内容，保存到字典里面 """  
 dit = {  
 '酒店': index['base']['hotelName'],  
 '评论': index['comment'].get('content'),  
 '价格': index['money']['price'],  
 '城市': index['position']['cityName'],  
 '区域': index['position']['area'],  
 '地址': index['position']['address'],  
 '距离': index['position']['poi'],  
 '纬度': index['position']['lat'],  
 '经度': index['position']['lng'],  
 '评分': index['score'].get('number'),  
 '评价': index['score'].get('desc'),  
 '环境': index['score'].get('subScore')[0].get('number')if index['score'].get('subScore') else None,  
 '卫生': index['score'].get('subScore')[1].get('number')if index['score'].get('subScore') else None,  
 '服务': index['score'].get('subScore')[2].get('number')if index['score'].get('subScore') else None,  
 '设施': index['score'].get('subScore')[3].get('number')if index['score'].get('subScore') else None,  
 '标签': ' '.join(index['base']['tags']),  
 }  
 # 写入数据  
 csv\_writer.writerow(dit)  
 print(dit)  
 # 下滑页面到底部  
 dp.scroll.to\_bottom()

## 2.6 运行截图

左边截图是正在进行数据采集

右边是通过url找到地址,然后监听json数据包，来获取数据



# **三、使用技术**

1. Python编程语言：用于编写网络爬虫程序和数据处理逻辑。
2. DrissionPage库：这是一个基于Chrome浏览器的自动化测试库，可以用来模拟用户操作，如点击、输入等，以及获取网页元素信息。
3. Csv模块：这是Python的标准库之一，用于读写csv文件。它被用来创建一个csv文件，并将爬取到的数据写入这个文件。
4. Time模块：这也是Python的标准库之一，提供了各种与时间相关的函数。，它可能被用来控制爬虫的爬取速度，或者计算程序运行的时间。
5. Pandas库：用来处理和分析数据。

# **四、项目发布或运行情况**

## 4.1运行状态

本项目已经完成开发和测试，并成功运行。通过运行程序，我们可以定期从携程旅行网站上采集酒店数据，并将数据存储到数据库中。采集到的数据可以用于后续的数据分析和应用开发。

代码已经上传至GitHub：https://github.com/oyj135/Python

## 4.2问题和改进

携程网站的反爬虫机制较为严格，需要不断调整爬虫策略。

数据清洗算法需要进一步优化，提高数据质量。

# **五、项目总结**

## 5.1 项目成果

通过本次项目，我们实现了对携程旅行网站上酒店数据的自动化采集，掌握了网络爬虫技术和数据处理方法。这个项目不仅提高了我们的编程能力，还为我们提供了宝贵的数据资源，为后续的数据分析和应用开发奠定了基础。。

## 5.2 经验教训

爬虫开发需要深入了解目标网站的反爬虫机制，并采取相应的应对措施。

数据清洗是数据分析的重要环节，需要投入足够的精力进行数据质量控制。

## 5.3 未来展望

扩展数据采集范围，采集更多类型的酒店数据。

集成数据可视化功能，提供更加直观的数据分析结果。

开发数据分析模型，挖掘酒店数据的潜在价值。