Technical Research on Crawling Website Visitors' Comments Based on Python

Chuanli Liu [1, 2, a], Lecai Cai [2], Xiang Gao [3]

[1]School of Automation & Information Engineering, Sichuan University of Science & Engineering, Zigong 643000, China;

[2]Yibin University, Yibin 644000, China;

[3]School of Mechanical Engineering, Sichuan University of Science & Engineering, Zigong 643000, China

[a]lcl9307@163.com

Abstract

There is a huge amount of complex information on tourism websites, but there is a problem of how to sift key information quickly through a clutter of information. In view of the problems of numerous comment information, this paper proposes an algorithm of intelligent crawling information, which is used to crawl the comment information of tourist attractions in Sichuan region of Ctrip by Python language.Meanwhile, aiming at the problem of anti-crawler mechanism in some websites, this paper puts forward a crawling method of accessing model browsers. The validity of the algorithm is verified through the test of crawling website information, which will help to run crawler work more efficiently.

Keywords

Comment information;Python; Crawler.

1. Introduction

With the development of times, people's consumption concept has also changed. More and more people pursue high-quality life, so there are a batch of tourists. According to The Main Data Report on Tourism Economy in the first half of 2018 which released on the official website of the Ministry of Culture and Tourism, in the first half of 2018, the national tourism consumption is booming, the tourism of the whole region focuses on a better life, the integration and innovation of tourism, culture, creativity and science and technology attracts much attention, and the trend of quality improvement and performance improvement is becoming more and more obvious.The number of domestic tourists reached 2.826 billion, an increase of 11.4% from a year earlier. The total number of immigration tourists reached 141 million, an increase of 6.9% from a year earlier.This shows that China's tourism industry is still in a stage of steady development.With the development of information technology, the internet information explosive increases[1].Before traveling, people will search the information about the tourist attractions from the website, and they will make a decision by looking at the previous comments, then the comment information is particularly important.Faced with a huge amount of information, users often find it difficult to make a decision, so a crawling algorithm is designed. The method of crawling information in this paper can effectively improve this situation.

2. Web crawler

2.1 Concept

Web crawler, also known as web spider or web robot[2], is a program or script that automatically crawls information on the World Wide Web according to certain rules[3].Web crawlers can be roughly divided into two categories: one is the general crawler which designed by the search engine service provider, and the other is the data crawler which obtain the required information on a specific URL.

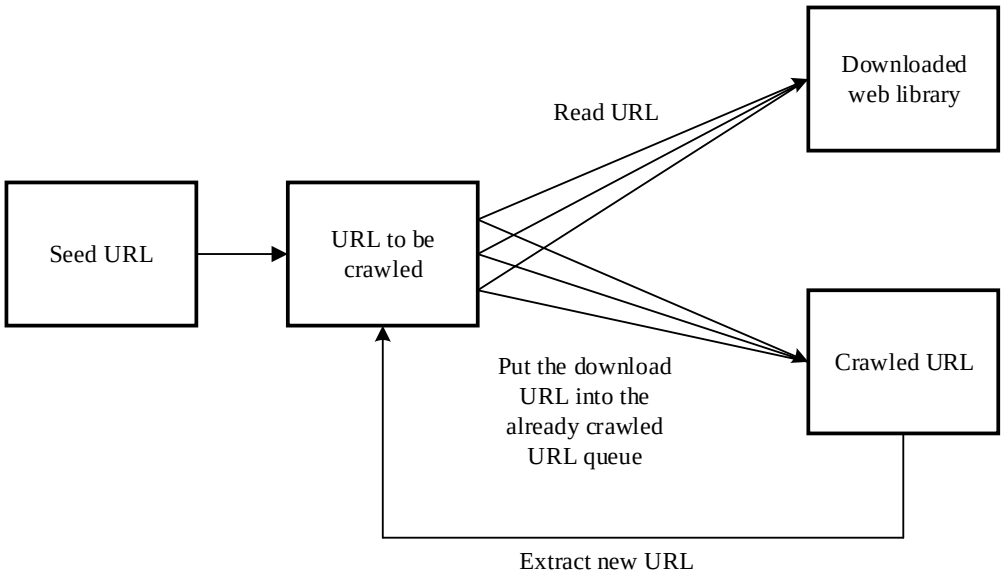
2.2 Python

Python, as a computer programming language, is an object-oriented scripting language with dynamic semantics [5].It is simple to use, syntactically concise, and has a wide variety of third-party function libraries [6].Originally designed Python primarily for automated scripting, which is increasingly used in large and stand-alone projects as versions are constantly updated.

Using Python to grab the interface of the web page itself is more concise than C++, Java . Compared with Shell, Perl, Python provides a more complete API to access web documents.Many web pages are not mechanized for stiff fetching, then you need to use simulated user login to browse the web, which can be easily solved by mechanize in python.The beautiful soap in Python can handle the text of the captured webpage very well, that is, it can handle most of the text processing with a few simple lines of codes.

2.3 Process

The general framework of web crawlers is shown in F.g 1.



F.g 1 General framework for web crawlers

It can be seen from Figure 1 that the workflow of the web crawler is as follows:

The first step selects a carefully selected seed URL;

Put the selected URL into the queue to be crawled;

Read the URL in the queue to be crawled, obtain the IP of the host, and save the webpage corresponding to the URL to the downloaded webpage. Finally put these URLs into the crawled URL queue;

Analyze the other URLs in the crawled URL, re-place these URLs into the URL to be crawled, and proceed to the next loop [7,8].

3. Web crawler design

3.1 Main technology

3.1.1 Pyquery

The pyquery library is a very powerful and flexible web parsing library that is often used in Python. pyquery allows jquery queries on xml documents and fast xml and html operations using lxml. Pyquery is faster and more usable than other parsing libraries.

3.1.2 Selenium

Selenium is a browser automation tool for automated web testing and web-based task management automation. Support multi-platform, multi-browser, multi-language to achieve automated testing. Selenium can operate the browser directly, exactly like the real user is actually working. These include actions such as clicking a link, entering text, and mimicking a mouse click.

3.2 Climbing experiment design

In order to verify the effectiveness of the crawling algorithm for a large amount of information crawling, ten pages of attraction information are selected for the characteristics of the Sichuan area attractions in the Ctrip. Each of the attractions does not exceed 300 pages of information.

The crawling method designed in this paper is to first crawl the first spot of the first page, then climb the first spot of the second page, the first spot on the third page, that is, the first one of all pages is crawled. After the attraction, climb the second attraction of all the pages, the third attraction, and so on. The main program code is as follows:

```
for start_item in range(start_items, end_items + 1):
for keyword in keywords:
url = 'https://piao.ctrip.com/'
html = get_one_page(url, keyword)
time.sleep(5)
parse_one_page(keyword, start_item, max_sights_pages, max_comment_pages, start_sights_pages,
start_comment_pages)
browser.quit()
```

First, the simulated user opens the ticket page of the Ctrip network to obtain the url of the scenic spot in Sichuan. The main implementation code is:

```
    def get_one_page(url, keyword):
        print('loading webpage information...')
        try:
            browser.get(url)
            pages_input = wait.until(
                EC.presence_of_element_located((By.CSS_SELECTOR, '#mainInput')))
            pages_submit = wait.until(
                EC.element_to_be_clickable((
                    By.CSS_SELECTOR, '#base_bd > div:nth-child(1) > div > div.main_right >
div.search_wrap.basefix > a')))
```

Then, by observing the source code of each attraction's webpage, we can find that the composition of the webpage source of each attraction is similar, and the difference lies in the difference of the digital code, so we obtain the URL representing each attraction page by the following procedure.

```
def parse_one_page(keyword, start_item, max_sights_pages, max_comment_pages,
start_sights_pages=1,
                   start_comment_pages=1):
```

```
        sights_url = browser.current_url

        print('attractions URL：{}'.format(sights_url))
        sights_html = browser.page_source
        pages_pattern = re.compile('<a href=".*?class="btn-last-page " data-reactid=.*?(\d+)</a>')
        sights_url_pages = re.findall(pages_pattern, sights_html)
```

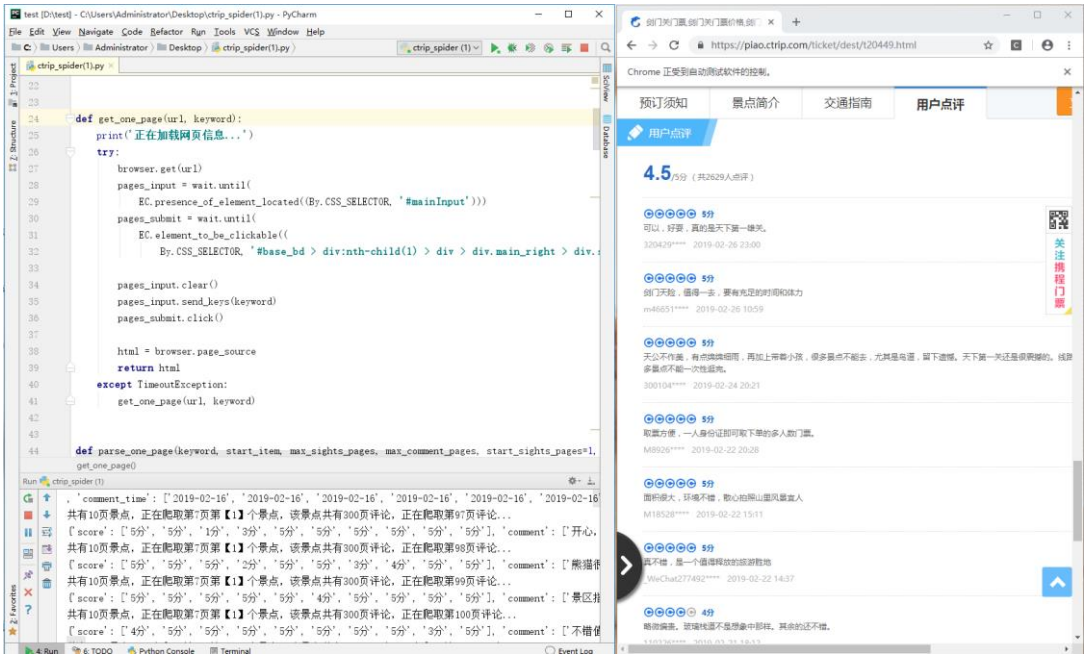
Finally, the webpage is parsed, and relevant information of the user comment in the webpage is captured, including the user name, the comment time, and the comment content. And save it as a txt document for future use.

```
comment = doc1('p').text()
score = doc1('h4').text()
name_and_time = doc1('div span').text().split()
people_name = name_and_time[1::4]
people_name = [people_name[i].replace('*', '') for i in range(len(people_name))]
comment_time = name_and_time[2::4]
```

4. Experimental results and analysis

4.1 Run page display

The page display of the entire program is shown in F.g 2.



F.g 2 Page display

As can be seen from the figure, the entire crawling process can be viewed through the window on the right side, which realizes the visualization of the crawling process. This is convenient for viewing and comparing the reptile process for errors, whether the crawled information is missing or not.

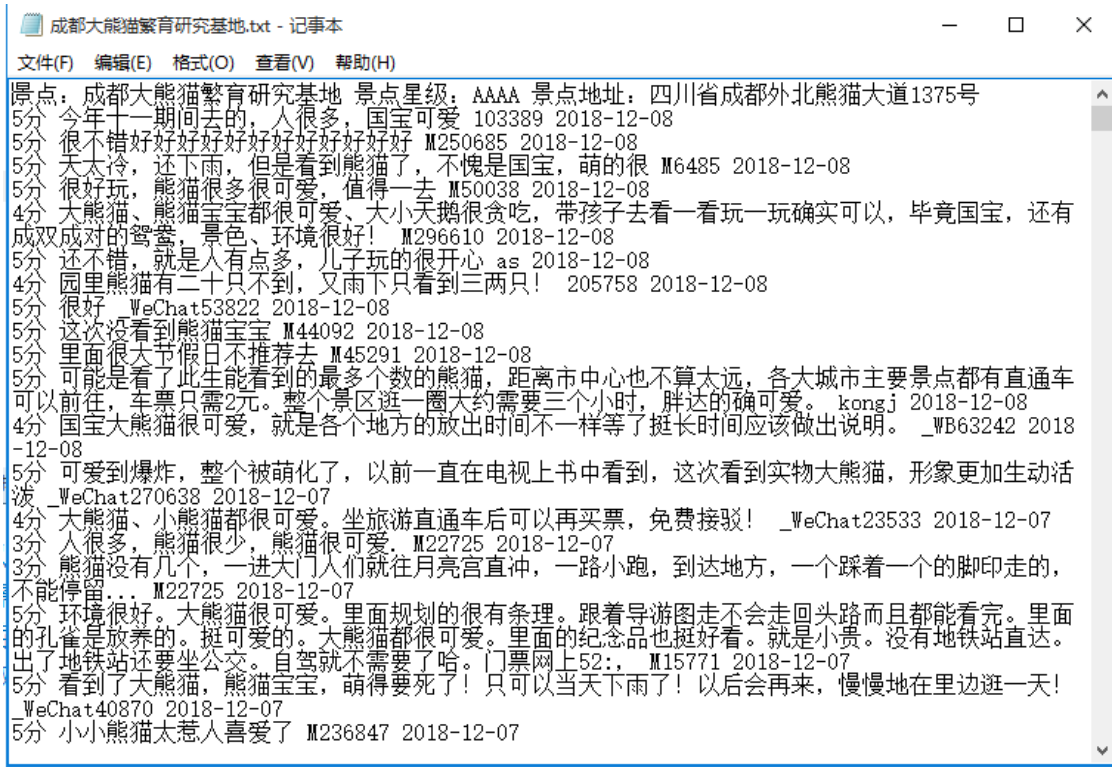
4.2 Txt document display

Save all the attractions you have crawled in a folder in the form of a txt file, as shown in F.g 3:

《放水大典·道解都江堰》.txt	8.4 KB	4.0 KB	文本文档
柏林童话.txt	25.2 KB	5.3 KB	文本文档
宝箴寨.txt	22.3 KB	10.1 KB	文本文档
毕棚沟.txt	273.1 KB	96.4 KB	文本文档
碧峰峡.txt	351.3 KB	116.3 KB	文本文档
碧峰峡野生动物园.txt	1.5 MB	82.9 KB	文本文档
成都大熊猫繁育研究基地.txt	338.6 KB	107.6 KB	文本文档
成都动物园.txt	823.0 KB	102.9 KB	文本文档
成都海昌极地海洋公园.txt	324.2 KB	100.3 KB	文本文档
成都欢乐谷.txt	306.7 KB	94.4 KB	文本文档
春熙路.txt	550.9 KB	100.2 KB	文本文档
翠云廊.txt	50.5 KB	10.3 KB	文本文档
大梁酒庄田园乐翻天乐园.txt	29.4 KB	10.2 KB	文本文档
叠溪 - 松坪沟风景区.txt	67.3 KB	26.7 KB	文本文档
都江堰景区.txt	355.8 KB	118.1 KB	文本文档

F.g 3 All attractions show

Open any txt document we can see inside the content as shown in F.g4, including the name of the attraction, the star rating, the location of the attraction, the user rating, the comment content, the user name, the comment time.



F.g 4 One of the attractions shows

As you can see from F.g4, the document contains the comments we need. By crawling a large amount of information designed to capture the work, this part can successfully verify the feasibility and effectiveness of the algorithm.

5. Conclusion

This paper designs a visual crawling method, using python as the basic language and using Selenium, Pyquery and other crawler technologies to crawl the comments of scenic spots in Sichuan region of Ctrip. A total of 200 scenic spots in Sichuan province and about 260,000 comments are

crawled.Finally the data saved as a TXT document for later using.The crawler which is carried out in the way of simulating user operation and the whole process of crawling can be seen intuitively, which is conducive to the smooth progress of crawler and the timely detection of errors.The designed crawling algorithm successfully acquired a large amount of comment data information of the website,which effectively helped tourists to quickly obtain the information that they needed from a large amount of information of the website.

Acknowledgements

Sichuan science and technology project(2019YFN0104).

References

[1] L.W. SUN, G.H. HE, L.F. WU: Research on the web crawler,Computer knowledge and technology, vol. 15(2010),p.4112-4115.
[2] J.W. LU:Design and implementation crawler of paper reference based on scrapy.Modern Computer, vol.03(2017) ,p .131-133.
[3] J.F. WANG,Y.PENG,M.WANG, et al.Sina microblog data capture technology based on web crawler, Management & Technology of SME,vol.01(2019),162-163
[4] L.R. GUO. Python-based web crawler design. Electronic technology & software engineering , 23(2017), p.248-249.
[5] Y. YUN. Design and implementation of web crab based on scrapy. Software development & application,09(2018), p.19-21+58.
[6] Q.R. JIA. Design and implementation of a python-specific web crawler. Computer knowledge and technology, 2017,p.12:47-49.
[7] J.Cho Crawling the web:Discovery and Maintenance of Large-scale Web Data.L.A.: Stanford University, 2002, 188.
[8] Q.Z. TAN, MITRA P.Clustering-based incremental web crawling.ACM Transactions on Information Systems, 28(2010)No. (4) ,p.85-89.
[9] HTTP threats evade normal protections.Computer Fraud & Security, Vol.9(2013) p. 132-136.