Design of Data Mining of WeChat Public Platform Based on Python

Fucheng You¹, Hechen Gong¹,* , Xinxin Guan¹, Yue Cao¹, Chenwei Zhang¹, Shuren Lai¹ and Yangze Zhao¹

¹Beijing Institute of Graphic Communication. No. 1, Xinghua Street (two sections), Daxing District, Beijing
Email: gonghechen123@163.com

Abstract. The 21st century is the artificial intelligence age; the new media in this condition quickly appeared people's eyes and become the information dissemination and the exchange important platform. In the 2011, people went into the "WeChat age", such as how to obtain the data in the WeChat public platform for interesting data has become the focus of the current research. This article is mainly aimed at the WeChat of the public platform Crawler search technology. First of all, the paper introduces the importance of web crawler in the age of data explosion, and then in the second part of the paper summarizes the domestic and foreign micro-trust Platform Network Crawler technology research, and then a brief overview of the network crawler related theory, working principle and Python in the field of the development of web crawler advantages. Finally, a Python-based WeChat public platform reptile system was designed to successfully obtain data. The crawled data can be used for market research and data analysis as raw data for machine learning and data mining.

1. Introduction
In recent years, under the background of big data, the rapid development of mobile Internet has brought great changes to people's lives. The data in [1] shows. At present, WeChat is one of the most popular information push social networking platforms. There are more than 600 million registered users and the number of registered WeChat public platforms developed by the WeChat team has reached 8 million. As many as possible, the WeChat public platform is a brand-new service platform that provides individuals, businesses and organizations with business services and user management capabilities. In the public number, we can push users a variety of content, such as voice, pictures, text or articles, etc. The literature [2] pointed out that the daily data produced by the WeChat public platform is huge, and there are huge social benefits behind the massive WeChat data. Therefore, it is of great significance to study the data mining technology of the WeChat public platform. It is necessary to perform data mining and data analysis on the massive data of the WeChat public platform and present it to users through data visualization.

In this context, this paper first studied the development status and dynamics of data crawling on the WeChat public platform at home and abroad, and then designed a crawler program to search WeChat partners via request, pyquery, phantomjs + webdriver, beautifysoup, etc. The engine (http://weixin.sogou.com/) sends an instant request to parse the crawl data in real time, using the pyExcelerator toolkit to save it locally to an Excel document. Finally, using the python visualization method to perform simple data analysis on the data, using the crawler program of this article can save the time of the data analyst and play a good role in filtering the massive data.
2. Research Status of Data Mining in Social Networks at Home and Abroad

In [3], we can know that in the research on new media, Twitter is the mainstream network communication platform in foreign countries. It is similar to domestic Weibo, and therefore its research on data mining is also the most. Although there is a certain difference between Twitter and the WeChat public platform, all of them are data mining for online media, and they are also technically conceivable. In [4], there are mainly two ways to crawl the current data. The first is to obtain data through the API interface, and the other is to crawl the network data through the crawler technology developed by the program development technicians to obtain the data. In [5] also affirms the first method. The shortcomings mentioned in [6] for obtaining data through the API are the need to obtain the right to use the API. However, in real life, it is very difficult to obtain access to the API. [7] proposed a web crawling and parsing page technique on the situation of needles and realized the situation in which ordinary users can autonomously crawl network data. In [8], when using the API to obtain data, there is a case where the acquired data is incomplete. In [9] proposes an article data acquisition scheme based on the article API and page parsing based on the issues raised in [8]. Currently, there are relatively few web crawlers for WeChat public platforms in China, and there are mainly two ways to crawl WeChat public platforms. The first method is to obtain the direct crawl of the WeChat public number article link in the WeChat app. The disadvantage of this method is that it is difficult to obtain the article address, and only for an article to obtain data, if you want to obtain a large amount of data, you must obtain multiple article addresses and then integrate the data. The workload is large. The second way is through obtaining the account and password of the WeChat public platform and obtaining data through the historical article link. However, the disadvantage of this approach is that ordinary users have difficulty obtaining the account and password.

3. Using python to Implement Related Technology of WeChat Public Platform

3.1. What is a Web Crawler

A web crawler (web spiders, web robots, In the FOAF community, it is more often referred to as a page chaser.) is a program or script that automatically grabs web information according to certain rules. The web crawler can simulate browser access to web resources. It can automatically collect all accessible web pages so that users can quickly search for the content they need.

3.2. Web Crawler Principle

The first step in the execution of the web crawler program is to set an address. The crawler will download the information content of the corresponding web page according to a specific path and strategy through its own function call and save it in the local file, which will also be in the current the next URL address entry is found in the web page. When the execution reaches the last URL address or meets the established requirements, the program automatically ends. However, when a user logs on to a website, login verification is generally performed. Therefore, it is a task that must be completed to implement web page login. As shown in Figure 1, it is the General Web crawler flow chart.

3.3. Python Language

Python is a high-level scripting language that combines interpretive, compiled, interactive, and object-oriented scripting. Python's design is very readable. Compared to other languages, English keywords are often used. Other punctuation marks in other languages have more distinctive grammar structure than other languages. The following are the advantages of Python:

- Short code size: Compared to the number of Java VS Python codes for the same function, Python is much smaller.
- More than one code: You can use the same code to handle different sizes of data, and concurrent user needs.
- Easy to learn: Python has relatively few keywords, a simple structure, and a well-defined syntax that makes learning easier.
- Free open source: Python open source, sharing era, make Python easier and faster.
• Standard Library: Python has a powerful and easy-to-use standard library. It is very compatible with cross-platform and makes programming easier.

The advantages of developing reptiles using python:
• Compared to the C++ language, it is more concise, and with less code, the crawler function can be implemented.
• Python provides a powerful framework.
• A variety of libraries are available, including web parsers.

The crawlers of the WeChat public platform in this article were debugged in the Python 3.6 environment.

![General Web crawler flow chart](image)

**Figure 1.** General Web crawler flow chart

4. Python Modules Related To Crawlers

4.1. URL Manager
Manage URLs to crawl and URLs that have been crawled. Send the URL to be crawled to the web page downloader. Delete the URL list.

4.2. Web Downloader
This module is the most central place for crawlers. Download the URL specified by the URL and save it as a string. This string is passed to the web parser for parsing. There are currently two tools in Python that support web downloads. The first is the lib2 package, and the other is the request third-party toolkit used in this article.

4.3. Web Parser
On the one hand, it will explain the valuable data; on the other hand, parse out the URL in the string and add it to the URL manager. These three modules form a loop. Only if there is a URL that is not crawled, this loop will continue.

As shown in Figure 2, it is a Python-based WeChat public platform web crawler flow chart.
5. Experimental Design And Process
The reptile design of this article is through the WeChat partner's Sogou search engine (http://weixin.sogou.com/), sending corresponding requests to indirectly crawl, sending instant requests to parse the data in real time and save it locally. The case of the paper designed to crawl is Beijing Institute of Graphic Communication. The English WeChat official seal number is bigc_xiaobao.

**Figure 2.** Python-based WeChat public platform web crawler flow chart

![Flow Chart](FlowChart.png)

**Figure 3.** Sogou WeChat interface

![Sogou WeChat Interface](WeChatInterface.png)
5.1. Get Homepage Entry Content

There are several ways to obtain the contents of the home page entry. You can use request, urllib, urllib2, or directly use webdriver + phantomjs, etc. You can use the request.get() method of this design to obtain the home page entry content.

The following is the Pseudocode for this section:

```python
def __init__(self, kw):
    self.kw = kw
    self.sogou_search_url = 'http://weixin.sogou.com/weixin?type=1&query=%s&ie=utf8&_sug_=n&_sug_type_=' % quote(self.kw)
    self.timeout = 5
    self.s = requests.Session()
```

Figure 4. Sogou WeChat Search Window

5.2. Search Entrance Address, Search for the Public Number as a Keyword

The following is the pseudo code for this section:

```python
def get_search_result_by_keywords(self):
    self.log('The search address is:%s' % self.sogou_search_url)
    return self.s.get(self.sogou_search_url, headers=self.headers, timeout=self.timeout).content
```

5.3. Obtaining the Public Number Home Link

From the obtained webpage content, get the public number homepage address. There are many methods for this step, beautifulsoup, webdriver, direct use of regularity, pyquery, etc. All you can do is use the pyquery method to find the public number homepage entry address.

The following is the pseudo code for this section:

```python
def get_wx_url_by_sougou_search_html(self, sougou_search_html):
    doc = pq(sougou_search_html)
    #print doc('p[class="tit"]')('a').attr('href')
    #print doc('div[class=img-box]')('a').attr('href')
    return doc('div[class=txt-box]')('p[class=tit]')('a').attr('href')
```

Figure 5. Entry address source code
5.4. Get A List of Articles on the Public Number Home Page

First you need to load the public number home page, which is phantomjs+webdriver, because the content of this home page needs JS rendering to load.

```python
def get_selenium_js_html(self, url):
    browser = webdriver.PhantomJS()
    browser.get(url)
    time.sleep(3)
    html = browser.execute_script("return document.documentElement.outerHTML")
    return html
```

Figure 6. WeChat public number home source code screenshot

5.5. Analyze Articles and Get Information

After the above steps, you first need to parse the list of articles, then find the entry address of each article through pyquery, enter each address, parse each article, and get the information we want. This design mainly through the above operation to complete the article name, address, profile, publication time and other information acquisition work.

The pseudocode for this section is as follows:

```python
title = article('h4[class="weui_media_title"]').text()
self.log('The title is: %s' % title)
url = 'http://mp.weixin.qq.com' + article('h4[class="weui_media_title"]').attr('hrefs')
self.log('The address is: %s' % url)
summary = article('.weui_media_desc').text()
self.log('Article summary is: %s' % summary)
```

5.6. Saving Data to Excel

Through the above implementation of the WeChat public number crawler function, access to the following information, screenshots of the acquisition information shown in Figure 7 and Figure 8.

Compared with the content of the original WeChat public account, we found that the title, address, content profile, date, and content were successfully captured.

Figure 7. Python run result graph
6. Conclusion
This article is designed for the data crawling on the WeChat public platform. Although there are still drawbacks to this design, the web crawler technology based on the WeChat public number must have a very high application value in the future. The WeChat public platform has become the mainstream nowadays, and it will generate a huge amount of data every day. Therefore, it is of great significance to analyze and organize these data.

7. Acknowledgment
This work was partially supported by Joint Funding Project of Beijing Municipal Commission of Education and Beijing Natural Science Fund Committee (KZ201710015010), Project of National Scientific Found (No.61370188), Project of Beijing Municipal College Improvement Plan (PXM2017_014223_000063) and New Project of Green Printing and Publishing Technology by Cooperative Creating Center (PXM_014223_000025) and BIGC Project (Ec201803 Ed201802 Ea201806)

8. References
[1] Syed K. T, Carson K. L, Juan J. C. Interactive Mining of Strong Friends from Social Network and its Applications in E-Commerce. Journal of Organizational Computing and Electronic Commerce, 2014, 242-243
[2] Yuan J, Yang J, Ding S. An Improved Pattern Matching Algorithm Based on BMHS. Distributed Computing and Applications to Business, Engineering & Science (DCABES), 2012 11th International Symposium on. IEEE, 2012: 441-445
[3] Java A, Song X, Finin T, et al. Why We Twitter: An Analysis of a Microblogging Community. 2009.
[4] Mannix J, Wilkes L, aly J. Pragmatism, persistence and patience: a user perspective on strategies for data collection using popular online social networks. Collegian, 2014, 21(2):127-133.
[5] Huang J, Xue J X. The predictive power of content and temporal features of posts in information dissemination in microblogging. Journal of China Tourism Research, 2015, 11(2):150-165.
[6] Zhiming Xu, Dong Li, Liu Ting, et al. Microblog Users’ Similarity Measures and Their Applications. Chinese Journal of Computers, 2014, 37(1):207-218.
[7] Bo Yang, Pengfei Zhao. Review of Recommended Algorithms. Journal of Shanxi University (Natural Science Edition), 2011, 34(3): 337-350.
[8] Yan X, Zhao H. Chinese Microblog Topic Detection Based on the Latent Semantic Analysis and Structural Property [J]. Journal of Networks, 2013, 8(4).
[9] Jiemin Chen, Yong Tang, Jianguo Li, et al. Research on personalized recommendation algorithm. Journal of South China Normal University (Natural Science), 2014 (5):8-15.