Design and Implementation of Network Data Reptile

Huawei Mei[1, a], Di Liu[2, b]

[1]School of North China Electric Power University, Baoding 071000, China;

[2]School of North China Electric Power University, Baoding 071000, China

[a]3094998@qq.com, [b]1246478371@qq.com

Abstract

With the rapid development of information technology, the global into a highly informative state, the network information resources show explosive growth, at the same time, the traditional means of information search has been far from being able to meet the needs of different industries, different positions of users, In order to improve the efficiency of Internet users search, web crawler as an important part of the search engine and the basis of its role is particularly important. This paper first introduces the background and significance of the research and the current research situation at home and abroad and the main contents of this paper. Then it introduces the basic concept of web crawler, the type of web crawler and the search strategy to use the web crawler system to extract and store the network data. Then it introduces the design and implementation of the web crawler system in detail, introduces the characteristics of the Python language used in the preparation of the web crawler, the advantages of the pycharm compiler and the urllib library and the tkinter graphical interface, and carries on the lottery record for the double color ball The crawler reptile is an example, and the urllib library is combined with the regular expression to perform the sub-string matching and the data crawling and subsequent storage function. Finally, the crawling result is analyzed and compared with the manual data. The time spent is compared and the objective conclusion is reached.

Keywords

Web crawler, Search strategy, Python, Sina aicai net, Regular expression.

1. Introduction

1.1 Background and significance of the subject research

With the development and popularization of the Internet, enterprises or individuals on the Internet information demand is also more and more dependent on engine search. Information and websites on the Internet are becoming more complex and numerous, with more than 1.6 billion sites on the Internet[1],and these indexed sites contain more than 21.7 billion pages[2], at the China Internet Network Information Center Information Center, CNNIC) 39th Report S3 , 2016 Year 12 Month China's search engine users reached 7.31 Billions[3], users need more after reasonable integration of data, the traditional search engine provides the results can not meet the needs of users. At this point, integration technology is critical.

Network reptiles[4] is an important means to integrate network data, is the cornerstone of new search engine technology, and plays an increasingly important role.

1.2 Research status at home and abroad

1.2.1 Overview of the current status of web crawlers

Web crawlers are divided into two types according to the implemented technology and system structure.[5]General Purpose Web Crawler, Focused Web Crawler, Incremental Web Crawler. The actual web crawler system is generally implemented by a variety of crawler technologies.

1.2.2 General Purpose Web Crawler

General Purpose Web Crawler[6] also known as Whole web crawler， The crawling object is expanded from some seed resource locators to the entire cyberspace, mainly collecting data for portal search engines and large network service providers.The crawling range and number of General Purpose Web Crawler[7]are very large，The crawling rate and storage space are very high. The order of crawling pages is relatively low. At the same time, because there are many pages to be updated, the parallel working method is generally adopted, but it takes a long time to update the page. Although there are certain deficiencies, it still has a strong application value.

1.2.3 Focused Web Crawler

Focused Web Crawler[8] also known as Theme web crawler， is a web crawler that selectively crawls pages related to a pre-set theme.Focusing on web crawlers [9] is a web page collection tool developed specifically for finding a specific topic. It does not require extensive coverage. Instead, it sets the goal to crawl web pages related to a specific topic content. A topic-oriented user search prepares the data foundation.

1.3 Subject research content

This paper mainly studies the focused web crawler in the web crawler. The web crawler mainly captures the historical lottery record of the Sina Love Color Network two-color ball, and writes the program in Python language using the PyCharm compiler. It mainly studies several aspects from the architecture, function modules and crawling results analysis of the crawler system.

2. Web crawler related theory

2.1 Web crawler overview

2.1.1 How Network Crawlers Work

Web crawlers are an automatic extraction tool for web pages and an integral part of Internet search engines. A traditional web crawler starts with one or several initial web resource locators and obtains the resource locator of the first page, and stops crawling the page until the condition set in advance is satisfied.

2.1.2 The role of web crawlers

Web crawlers download web pages from the World Wide Web for search engines and are a program that automatically extracts web pages. The emergence of web crawlers is the product of the network world of information explosion today. It is an important tool for different users to search network resources. It can effectively improve the user's search efficiency and the accuracy of search information, thereby improving the user's work efficiency. It has an irreplaceable position in today's information world and is an integral part of search engines.

2.2 Web crawler search strategy

2.2.1 Width-first search algorithm

The breadth-first search algorithm (also known as breadth-first search) is one of the easiest graph search algorithms. This algorithm is also the prototype of many important graph algorithms. Dijktra single-source shortest path algorithm and Prim minimum spanning tree algorithm are used. Similar to the breadth-first search. The breadth-first search algorithm is a node that traverses the tree along the width of the tree, and the algorithm aborts if the target is found. The design and implementation of this algorithm is relatively simple and belongs to blind search. At present, to cover as many web pages as possible, a width-first search method is generally used.

2.2.2 Depth-first search

The search strategy followed by depth-first search is to search for graphs as deeply as possible. In the depth-first search, for the newly discovered vertices, if it has an edge that is not detected as a starting point, it continues along this side. When all sides of node v have been explored, the search will go back to the beginning node where the found node v has that side. This process continues until all nodes that have been found to be reachable from the source node. If there are still undetected nodes, select one of them as the source node and repeat the above process. The entire process is repeated until all nodes are discovered. Depth-first in many cases can cause crawler trapped problems, so it is neither complete nor optimal.

2.2.3 Focus on search strategy

Search engines based on first-generation web crawlers typically crawl less than 1,000,000 web pages, rarely re-collecting web pages and refreshing the index. And its retrieval speed is very slow, generally waiting for 10s or even longer. With the exponential growth and dynamic changes of webpage information, the limitations of these general-purpose search engines are getting larger and larger. With the development of science and technology, the focus crawlers that target relevant web resources are born. Focusing on crawler crawling strategies only picks out a specific topic page, accesses according to the "best priority principle", and quickly and efficiently obtains more topic-related pages, mainly through content and web link structure to guide further Page crawling.

2.2.4 Search strategy based on content evaluation

The search strategy based on content evaluation mainly evaluates the value of the link based on the similarity between the topic (such as keywords and topic-related documents) and the link text, and determines its search strategy.The link text refers to the description text around the link and the text information on the link resource locator. The similarity is usually evaluated by the following formula:

$$\mathrm{sin}(\mathrm{d}_i, \mathrm{d}_j) = \frac{\sum_{k=1}^{m} w_{ik} \times w_{jk}}{\sqrt{(\sum_{k=1}^{m} w_{jk}^2)(\sum_{k=1}^{m} w_{jk}^2)}} \tag{1}$$

Where di is the feature vector of the new text, dj is the center vector of the jth class, m is the dimension of the feature vector, and wk is the Kth dimension of the vector.Because the web page is different from the traditional text, it is a semi-structured document, which contains many structural information. The web page does not exist separately. The links in the page indicate the interrelationship between the pages, so some scholars have proposed based on the link. A method of structural evaluation of link value.

2.2.5 Search strategy summary

Through the analysis of the various advantages and disadvantages of various search strategies, the research of web crawler search strategy is of great significance to the application and development of search engines. A good strategy is to get more topic-related pages with less network resources, storage resources and computing resources consumption within a reasonable time limit. Therefore, the strategies used by future web crawlers should be developed in terms of improving the accuracy of link value prediction, reducing the space-time complexity of computing, and increasing the adaptability of web crawlers.

2.3 Network data extraction and storage

This article uses Python language to develop, using Python language to write web crawlers. There are two common ways to extract and store network data: (1) scrapy framework + database storage (2) urllib library combined with regular expression + local txt document storage.The crawler system designed in this paper uses regular expressions to extract the substrings in the webpage source code, and stores the extracted data into the txt document. This scheme is more efficient and easy for users to operate.

ISSN: 2414-1895 DOI: 10.6919/ICJE.201908_5(9).0002

2.4 Chapter summary

This chapter introduces the related theories and techniques related to the web crawler system designed in this paper. It introduces the working principle of web crawlers, the types of crawlers and their advantages and disadvantages, the common search strategies of several crawlers, and the preparation of web crawler systems. The programming language (Python) used, the advantages and disadvantages of the Python language, the compiler used, and the advantages of the compiler, finally introduced the extraction and storage of regular expressions and network data.

3. Crawler system design and implementation

3.1 Introduction

With the continuous development of information technology, there are more and more lottery fans in China. In order to facilitate the analysis of the winning probability of lottery tickets for all kinds of lottery tickets, a crawling system that can capture the history of two-color ball history and save it to the local crawler system appears. Particularly important. Lottery enthusiasts can use this system to conveniently and concisely save the history of Sina Love Color Network two-color ball, no longer need to pay attention to and count the lottery record every day, which greatly facilitates lottery enthusiasts.

3.2 System Requirements Analysis

(1) Data download function: Ability to download specific data from web pages.(2) Web analytics: Extract page titles, keywords, and summaries, extract web links, and add new links to the resource locator queue.(3) Storage function: The data downloaded from the webpage can be saved to a file of a certain format in a correct format, so that the user can organize and analyze the data later.(4) Interface: It is convenient for users to use the crawler program, so that users can understand the function of the system at a glance.

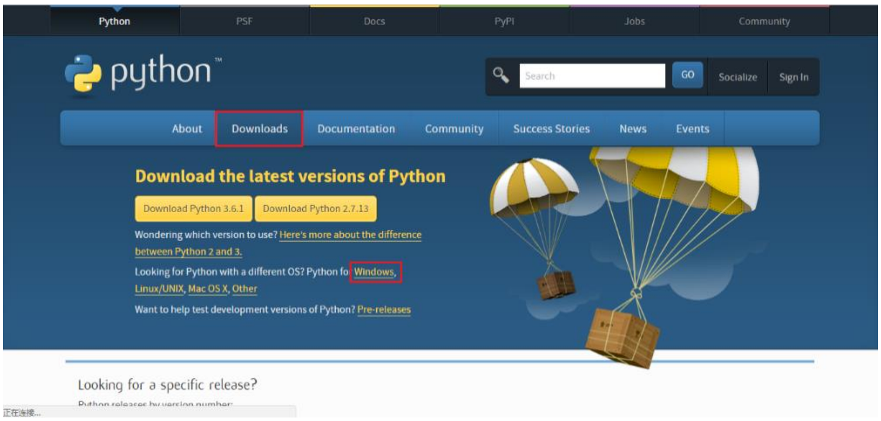
3.3 Experimental platform and development environment

Experimental platform and development environment

This system is written in Python 3.5 language under Windows 10 operating system. Select JetBrains PyCharm 2016.1.3 compiler for development. The hardware platform is: memory 6G, hard disk 656G (including solid state hard disk 256G, mechanical hard disk 400G), network bandwidth 4M.

Build a Python development environment

Since the operating system is Windows 10 64-bit, first go to the Python official website https://www.python.org to download the corresponding Python 3.5.1 installation package. Click to enter the windows version of the Python installation package link, as shown in Fig. 1.



Download the latest versions of Python

Wondering which version to use? Here's more about the difference between Python 2 and 3.

Looking for Python with a different OS? Python fo Windows, Linux/UNIX, Mac OS X, Other

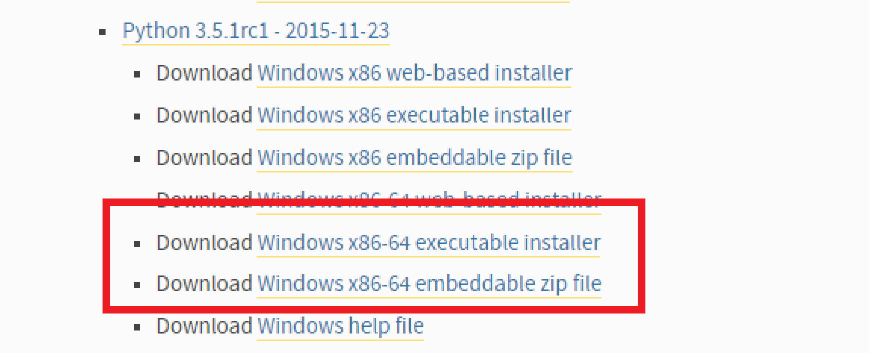
Want to help test development versions of Python? Pre-releases

Looking for a specific release?

Python releases by version number:

Fig. 1 Python official website to download the installation package

Select the 64-bit installation package, as shown in Fig. 2, click to download. After downloading to the local, double-click to open, install, check the add python to Path option, and add the Python installation path to the Path variable of the system environment variable. Then choose the appropriate installation path.



- Python 3.5.1rc1 - 2015-11-23
  - Download Windows x86 web-based installer
  - Download Windows x86 executable installer
  - Download Windows x86 embeddable zip file
  - Download Windows x86-64 web-based installer
  - Download Windows x86-64 executable installer
  - Download Windows x86-64 embeddable zip file
  - Download Windows help file

Fig. 2 Python 3.5.1 64-bit installation package

3.4 Programming language selection

3.4.1 Python language

As a high-level programming language, Python is very accidental, but it is the inevitable way for programmers to love it. Python's positioning is "elegant", "clear", "simple", so Python programs always seem easy to understand, beginners learn Python, not only easy to get started, but also in the future, you can write very very complex programs.

The advantages of the python language:

(1)As a newcomer to beginner Python, Python is very simple and very compatible with human natural language habits. (2)Easy to learn: Although Python is written in C language, it eliminates the very complicated pointers in c and simplifies the syntax of Python.(3)Python is one of FLOSS (free/open source software). Simply put, you are free to publish a copy of the software, read its source code, make changes to it, and use part of it for new free software. Python wants to see a better person create and improve often.(4)Portability: Due to its open source nature, Python has been ported to many platforms (it has been modified to work on different platforms).(5)Python supports both process-oriented function programming and object-oriented abstract programming. (6)Scalability and embeddability.

PyCharm compiler

(1)Strong real-time performance.(2)Can be edited when submitting changes.(3)Code review.(4)Refactoring. (5)Quickly view documents.(6) Docutils suppor.(7)Complete plugin system

3.4.3 Regular expression and urllib library

In 1956, the mathematician Stephen Klein designed a system of symbolic rules based on the work of the early nervous system, which is a regular set of rules. This system was quickly applied to the scanning design of the compiler by computer scientists. And lexical analysis. Regular expressions have powerful text processing capabilities, so they are quickly used in Unix-based operating systems, Perl, PHP, Delphi, JavaScript, C# (.NET), Java, Python, Ruby, and other languages and development environments.

A regular expression can be a regular notation or a regular notation. It is a single string to describe and match the result of a certain grammatical expression. The language used to describe a particular structure or rule of a string is executed by the relevant engine.

Regular expressions are widely used to extract substrings, manipulate problems, and extract data.

In Python 3.X, there is only the urllib library, which can be seen as a merge of urllib and urllib2 in Python 2.X. The urllib library is a Python standard web request library that contains functions for network data requests, handling cookies, changing request headers and user agents, redirects, authentication, etc.

3.5 Based on urllib library and regular expression crawler technology

The web crawler system designed in this paper uses the urllib library to crawl with regular expressions.

3.5.1 Simulate login, set proxy

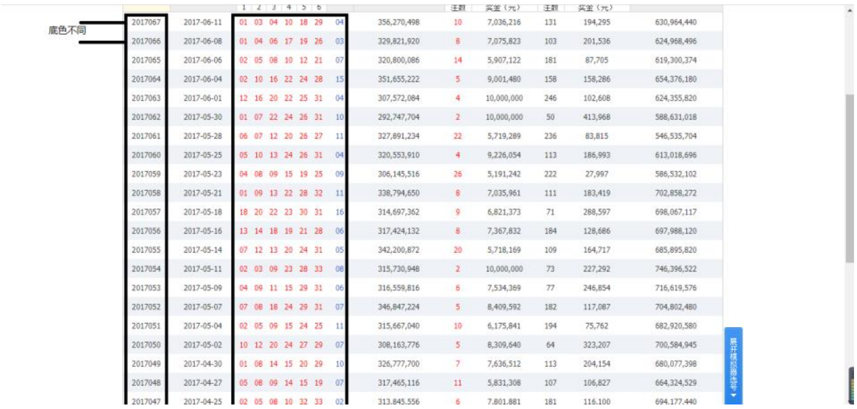
The getHtml() function was first designed because some websites must log in to crawl the data on the website, so use http.cookiejar.CookieJar to generate a cookie to simulate the user login and keep the login status in order to get the relevant content on the website. The function body uses the code cj = http.cookiejar.CookieJar() to create an empty cookie object, and then creates the http.cookiejar module to process the cookie so that the cookie is passed when the request occurs. Most websites don't want their data to be accessed and crawled by the crawler, so they will be set up to detect if the connection is a crawler, and if it is a crawler it will refuse to be accessed.In order for the crawler to successfully access the target site and crawl the required data, we must set up a proxy to simulate browser access, using the opener.addheaders=[('User-Agent','Mozilla/5.0 (Windows NT 6.1;WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2272.101 Safari/537.36'), ('Cookie', '4564564564564564565646540')] statement to simulate Google Chrome login. In addition, you need to install the opener as the global URL opener used by urlopen(). When you call urlopen() later, you will use the installed opener object. Once everything is ready, you can open the target site and recode the source code of the target site into a string for subsequent substring matching using regular expressions.

3.5.2 Web source import collection

The source code of the webpage is identified, and the whole part of the data is copied into the table collection. Observing the source code of the webpage can be seen that the data to be crawled is in the label, so the label character can be used to distinguish, <table class=" All strings between fzTab nbt"> and </table> are all put into the table collection, using table=html[html.find('<table class="fzTab nbt">'):html.find(' </table>')] can achieve the above functions.

The background color is different, and the lottery records are divided into two categories for crawling.

As the background color of the two-color ball lottery record in Sina Love Color Network is different, as shown in Figure 3-4, it can be seen that the background color of 2017067 and 2017066 is different, and the background color is alternately changed. The circled part of the figure is the data that is crawled. Take the recent 30 lottery records as an example, divide into odd and even parts, use different strings as the dividing standard, and use regular expressions to perform substring matching. For example, the latest two lottery records are the first one of the odd part and the first part of the even part.



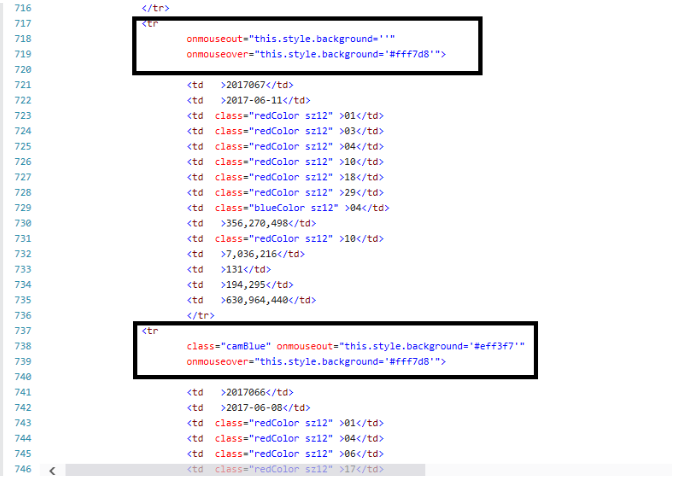
		1	2	3	4	5	6			注数	奖金（元）	注数	奖金（元）	
2017067	2017-06-11	01	03	04	10	18	29	04	356,270,498	10	7,036,216	131	194,295	630,964,440
2017066	2017-06-08	01	04	06	17	19	26	03	329,821,920	8	7,075,823	103	201,536	624,968,496
2017065	2017-06-06	02	05	08	10	12	21	07	320,800,086	14	5,907,122	181	87,705	619,300,374
2017064	2017-06-04	02	10	16	22	24	28	15	351,655,222	5	9,001,480	158	158,286	654,376,180
2017063	2017-06-01	12	16	20	22	25	31	04	307,572,084	4	10,000,000	246	102,608	624,355,820
2017062	2017-05-30	01	07	22	24	26	31	10	292,747,704	2	10,000,000	50	413,968	588,631,018
2017061	2017-05-28	06	07	12	20	26	27	11	327,891,234	22	5,719,289	236	83,815	546,535,704
2017060	2017-05-25	05	10	13	24	26	31	04	320,553,910	4	9,226,054	113	186,993	613,018,696
2017059	2017-05-23	04	08	09	15	19	25	09	306,145,516	26	5,191,242	222	27,997	586,532,102
2017058	2017-05-21	01	09	13	22	28	32	11	338,794,650	8	7,035,961	111	183,419	702,858,272
2017057	2017-05-18	18	20	22	23	30	31	16	314,697,362	9	6,821,373	71	288,597	698,067,117
2017056	2017-05-16	13	14	18	19	21	28	06	317,424,132	8	7,367,832	184	128,686	697,988,120
2017055	2017-05-14	07	12	13	20	24	31	05	342,200,872	20	5,718,169	109	164,717	685,895,820
2017054	2017-05-11	02	03	09	23	28	33	08	315,730,948	2	10,000,000	73	227,292	746,396,522
2017053	2017-05-09	04	09	11	15	29	31	06	316,559,816	6	7,534,369	77	246,854	716,619,576
2017052	2017-05-07	07	08	18	24	29	31	07	346,847,224	5	8,409,592	182	117,087	704,802,480
2017051	2017-05-04	02	05	09	15	24	25	11	315,667,040	10	6,175,841	194	75,762	682,920,580
2017050	2017-05-02	10	12	20	24	27	29	07	308,163,776	5	8,309,640	64	323,207	700,584,945
2017049	2017-04-30	01	08	14	15	20	29	10	326,777,700	7	7,636,512	113	204,154	680,077,398
2017048	2017-04-27	05	08	09	14	15	19	07	317,465,116	11	5,831,308	107	106,827	664,324,529
2017047	2017-04-25	02	05	08	10	32	33	02	313,845,556	6	7,801,881	181	116,100	694,177,440

Fig. 3 Sina love color network screenshot

3.5.3 Odd part

The odd part uses the table.split function to use the '<tr\r\n\t\t onmouseout=' string as the standard for partitioning, as shown in the source code in the first black box in Fig. 4. The code is divided into 16 parts and stored in the temp1 table. The string before '<tr\r\n\t\t onmouseout= ' is temp1[0], and the last string is temp1[15]. After temp1 removes the first element and assigns it to

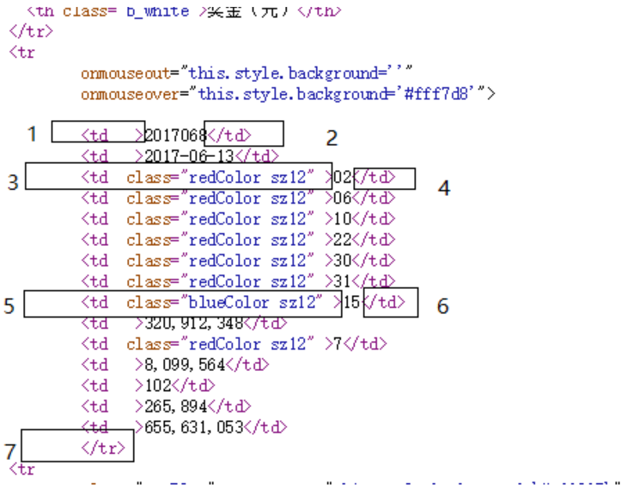
allTmp1, and then uses the string '</tr>' as the division criterion, the string between the two black boxes in Fig. 4 is saved to the tr1 set, code number1 =tr1.split('<td >')[1].split('</td>')[0] assigns the string part in the middle of black box 1 and black box 2 in Fig. 5 to number1, this When number1=2017067, it is a string type, this is the issue number we need to grab.



```
                </tr>
                <tr
                        onmouseout="this.style.background=''"
                        onmouseover="this.style.background='#fff7d8'">

                        <td   >2017067</td>
                        <td   >2017-06-11</td>
                        <td  class="redColor sz12" >01</td>
                        <td  class="redColor sz12" >03</td>
                        <td  class="redColor sz12" >04</td>
                        <td  class="redColor sz12" >10</td>
                        <td  class="redColor sz12" >18</td>
                        <td  class="redColor sz12" >29</td>
                        <td  class="blueColor sz12" >04</td>
                        <td   >356,270,498</td>
                        <td  class="redColor sz12" >10</td>
                        <td   >7,036,216</td>
                        <td   >131</td>
                        <td   >194,295</td>
                        <td   >630,964,440</td>
                        </tr>
                <tr
                        class="camBlue" onmouseout="this.style.background='#eff3f7'"
                        onmouseover="this.style.background='#fff7d8'">

                        <td   >2017066</td>
                        <td   >2017-06-08</td>
                        <td  class="redColor sz12" >01</td>
                        <td  class="redColor sz12" >04</td>
                        <td  class="redColor sz12" >06</td>
                        <td  class="redColor sz12" >17</td>
```

Fig. 4 Web page source screenshot 1

Redtmp1=tr1.split('<td class="redColor sz12">') takes the string in the black box 3 of Fig. 5 as the segmentation criterion, and removes the first and last unneeded elements, ie, Figure Fig. 5 The string between black box 3 and black box 5 is stored in reds1. The for loop is used to take out the elements in reds1, and the string in black box 4 in Fig. 5 is used as the segmentation criterion. After the first element, redstr1.split('</td>')[0], the string is the number of a blue ball. After all the elements in reds1 are traversed in turn, all the blue ball numbers can be obtained, and the blue ball number is output in the form of result1_=result1+'blue ball'+blue1+ "\n", for example: 2017068 lottery number: 02, 06,10,22,30,31, basketball: 15. Inventory[i] = result1_ store the odd lottery record from inventory[0], the growth step of i is 2.



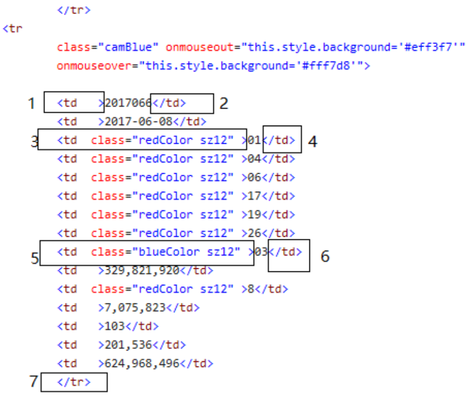
```
  <th class="b_white" >奖金（元）</th>
</tr>
<tr
        onmouseout="this.style.background=''"
        onmouseover="this.style.background='#fff7d8'">

        <td   >2017068</td>
        <td   >2017-06-13</td>
        <td  class="redColor sz12" >02</td>
        <td  class="redColor sz12" >06</td>
        <td  class="redColor sz12" >10</td>
        <td  class="redColor sz12" >22</td>
        <td  class="redColor sz12" >30</td>
        <td  class="redColor sz12" >31</td>
        <td  class="blueColor sz12" >15</td>
        <td   >320,912,348</td>
        <td  class="redColor sz12" >7</td>
        <td   >8,099,564</td>
        <td   >102</td>
        <td   >265,894</td>
        <td   >655,631,053</td>
        </tr>
<tr
```

Fig. 5 Odd number to be fetched

3.5.4 Even part

The even part uses the table.split function to use the <tr\r\n\t\t class="camBlue" string as the partitioning standard, as shown in the source code in the second black box in Fig. 5. The source code is divided into 16 parts and stored in the temp2 table. The string before the <tr\r\n\t\t class="camBlue" string is temp2[0], and the last string is temp2[15]. After temp2 removes the first element, assign it to allTmp2, and then use the string '</tr>' as the division criterion, and save the string between the two black boxes in Figure 3-5 to the tr2 collection, code number2 =tr2.split('<td >')[1].split('</td>')[0] assigns the string part in the middle of black box 1 and black box 2 in Fig. 6 to number2, this When number2=2017067, it is a string type, this is the issue number we need to grab.

Redtmp2=tr2.split('<td class="redColor sz12">') takes the string in the black box 3 of Fig. 6 as the segmentation criterion, and removes the first and last unneeded elements, ie, Fig. 6 The string between black box 3 and black box 5 is stored in reds2. The for loop is used to take out the elements in reds2, and the string in black box 4 in Fig. 6 is used as the segmentation criterion. After the first element, redstr2.split('</td>')[0], the string is the number of a blue ball. After all the elements in reds2 are traversed in turn, all the blue ball numbers can be obtained, and the blue ball number is output in the form of result2_=result2+'blue ball'+blue2+ "\n", such as: 2017067 lottery number: 01, 03,04,10,18,29, basketball: 04. Inventory[j] = result2_ store the even lottery record from inventory[1], the growth step of j is 2.



```
        </tr>
<tr
        class="camBlue" onmouseout="this.style.background='#eff3f7'"
        onmouseover="this.style.background='#fff7d8'">

        <td   >2017066</td>
        <td   >2017-06-08</td>
        <td  class="redColor sz12" >01</td>
        <td  class="redColor sz12" >04</td>
        <td  class="redColor sz12" >06</td>
        <td  class="redColor sz12" >17</td>
        <td  class="redColor sz12" >19</td>
        <td  class="redColor sz12" >26</td>
        <td  class="blueColor sz12" >03</td>
        <td   >329,821,920</td>
        <td  class="redColor sz12" >8</td>
        <td   >7,075,823</td>
        <td   >103</td>
        <td   >201,536</td>
        <td   >624,968,496</td>
        </tr>
```

Fig. 6 Even number of strings to be fetched

3.5.5 Sort the crawl results and save them locally

Assigning initial values 0 and 1 to i and j respectively, and setting the same step size 2, you can sort the odd and even lottery records by the number of periods, using text_file = open("lottery record.txt", "w") The statement first creates a txt blank document named "Calling Record", and then uses the text_file.writelines(inventory) statement to write the contents of the inventory collection to the txt blank document.

3.6 Tkinter graphical interface library application

Tkinter is a graphical interface for TK in Python that facilitates graphical interface design and interactive programming. The advantages of tkinter are easy to use, easy to program, simple interface, and very compatible with Python. Tkinter is integrated in Python 3.x without additional installation; the disadvantage is the lack of suitable visual interface design tools, the need for code to complete the window design and element layout, and the inability to intuitively drag and drop controls like C#.

The crawler system designed in this paper uses tkinter graphical interface library for interface design, as shown in Figure 3-8. You can see that the interface mainly has title, target website URL, crawl information button, crawl content display box, and exit button.

To use the tkinter graphical interface library, first import the tkinter graphical interface library using the Python statement from tkinter import *, then use the Tk() method in the library to create the main window and assign the window name to root, using root.title(" The Chinese Sports Lottery Record Reptile Tools V1.0") statement sets the window title to "China Sports Lottery Record Reptile Tools V1.0".

The destination URL is displayed as a label. Use the statement label= Label(frame,text="destination URL: http://zst.aicai.com/ssq/openInfo", font=("Microsoft Yahei", 15), fg="blue").pack( ) Initialize a label and set the display to "Destination URL: http://zst.aicai.com/ssq/openInfo", 15th Microsoft Yahoo font, font color is blue, and use pack() Function for intelligent layout.

Use the text control to preview the lottery record crawled from Sina Love.com, and initialize a Text box with the text=Text(frame,width=35,height=6,font=("Microsoft Yahe",15)) statement. Set the width of the text box to 35 pixels, the height to 6 pixels, and display the content in the No. 15

Microsoft Yahoo font. The font color defaults to black, and the text box is automatically laid out using text.pack().

3.7 Design and implementation of system function modules

3.7.1 Web crawler architecture

The aggregate web crawler is divided into five parts: link analysis, web page classification, data storage, download module, and web page preprocessing[9]. The download module downloads the initial webpage from the network according to the initial uniform resource locator, stores the webpage in the webpage library, and then takes the downloaded webpage out of the webpage library, performs webpage preprocessing, analyzes the webpage, and denoises the webpage, Chinese word segmentation , extraction of web page meta information, etc.

Then, the webpage classifier is used to classify the webpage. If the webpage classifier is used, the webpage is subjected to a uniform resource locator interpretation, and the parsed domain name is stored in the domain name database, and the uniform resource locator is stored in the uniform resource locator library. In order to facilitate downloading the module to download the webpage. If the page is not in the topic category, return the page category meta information to the web library.

3.7.2 Web crawler workflow

First obtain the initial resource locator, crawl the webpage according to the initial resource locator, extract a new uniform resource locator from the crawled webpage and add it to the queue, and then evaluate the uniform resource locator and the webpage according to the relevance standard. If the condition is met, stop, otherwise continue to select the uniform resource locator repeatedly to crawl the webpage until the end of the condition is met. Design and implementation of SPIDER's various functional modules:

Download module: The download of the webpage is the main job of the web crawler. There is a module in the download module for scheduling the resource locator, and the module distributes the resource locator to each download thread according to the search strategy.

Web page preprocessing: used to find out the links that web pages contain. Some crawlers even use pattern matching methods. Focusing on web crawlers requires careful analysis of web pages. The preprocessing of web pages has a great impact on the subsequent analysis of web pages. Denoising is an important part of web page preprocessing. In addition to noise reduction, web page preprocessing also includes interpretation, Chinese word segmentation, page meta information extraction, stop word deletion, and even natural language understanding, such as lexical, syntactic analysis, etc.

Web page classification: The classification of web pages is mainly to judge the relevance of web pages to given topics. Semantically, a topic can be an article, a paragraph, a phrase or even a word, a concept. The scope of the topic concept can be large or small. The choice of theme is the basis for extracting subject information [10].

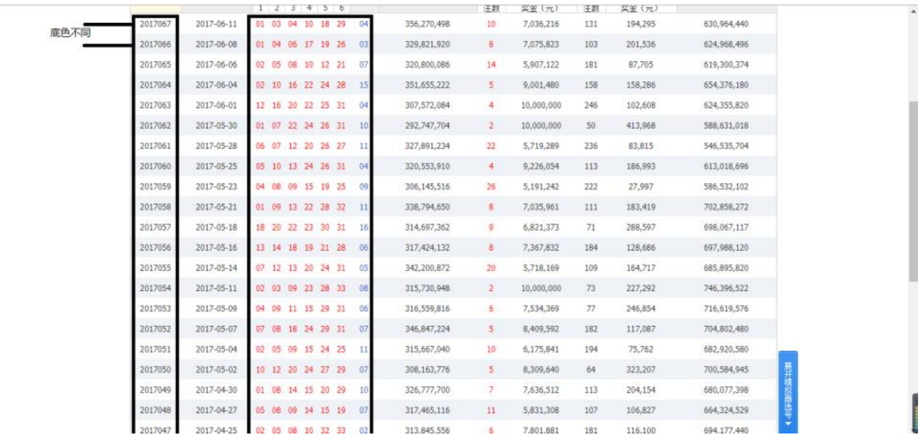
Link Analysis: Link analysis is the key to continuous crawling of web crawlers. The link analysis process first needs to get all the links, remove the dynamic links and obvious advertisement links, etc., then convert the relative resource locators into absolute resource locators, delete the specified links according to the labels and protocols, and finally score the remaining links according to the standard. Finally, the obtained link is stored in the database to be crawled.

Data storage: Web crawlers need to manage a large amount of data, including resource locators, system operating parameters, web pages, domain names, link ties between resource locators, file types, classification systems, and classification training sets. Usually manage these data with disk files.

4. Analysis and outlook

4.1 Analysis of crawling results of web crawler data

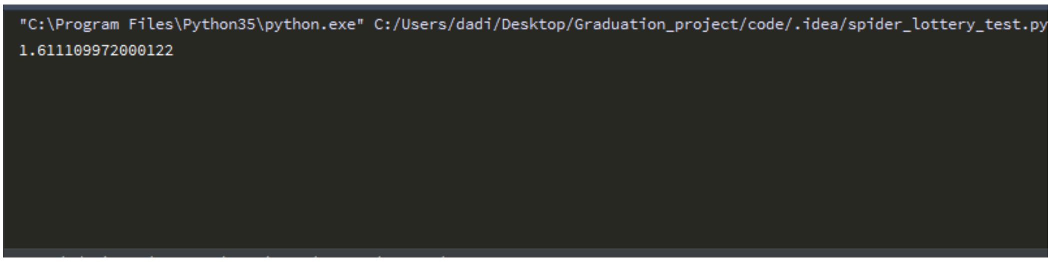
The web crawler system designed in this article saves the crawl results to the file where the program is located. After opening, you can see the data you have crawled. Compare the screenshots of the Sina Color Network two-color ball lottery record in Fig. 7. (The data in the two black boxes is the data that needs to be crawled.) You can find that the data you crawled is It is completely correct.



		1	2	3	4	5	6			注数	奖金（元）	注数	奖金（元）	
2017067	2017-06-11	01	03	04	10	18	29	04	356,270,498	10	7,036,216	131	194,295	630,964,440
2017066	2017-06-08	01	04	06	17	19	26	03	329,821,920	8	7,075,823	103	201,536	624,968,496
2017065	2017-06-06	02	05	08	10	12	21	07	320,800,086	14	5,907,122	181	87,705	619,300,374
2017064	2017-06-04	02	10	16	22	24	28	15	351,655,222	5	9,001,480	158	158,286	654,376,180
2017063	2017-06-01	12	16	20	22	25	31	04	307,572,084	4	10,000,000	246	102,608	624,355,820
2017062	2017-05-30	01	07	22	24	26	31	10	292,747,704	2	10,000,000	50	413,968	588,631,018
2017061	2017-05-28	06	07	12	20	26	27	11	327,891,234	22	5,719,289	236	83,815	546,535,704
2017060	2017-05-25	05	10	13	24	26	31	04	320,553,910	4	9,226,054	113	186,993	613,018,696
2017059	2017-05-23	04	08	09	15	19	25	09	306,145,516	26	5,191,242	222	27,997	586,532,102
2017058	2017-05-21	01	09	13	22	28	32	11	338,794,650	8	7,035,961	111	183,419	702,858,272
2017057	2017-05-18	18	20	22	23	30	31	16	314,697,362	9	6,821,373	71	288,597	698,067,117
2017056	2017-05-16	13	14	18	19	21	28	06	317,424,132	8	7,367,832	184	128,686	697,988,120
2017055	2017-05-14	07	12	13	20	24	31	05	342,200,872	20	5,718,169	109	164,717	685,895,820
2017054	2017-05-11	02	03	09	23	28	33	08	315,730,948	2	10,000,000	73	227,292	746,396,522
2017053	2017-05-09	04	09	11	15	29	31	06	316,559,816	6	7,534,369	77	246,854	716,619,576
2017052	2017-05-07	07	08	18	24	29	31	07	346,847,224	5	8,409,592	182	117,087	704,802,480
2017051	2017-05-04	02	05	09	15	24	25	11	315,667,040	10	6,175,841	194	75,762	682,920,580
2017050	2017-05-02	10	12	20	24	27	29	07	308,163,776	5	8,309,640	64	323,207	700,584,945
2017049	2017-04-30	01	08	14	15	20	29	10	326,777,700	7	7,636,512	113	204,154	680,077,398
2017048	2017-04-27	05	08	09	14	15	19	07	317,465,116	11	5,831,308	107	106,827	664,324,529
2017047	2017-04-25	02	05	08	10	32	33	02	313,845,556	6	7,801,881	181	116,100	694,177,440

Fig. 7 Sina love color network two-color ball draw record

Use the time library in the Python standard library to time the time spent crawling the last 30 lottery data of the web crawler. When the network is in good condition, run the crawler 100 times repeatedly to calculate the crawl data consumption. The average time is 1.611109972000122 seconds, as shown in Table 1 and Fig. 8. The average time spent manually counting a lottery record is about 10 seconds. It takes about 5 minutes to count the 30 lottery records. Therefore, using the web crawler designed in this article can undoubtedly save the time of two-color ball lovers.



```
"C:\Program Files\Python35\python.exe" C:/Users/dadi/Desktop/Graduation_project/code/.idea/spider_lottery_test.py
1.611109972000122
```

Fig. 8 Average time spent on data crawling 100 times

Table 1 Statistics of the operation time of the program 100 times

Reptile program run times	Total running time (seconds)	Average time in one run (seconds)
100	161.1109972000122	1.611109972000122

4.2 Outlook

Although the web crawler system designed in this paper has realized the function of crawling the Sina Love Color Network two-color ball lottery record, and can save the data to the local txt document, there are still improvements.

Due to the use of regular expressions for data crawling, the front-end design of different websites is not the same, so the crawler system designed in this paper can only crawl for the two-color ball lottery record of Sina Love Color Network. In the improvement of the subsequent crawler system, the crawler system can be improved into a tool that can crawl data for many websites.

This article crawls relatively little data, so it does not implement the crawler system to crawl data in parallel. But if you need to use the crawler system to crawl a lot of data, the crawler system is undoubtedly not suitable. In the perfect work of the subsequent crawler system, the function of parallel crawling data can be added to the crawler system.

The crawled data is saved to a local txt file for convenience. However, it is impossible to directly filter and count the data that is crawled. In the perfect work of the subsequent crawler system, you can store the data in the crawled data in order to operate on the data.

5. Conclusion

At present, information technology is developing rapidly, and society has entered a state of high informationization. At the same time, information on the Internet has also exploded, and traditional information search methods are far from being able to adapt to the current situation. In response to this phenomenon, a focused web crawler system was designed and developed to enable users to better collect the information they need and greatly improve the user's work efficiency. The web crawler system designed in this paper mainly focuses on the data crawling of Sina Love Color Network two-color ball lottery record.

In the design process of this web crawler system, targeted, efficient and easy to operate is the main advantage of the system. Using urllib library combined with regular expressions, sub-string matching on Sina love color webpage source code, has a strong pertinence.

The crawler system can capture 30 network data on average at 1.611109972000122 seconds and save the network data to the local TXT file.

Use tkinter graphical interface library for interface design, easy to understand, user-friendly.

The web crawler system can greatly facilitate the two-color ball enthusiasts to collect the two-color ball lottery data, and improve the user's efficiency so that the user can analyze the data. However, there are still some shortcomings in the actual use process, such as the ability to crawl the two-color ball lottery record of Sina Love Color Network, the function of crawling data in parallel without the crawler system, and the direct screening of the crawled data. , statistics, etc.

Acknowledgements

During this period of study, I was first my teacher, Huawei Mei. During the learning process, Teacher Mei patiently guided me. After work, I often helped me to comment on the paper. I would like to thank him for his guidance.Secondly, I would also like to thank my friends around me for helping me. I also thank the brothers who played basketball with me. When I am in trouble, I can take the initiative to ask me to go out to play and relieve my mood. Finally, I would like to thank my parents for their support of my education and my study and study.

References

[1] Netcraft: There are over 10 billion Websites in the World Wide Web (WWW). [EB/OL]. [2014-09-18]. http://tech.huanqiu.com/internet/2014-09/5142584.html.

[2] Livescience: The indexed web contains at least 46 billion pages (WWW). [EB/OL]. [2016-03-21].http: // www.cankaoxiaoxi.com/science/20160321/1105602.shtml

[3] CNNIC thirty-ninth Internet Report[EB/OL].[2016-01-22].http:// www.cac.gov.cn/ cnnic39/ index.htm

[4] Shokouhi M, Chubak P, RaeesyZ.Enhancing Focused Craw-ling with Genetic Algorithms[C] // International Conferenceon Information Technology:Coding and Computing (ITCC05)-Volume II.[s.l.] [s.n.],2005: 503-508.

[5] Li Shengwei, Yu Zhihua, Cheng Xueqi. Research Progress in Web Information Collection [J]. Computer Science, 2003.

[6] Sun Liwei, He Guohui, Wu Lifa. Research on web crawler technology: computer knowledge and technology, 2010.

[7] Wu Anqing, Zhang Yingjiang, Tu Jun.Study on the ROBOT Integrated Crawling Strategy of Subject Search[J].Journal of Wuhan University of Technology,2006,28(2):74-75.

[8] S. Chakrabarti, M. van den Berg and B. Dom. Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery [C]. InProceedings of the 8th International World Wide Web Conference, Toronto, Canada, 1999.

[9] Zhu Liangfeng. Research and design of the theme web crawler [D]. Nanjing, Nanjing University of Science and Technology, 2008.

[10] Wang Fengsong. A new generation of intelligent search engine - network code [J]. Network World, 1999, 13(2): 12-21.