International Refereed Journal of Engineering and Science (IRJES)
ISSN (Online) 2319-183X, (Print) 2319-1821
Volume 6, Issue 1 (January 2017), PP.62-67

Research on Key Technology of Web Reptile

Li Yuntao, Yu Su
(College Of Mechanical Engineering, Shanghai University of Engineer Science, China)

Abstract: This paper mainly introduces the web crawler system structure. Through the analysis of the architecture of web crawler we obtainedfive functional components. They respectively are: the URL scheduler, DNS resolver, web crawling modules, web page analyzer, and the URL judgment device. To build an efficient Web crawler the key technology is to design an efficient Web crawler, so as to solve the challenges that the huge scale of Web brings.
Keywords: DNS, URL, web crawler, web pages

II. NTRODUCTION

Web crawler is application which will retrieve the Web pages on the Web, usually as a search engine page collection subsystem. In the network initial stage, the network reptiles existed, and there are plenty of research achievements on web crawlernowadays. The first crawler is Google crawler, the crawler features mainly aimed at the different crawler components can respectively complete their own processes. In the process ofthe maintenance of separate URLserver, download URL set is necessary. Web access canalso be achieved by the crawlers. In the process of the index, the crawlers can also extract hyperlinks and keywords on the Web pages. In the process of solving processes, the URL should be able to realize the relative path to the absolute path of transformation.

At present, the commonly used on the market of engines such as Google and baidu, these engines crawlers technology are confidential. And the crawler realization strategies we are familiar with mainly are: breadth-first, Repetitive, definitions,deep crawling methods.

The basic work of theprocess of any Web crawler is similar, setting the initial seeds of the URL set S as input information of Web crawler, repeat the following steps: first of all, from the set S in some strategy to obtain a URL to crawl, parse out the URL in the host name of the IP address, using the HTTP protocol to connect the IP address of the Web server and download the URL corresponding page, download page after extracting all URLlinks of the page. For each of the URLlink, if it is a relative URL, convert it into absolute URL, and then determine whether the URL link has already downloaded or whether exists in the set S. For the URL link which has not been downloaded and doesn't exist in set S, add it to the set S[1].

Building a web crawler generally requires the following five features[2]: a collection of storage to grab the URL which is called URL scheduler.This URL scheduler can in some strategies select a URL from the URL set and provide it to the web crawler as a fetching goal;a DNS resolver, which is used toparse out the correspondingIP address according to the host name in the URL; a web page fetching module, using the HTTP protocol to download a URL of the page; a web page analyzer, responsible for extracting all theURL links or some information of interestfrom a web page;a URL judgement device,which can judge whether a given URL appeared before according to the URLaccess history.

According to the features of web crawler we can get its system structure, as shown in figure 1:

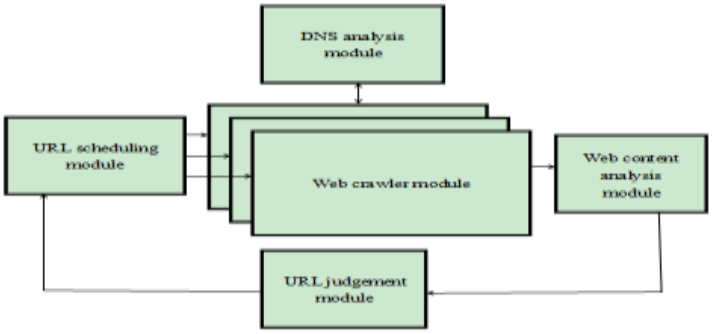

Figure 1 Web crawler architecture

We can see how each feature to cooperate to each other to complete the task of Web information

gatheringfrom the architecture of Web crawler. To design an efficient Web crawler it is essential to make close cooperation between each feature, and the ability to handle huge amounts of data on the Web is also needed. The following will be based on the web crawler features to study the key technologies which a highly efficient web crawler needs.

II. URL SCHEDULING

URL scheduler is mainly responsible for savingthe URL set which is to be grabbed. The URL scheduler provides two main operations: the one operation is to get a URL from the URL set to be fetched and provide it to the web crawler module to download. The other one is to add the URL link from the web page analysis module to the URL set to be fetched. The order of the URL to jointhe scheduler is determined by the external, but URL pop-up sequence is determined by the scheduler itself, that is to say the URL scheduler determines the order of the web crawler scraping of the page.

The simplest method toimplement the URLscheduleris to use a FIFO queue preserving all the URL sets to be grabbed. The URL order out of the queue is consistent with the order of their team. However, due tomost of the links a Web page contains are relative URLslocated on the same host, sothese relative URL links which are extracted by the web analysis module will enter the URL in order. Then they will leave the queue at the same time, this will make fetching thread continuous to grab the multiple URL links in the same host, causing a frequent access to the same Web server by the fetching thread at the same time, which can result in a denial of service attack.Such behavior is considered to be impolite, so the URL scheduler design requirements must meet the requirements of courtesy[3].

The order of the pop-up in a URL from the scheduler is URL scheduling strategy. It also determines the order of the web crawler scraping of the page. Therefore, the key of the URL scheduler is how to use a reasonable strategy to decide the order of the URL pop-up.Research on what strategies to grab web pages can be an important access to the Web page collection.

The usual scheduling strategy of the URL scheduler is to use a priority queue to store all the URL collection to be grabbed. The definition of priority is for web important degree evaluation. Define different priority rules will make different URLscheduling order, so the scraping of the page order is different, then the network eventually acquire different set of web pages. Judge strengths and weaknesses of a URL scheduling strategy, namely is to evaluate the quality of web page fetching strategy in web crawler. The standard of the evaluation is base on the sum of weight of the importance degree of the web page that the URLscheduling strategy obtains, and theURL scheduling policy which obtains the largest value is considered to be the optimal scheduling strategy.

III. DNSANALYSIS

DNS is the abbreviation of the domain name system. The system is used for naming hierarchy of the organization to domain of computer and network services. The host on the Internet is the 32-bit IP address. IP address is not easy to remember, however, the domain name is to facilitate the people to the IP address in memory. They conversion between them is called domain name resolution. Domain name resolution needs to be done by special DNS server. DNS is the server for domain name resolution.

DNS is a worldwide distributed service. When needs the DNS server to resolve a domain name, if the current DNS server cannot resolve this domain, it willrequest the upper layer DNS server to do. This process will recursively go on until the requested domain is resolved properly. Therefore, a DNS request may take several seconds or several dozen of seconds to complete. Through a browser to access the Web, DNS request will not cause too big effect.For Web crawler search engines, however, because of the need to frequently access the Web server, if DNS resolution time is too long,it will become the main bottleneck of Web crawler. So, for efficient Web crawler, we must solve the DNS bottleneck problem.

First of all, analyze the corresponding relationship between domain name and IP. In the Internet, domain name and IP address of the corresponding exists complex relationships, so a number of different URLs might point to the same physical pages[4]. In this case, if the web crawler to URL as grab goals, will lead to the same page repeated grab, causing the waste of resources. In order to solve this problem, we need to make clear of the relationship betweenthe domain name and its corresponding IP address.

For the relationship between Domain name and its corresponding IP address, there are four kinds of circumstances: one-to-one, one-to-many, many-to-one and many-to-many. One-to-one won't cause repeated grab, and the other three kinds of circumstances are likely to cause repeated grab.

That a domain name corresponds to multiple IP may due to the condition of the DNS rotation, and the situation is more common in commercial sites.Because the visit of commercial siteper unit time is big, it need to replicate server content, through DNS rotation to achieve load balancing, meeting the needs of a large

number of users to access at the same time.

The situation that Multiple domain names corresponds to one IP addressmay be due to the virtual host IP technology. The virtual host technology is the Internet server technology used to save server hardware costs. Virtual host technology is mainly used in HTTP service. This technology makes one or all of the contents of a server logically divide into multiple service units. It seems that there are several servers for visitors, so as to make full use of server resources. The contents of a plurality of virtual hosts, which are present in the physical server with the same IP address identification, are mutually exclusive, and each virtual host has its own independent domain name, but corresponds to the same IP address.

The situation that multiple domain names corresponds to multiple IPis due to thatmultiple domain names corresponds to a site, and each domain name corresponds to multiple IP address at the same time. This situation is equivalent to the above two kinds of domain name and IP address of the corresponding relationship.

Web crawler must solve the repeated grab caused by the relationship between the last three kinds of domain name and its corresponding IP address, which reduces fetching time, and savethe storage space of the page. In order to solve the problem, we must find the multiple domain names and its corresponding IP address which point to the same URL. This is a gradual process of accumulation. First accumulates to a certain number of domain name and IP address, and then grab the home page which the domain names and IP corresponds. Then grab the first few pages which links to the home page.Compare the results, if they are the same, it should be classified as a group. After the crawl we can only choose one of them to grab henceforth.

The second is the custom DNS.

Operating system usually provides a DNS interface. For example, UNIX operating system use the gethostbyname function to achieve this function. However the interface provided by the operating system usually are synchronized. When fetching threadusing this interface to analyze DNS, the thread can be blocked until the analysis of DNS is completed. This is the time to DNS spending accounted for 70% of the single grab work and become the main bottleneck of web crawler. So in order to improve web scraping efficiency, it is essential to customize a DNS resolver instead of the operating system to provide DNS interface.

Custom DNS resolver can use three ways to improve the efficiency of DNS.

The first is to customize a DNS client, parallel processing of multiple DNS parsing requests in multi thread mode, and this way can help the load balancingin more than one name server, avoid frequently sending the request to the same name serverwhich will lead to a similar denial of service attacks.

The second method is to use the cache server to save the NDS analytic results, this approach can greatly decrease the times of the DNS analysis.Only when encountering for the first time a new domain namedo the process ofDNS request, but after it all, obtain DNS results from the cache server. In actual use, to design a proper cache server refresh strategy, at the same time try to put the results of DNS cache in memory, so that we can consider using a dedicated machine as a caching server.

The third method is to use prefetching strategy. When the analysis moduleextract the URL links in theweb page, send them to DNS client to do the process of analysis, rather than waiting for the URL links to be scheduled to grab. This strategy makes it possible that when the URL is scheduled, the DNS parsing results may be stored in the cache server to avoid the actual DNS parsing and reduce the time of web crawling.

IV. WEB SCRAPING

Web crawler grabbing a Web page takesroughlyabout a few seconds. The time of this process include getting the target URL from the URL scheduler, DNS parsing, connecting to the Web server to download page, extracting all the URL links of the web page, and finally saving the page content and linkinformation to local disk.If the web crawlerdownloads each URL corresponding web pagesone by one, it is unable to grab a large number of web pages in a short time.This method is obviously not suitable for large-scale grid clustering task. There are two ways to solve this problem. One is with the method of multi-thread parallel fetching, another is to use single-threaded manner to grabasynchronously.

Web crawler in the process of scraping of the page, also must follow the Internet rules. If the administrator of this website make a statement thatsome content on the site is prohibited from the web crawler accessing, the web crawler is need to abide by the rules, otherwise will be considered to be unfriendly.This rule is known as the crawler prohibition agreement[5].

The multi-threaded parallel crawling mode is that the Web crawler starts multiple grab thread at the same time, each grab thread connecting to different Web server for Web scraping. This approach has the advantage that the program structure is simple, each thread fetching independently. Mercator web crawleradopts this way[6].

However, multi-threaded parallel crawling mode also can produce some performance cost[7]. First, the number of grab threads is limited by a web crawler memory address space, and the number of grab threads

directly determines the efficiency of web scraping. Second, each crawling thread needs access to some shared resources, which will bring the overhead of thread synchronization. Due to the performance bottleneck of the web crawler mainly lies in the DNS resolution,the network I/O when downloading the web pages and the disk I/O when storing the web information, so when the grab thread respectivelycomplete a crawl task and a web information storage, will lead to a large number of cross-random disk I/O. The external performance of the disk is constantly seeking, so the speed is very slow, and it has a great influence on scraping performance. This kind of defects can be improved through a single thread asynchronous grab way.

Google's web crawler and the Internet Archive web crawler adopt single thread asynchronous way tocrawl the web pages. Asynchronous mode is set non-blocking Socket. Because the calls of function connect, send, recvreturnsimmediately, not waiting for completion of the network operation[8], thus can create multiple continuously Socket to connect to different Web server, then check the status of these Socket by polling, once a Socket is ready, call the corresponding processing process to complete the preservation of Web information.

This single thread asynchronous grab way reflects more efficient storage management strategy. The disk I/O which is caused by web information storage works in order, in the Shared pool of tasks do not need synchronization method, avoiding the cross disk I/O caused by the disk constantly searching.This method also has the advantage of saving time and improving the performance of the web crawler. Moreover single threaded asynchronous crawl is easy to expand on multiple machines, so is the preferred way of distributed network.

Crawler ban protocol is a plain-text file which is created by the web site administratorunder the root directory of the web site.In this file, youdeclare the parts of the site that you do not want to be accessed by the web crawler. Web crawler need to consciously comply with the agreement, so web crawler and the web site can get along well, otherwisethe web crawler may be blacklisted by the webmaster as unfriendly.

When a Web crawler visitsa newweb site, it should be the first to check the Web server root directory of the robots.txt file.For each path name prefixof the file, the specified crawler should not be accessed.For example, the content of a robots.txt is:

```
# AltaVista Search
The user-agent: AltaVista Intranet V2.0 W3C Webreq
Disallow: / Out - Of - Date
#exclude some access - controlled areas
User-agent: *
Disallow:/Team
Disallow:/Project
Disallow:/Systems
```

The robots.txtstatement prohibits the AltaVista web crawler to access the content of the /Out - Of - Date directory.

V. WEB ANALYTICS

When the crawler crawls to a corresponding resource of URL, the web page analysis module will first determine whether the resource is the target type of the crawler.For the purpose of this article research is the focus of the HTML page. This can be determined according to the content-type field in the HTTP response header of the Web crawler by the web server.If the field value is text/HTML, it means that the resource is a HTML page, and goes on the next step, otherwise dispose of it.

For each HTML page, the page analysis module will judge whether the content of the page has occurred, because there may be different URL corresponding to the same web page information.That is the so-called mirror web page or reproduced[9]. The same content of web pages you just need to store once in a disk. This can not only reduce the cost of storage space, but also when these pages as a search engine search results will not return a lot of duplication. This process is called web pagede-replication.

After dealing with the web page de-replication, web page analysis moduleextracts URL links and some of the pages of interest information. This process is usually handled by regular expressions. Regular expression is a very powerful text analysis tools. It is widely used in JavaScript, PHP scripting language.The regular expression was introduced from Java SDK 1.4. Regular expressions with some special symbols (called meta-characters) represent certain features of a set of characters, and specify the number of matches. The text that contains a meta-character no longer represents a specific content, but rather forms a textual pattern that matches all the strings that match the pattern. By using regular expressions to extract the URLlinks, put these URL links into the UR scheduler, so that you can guarantee the web crawler run forever. At the same time, the web page analysis module can use regular expressions to extract the web page information of interest, such as the page title, the anchor text for the URL links, and so on. This information can be for the use of the search engine subsystems of indexing and retrieval.

After the Web page analysis module extracts all the URL links in the web page.Make sure that only when these URL links pass through the URL referee can they be added to the URL scheduler. Because these extracted URL links of the web page might have been downloaded. The operation of the URL referee can avoid the same web page be downloaded many times and waste of fetching time and storage space.

As the crawling process of the web spider goes on, the grabbed web pages are more and more and so is the number of the URL links.Every time when extracting a web URL from a grabbed web page, how to quickly perform a referee operation in a huge URL setis a key issue in the design of URL referees. Therefore, the URL referee must solve the mass data storage and efficient look-up operation in huge amounts of data.

In order to improve the efficiency of URL referee, it is necessary to put as many as grabbed URL links into the web crawler's memory. This can be achieved by computing a piece of fingerprint information for each URL instead of a URL string itself. Web crawler memory, however, no matter how large it is, finally cannot satisfy the increase of the number of the URL set. Fingerprints that can't be put into memory can be stored on disk. Only keepthe most frequently usedsubset of URL links in memory, through a least recently used update mechanism to maintain a subset of URL links in the memory.

Finding the URL fingerprint information is faster than searching for the URL character itself, because the URL fingerprint is of fixed length and the information type is of value. You can apply binary search on it to improve search efficiency. However, the scale of the URL set makes the search efficiency very difficult to guarantee. Solving the problem of finding information inhuge amounts of informationusually adopts the following two efficient lookup tables. One is the application of B tree data structure, the other one is the application of Bloom Filter data structure.

The B-tree[10] is a balanced search tree for external memory lookups. Each node in the tree is the same size as a disk block, so that each node can contain a lot of keywords. Each node in the B-tree can also have a lot of child nodes, which will lead to the B-tree height is very low. Only need a small amount of disk access, when performing a lookup, so it has a high efficiency. As the process of judging repeating URLneed to combine the use of internal and external memory,B-tree can efficiently complete the judge of URL repetition.

Bloom Filter is a kind of random data structure with high spatial efficiency. It uses an array to represent a collection succinctly. But when judging an element whether belongs to a set, it is possible to mistakenly identify elements that do not belong to this set as belonging to this set. This feature is also known as false positive. Therefore, Bloom Filter is not suitable for those"zero error"applications. In applications that can tolerate low error rates, Bloom Filter uses minimal error rate in exchange for a great saving storage space. In the process of judging URL repetition, in order to improve the search efficiency we can tolerate low error, because the errorcaused by a small amount of repetition can be got rid of in the process of creating the index page. Bloom Filter is perfect for the realization of the URL repetition judge device. Internet Archive adopts this way for the judging of URL repetition.

VI.CONCLUSION

This paper mainly introduces the five functional components in the web crawler technology. Through the five components can realize preliminary web crawler. But the information on the Web has the characteristics of heterogeneity and dynamic, due to the limitation of time and storage space, even the largest search engine cannot crawl back all the pages on the Web. So in the process of actual crawling pages we should have a theme to grab, only crawl some specificweb content, rather than all of the content are crawled over. So for web crawl technology, based on this paper, should further study on the theme web crawler.

REFERENCES

[1]. A.Heydon, M. Najork. Mercator. A Scalable, Extensible Web Crawler, In Proceeding of the 10th International World Wide Web Conference, April 2001

[2]. Marc Najork, Allan Heydon. High-Performance Web Crawling, Handbook of massive data sets (Kluwer Academic Publishers, Norwell, MA, 2002)

[3]. Yang Sun, Isaac G. Council, C. Lee BotSeer: An automated information system for analyzing Web robots. Eighth International Conference on Web Engineering, 2008

[4]. J.Cho, N.Shivakumar, and H.Garcia-Molina. Finding replicated web collections. In ACM SIGMOD, pages 355-366, 1999

[5]. Yang Sun, Ziming Zhuang, Isaac G. Councill, and C. Lee Giles.Determining Bias to Search Engines from Robots.txt. IEEE/WIC/ACM International Conferernce on Web Intelligence. 2007

[6]. Mike Burner. Crawling towards Eternity, Building an archive of the World Wide Web, Web Techniques Magazine, 2(5), May 1997

[7]. J. Cho and H. Garcia-Molina. ParallelCrawlers.In Proceedings of the eleventh international conference

on World Wide Web, pages 124-135, Honolulu, Hawaii, USA, May 2002

[8]. W. Richard Stevens, Stephen A. Rago. Advanced programming in the UNIX environment second edition(People's Posts and Telecommunications Press, 2006)

[9]. R. Baeza-Yates, F. Saint-Jean, and C. Castillo. Web structure, dynamics and page quality. In Proceedings of String Processing and Information Retrieval(SPIRE), volume 2476 of Lecture Notes in Computer Science, page 117-132, Lisbon, Portugal, 2002

[10]. Thomas H. Cormen Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduction to algorithms (Beijing: Machinery Industry Press, 2008), pages 262-276