Implementation of Web Data Mining Technology Based on Python

Guilian Feng*

Institute of Physics and Electronic Information Engineering, Qinghai Nationalities University, Qinghai, Xining 810000, China

*Corresponding author e-mail: fenglaoshi2021@qhmu.edu.cn

Abstract. With the arrival of the era of big data, people have gradually realized the importance of data. Data is not just a resource, it is an asset. This paper mainly studies the realization of Web data mining technology based on Python. This paper analyzes the overall architecture design of distributed web crawler system, and then analyzes in detail the principles of crawler's URL function module, crawler's web crawl function module, crawler's web page parsing function module, crawler's data storage function module and so on. Each function module of the crawler system was tested on the experimental computer, and the data information was summarized for comparative analysis. The main significance of this paper lies in the design and implementation of a distributed web crawler system, which, to a certain extent, solves the problems of slow speed, low efficiency and poor scalability of traditional single computer web crawler, and improves the speed and efficiency of web crawler in grasping information and web page data.

Keywords: Python Language, Data Mining, Web Crawler, Distributed Crawler

1. Introduction
In today's global Internet data show the number of the rapid growth of changing type, enterprises under the background of big data it can use the tool in the analysis of the data hiding a trend, a relationship, or some kind of model by data analysis technology, analysis the hidden relationships can provide effective prediction to the enterprise, this also is each enterprise maintain its core competitiveness of the effective means, in and abroad comparison domestic companies the ability of modern information office is not yet perfect, the domestic companies important data storage capacity has shortcomings, all these result in the lack of data information processing and sources in the production environment. If companies want to obtain a large number of effective data, they may have to rely on web crawler to collect it. At present, the efficient web crawler technology is of great strategic significance to cope with the increasingly fierce market competition of enterprises [1]. Crawler is a program that can automatically communicate with the website and obtain the content of the Web page. It is the core of Web information retrieval technology. Currently, crawlers can be divided into three categories according to different working mechanisms: directional crawler, universal crawler and focused crawler [2].
Since the 1990s, search engine has entered a stage of rapid development, and web crawler has gradually become the core component of search engine. Many famous people and companies at home and abroad have done a lot of research and application on web crawler. Mercator is mainly a distributed web crawler designed and implemented with Java programming language. Mercator mainly includes two main modules: protocol and processing. The task of the protocol module is mainly to obtain webpage information according to the HTTP protocol, while the task of the processing module is mainly to process the information obtained by the protocol module, analyze and process it [3-4].

This paper studies the related technologies involved in distributed web crawler, which mainly includes distributed technology and crawler technology. This paper then uses the distributed technology to complete the distributed characteristics of web crawler and realize an efficient and real-time distributed web crawler system.

2. Data Mining Technology Based on Python

2.1 Overview of Data Mining

(1) Data mining function

Over the years, the development of data mining has improved with the passage of time, and its data analysis methods can be used to discover prediction information in large data sets and identify data mining models and related object information [4]. In addition to collecting and managing data, data mining can also analyze and predict massive data information [5]. Its function is specific mining patterns that can be analyzed. The most common types of functionality and patterns in data mining are:

Data can be associated with classes. The acquisition of classes can be described as data characterization and data differentiation. Classify objects according to their properties. Data discrimination refers to the comparative analysis of one feature of a data set and features of several other similar objects [6]. Data characterization refers to the representation of sample numbers by various charts or lines.

The patterns generated from the sample data set are called high-frequency mining patterns. These schemas can interpret and correlate data. Correlation and association mean that two or more attributes of something have certain characteristics. That is, there are related attributes between the data [7]. An unknown feature is usually inferred from some existing correlations in a data set.

Classified prediction refers to the prediction of future data attributes by looking at historical data. For example, according to the historical use of Huabei to predict the future use of Huabei consumption limit. The main operation of classification is to first establish a model and use the model to classify the sample data set [8, 9]. There are many models that can be classified, such as decision tree model, neural network, etc.

According to the association rules of the data, the relevant features of the data object are judged, and the features with high correlation are put together. By designing related product groups, the probability of customers buying related items is also greatly increased.

Time series data series pattern mining refers to that the sample data set is a time series pattern, and the data behavior under the time series is analyzed to infer and predict the data and the future behavior state.

(2) Association rules

The purpose of correlation analysis is to explore the interrelationship and dependence between different things. Correlation refers to the interaction between two features, not causality. Association relationships can be divided into Association Rules and Sequence Pattern Mining. Association rules are not purposeful predictions.

The most common association rule is the shopping basket case, in which the merchant looks for the relationship between different items by the items the customer will put into the basket, and analyzes the shopping habits of the customer by knowing which items are frequently purchased simultaneously by the customer. This kind of correlation discovery can effectively help merchants to
carry out marketing promotion.

The purpose of association rules is to find the influence between different characteristics. If we compare X and Y to two characteristic terms, X and Y are disjoint item sets. X is the lead term, Y is the successor term. The purpose of an association rule is to quantify the relationship between X and Y. Association rules judge influence based on support degree and confidence degree. Support refers to the probability of simultaneous occurrence of X and Y feature items, and confidence refers to the probability of occurrence of Y feature after occurrence of X feature. These two metrics are used to measure the influence between X and Y.

The corresponding support degree of X and Y is:

\[ \text{sup}(X, Y) = \frac{\text{num}(XY)}{\text{ALL}} \]  (1)

The confidence degree of X to Y is:

\[ \text{con}(XY) = \frac{P(X | Y)}{P(Y)} \]  (2)

The experimental result of association rule analysis is to measure whether X feature has an effect on Y. How much? First of all, two thresholds need to be set to confirm the validity of the analysis. The setting of thresholds depends on different situations. However, only when the probability of a certain data item is greater than the threshold can the validity of the event be guaranteed. This probability refers to the support. Only when this threshold value is less than the frequency of simultaneous occurrence of X and Y, X and Y can have an interactive relationship, and this threshold value becomes the minimum support. Secondly, according to the previous correlation analysis, X can affect the occurrence of Y event only if the confidence is higher than this threshold, which is called the minimum confidence. So, the correlation between things we just need to determine the level of support and confidence between different things in different situations.

2.2 Distributed Crawler Architecture

(1) Functional modules

The operation of the distributed crawler is responsible for by the modules of URL queue, page crawling, page parsing, similarity analysis, page storage, task management, cluster management and agent service [10].

The URL queue is responsible for storing all URLs waiting to be captured, and these URLs will be stored in the queue in descending order of topic similarity. The page fetching module will crawl the page pointed to by the URL according to the specified URL. A proxy buffer is set in the page fetching module, which stores the proxy sent by the proxy service. The proxy used in the process of page fetching is an open proxy filtered by the proxy service module. When the page information is returned, the page parsing module is responsible for parsing the returned page, and extracting the information needed by the system, including the title of the page, all the URLs in the page, etc. After the page is parsed, the page content and the parsed information will be transferred to the similarity analysis module, which can analyze the topic similarity of the page and the URL in the page. The analyzed URLs are inserted into the URL queue. During the insertion process, the URL queue uses Bloom filter to filter out URLs that already exist in the queue or have been crawled. After that, the information parsed from the page and the content of the page will be cached, and when the crawler task is completed, the information will be stored in HDFS and database.

In the system, the operation of crawler takes the task as the unit, and the task management module is responsible for the management of all crawler tasks in the system. The running state of the crawler task can be divided into three types: the running task, the waiting task and the completed task. For the running task, the task management module will monitor and record the running state of the task in real time, including the time consuming, completion rate and other information. When the number of pages acquired by a crawler task reaches the standard, the task management module will take out a new task from the task queue and start running. At the same time, other threads are used to store the data
acquired by the previous task, so as to improve the system's concurrency during the handover of new tasks and new tasks. For the tasks waiting to run, the task management module adopts the task queue, and stores the tasks waiting to run in the queue according to the time sequence of task creation. In addition, there are multiple servers in the system, including crawler server cluster, HDFS cluster, database server, proxy server and so on. The availability and usage status of these servers is monitored by the cluster management module.

(2) Cluster planning
The data obtained from the Web can be divided into two broad categories: agent information and page information. The former provides agents for distributed crawlers, while the latter is the target of distributed crawlers [10].

The global physical node planning of distributed crawler includes a database server, a proxy server, a crawler master server and multiple crawler slave servers, an HDFS master server and multiple HDFS slave servers.

In this plan, the crawler master server, proxy server and database server have only one logical node, and they all communicate with each other, so they are put on one physical node. There is also only one HDFS primary server, but to avoid overloading Node 1, no HDFS primary server is deployed on Node 1. There are two distributed clusters in the plan, one is HDFS storage cluster, the other is crawler cluster. For these two clusters, the HDFS nodes and crawler slave servers are placed in pair-to-pair-to-pair-to-nth physical machines in section 2. In the mode of communication, the HDFS cluster using internal definition of RPC communication, between the crawler master server and database server, proxy server and database server use internal define TCP communications between the user and the crawler between the primary server, the crawler slave servers, crawler master-slave between server and proxy server is using the HTTP protocol.

3. Model Implementation and Training

3.1 Experimental Environment
The configuration of each physical machine in the development and operation environment of the distributed crawler in this experiment is shown as follows:

- CPU: Intel Core i7
- Hadoop: Hadoop 2.6.4
- The JDK: JDK1.8
- IDE: the IDEA of 2019

3.2 Configure SSH
The SSH service authentication method mainly has two kinds of the first kind of authentication method is password authentication, can send a request from the client to the server, first through secure encrypted user name and password for the remote server, and then on the remote server to decrypt the send come over get user name and password, then check if the same returns success, if you don't like will return failure. This second is the public key authentication, can be confirmed for the customer side through the form of the number of signatures, currently can be used in the system with RSA and DSA two methods for the number of signatures. A local client can send a request for a user name and public key to a remote service. The authentication process basically starts with the service checking the request for the public key and failing if it doesn't meet the requirements. If it does meet the requirements, then it can check the client of the service and return the relevant information in the form of a checkmark.

3.3 Test Scheme
(1) Performance testing
There are two main test targets for performance testing, which are the web crawler in the single computer environment and the web crawler in the distributed network environment. The specific test
scheme of the two objectives is to give a URL seed file to start crawling and then record the total number of web pages crawled; Give a URL torrent file and set the map and reduce numbers.

(2) Extended testing
An extensible test was designed according to the characteristics of the system. Hadoop was configured to add nodes and then the efficiency of crawler system to capture data was examined.

3.4 Test Data
First, set the number of threads to 20 and the system maximum to 100, since there are five Slaver nodes. Then set the maximum delay time to 15 seconds as appropriate, and set the Map number of Hadoop nodes to 8 and Reduce number to 2.

4. Test Results

4.1 Performance Test

As shown in Figure 1, the distributed crawler ran for a total of 13.53 hours and captured a total of 622.91MB, with an average grasping speed of 46.28. The minimum file size of distributed crawler is 48.72MB, the time is 0.97 hours, and the speed is 47.23. The maximum file size was 181.65MB, took 3.74 hours, and the speed was 50.54.

4.2 Extensibility Test Results
First of all, the number of nodes in the cluster was changed to 1, 3 and 5 respectively for testing, and then the total number and speed of data were recorded and the results were then summarized.

| Speed  | 1 Hour | 2 Hour | 4 Hour | 6 Hour |
|--------|--------|--------|--------|--------|
| 1 Node | 30     | 32     | 31     | 30     |
| 3 Node | 48     | 47     | 46     | 45     |
| 5 Node | 68     | 69     | 66     | 67     |

Table 1. Crawling speed at different node depths
As shown in Table 1 and Figure 2, as the running time gets longer, the crawler system becomes more and more stable, and the crawling speed of web crawler will eventually reach a stable value. The node changes in this cluster are almost identical to the final fetching speed changes. When the number of nodes in the cluster increases, the crawling speed of the web crawler will also increase. However, due to the influence of some other reasons, there is no linear relationship between the speed and the number of crawlers. According to the above analysis, it can be concluded that the web crawler implemented this time has the characteristics of better performance, higher performance and faster speed than previous crawlers.

5. Conclusions
In this paper, the basic principles and distributed knowledge of web crawler are analyzed in detail, and the distributed web crawler system is designed and implemented. Firstly, the principle and technology of distributed web crawler are described in detail. Design a distributed web crawler system and detailed design of the system architecture and the detailed design of each system function. The environment required by the distributed web crawler system is built on the experimental computer machine, and then the crawler system is tested on the computer with the built environment. The main significance of this paper is that the distributed web crawler system is designed and realized, and the system improves the shortcomings of traditional web crawler, and uses the distributed web crawler to deal with the situation of the crazy growth of network data.

References
[1] Elaraby M E , Moftah H M , Abuenlen S M , et al. Elastic Web Crawler Service-Oriented Architecture Over Cloud Computing[J]. Arabian Journal for Science & Engineering, 2018, 43(12):8111-8126.
[2] Ahram T Z , Nicholson D . [Advances in Intelligent Systems and Computing] Advances in Human Factors in Cybersecurity Volume 782 || Using Dark Web Crawler to Uncover Suspicious and Malicious Websites[J]. 2019, 10.1007/978-3-319-94782-2(Chapter 11):108-115.
[3] Nisha N , Rajeswari K . Study of Different Focused Web Crawler to Search Domain Specific Information[J]. International Journal of Computer Applications, 2016, 136(11):1-4.
[4] Weng Y , Wang X , Hua J , et al. Forecasting Horticultural Products Price Using ARIMA Model and Neural Network Based on a Large-Scale Data Set Collected by Web Crawler[J]. IEEE Transactions on Computational Social Systems, 2019, 6(3):547-553.
[5] Xu L, Jiang C, Wang J, et al. Information Security in Big Data: Privacy and Data Mining[J]. IEEE Access, 2017, 2(2):1149-1176.

[6] Helma C, Cramer T, Kramer S, et al. Data mining and machine learning techniques for the identification of mutagenicity inducing substructures and structure activity relationships of noncongeneric compounds[J]. J Chem Inf Comput, 2018, 35(4):1402-1411.

[7] Sorour S, Goda K, Mine T. Comment Data Mining to Estimate Student Performance Considering Consecutive Lessons[J]. Educational Technology & Society, 2017, 20(1):73–86.

[8] Saleh A I, Rabie A H, Abo-Al-Ez K M. A data mining based load forecasting strategy for smart electrical grids[J]. Advanced Engineering Informatics, 2016, 30(3):422-448.

[9] Ge D, Ding Z, Ji H. A task scheduling strategy based on weighted round robin for distributed crawler[J]. Concurrency & Computation Practice & Experience, 2016, 28(11):3202-3212.

[10] Zhang Y, Chen J, Liu B, et al. COVID-19 Public Opinion and Emotion Monitoring System Based on Time Series Thermal New Word Mining[J]. Computers, Materials and Continua, 2020, 64(3):1415-1434.