Research on Crawling Network Information Data with Scrapy Framework

Dashan Wang, Qingbin Zhang, and Shaoxian Hong
(Corresponding author: Shaoxian Hong)

Hainan College of Vocation and Technique, China
No. 95, Nanhai Avenue, Haikou, Hainan 570216, China
(Email: xianyi060760@163.com)

(Received Jan. 31, 2019; Revised and Accepted Feb. 18, 2020; First Online Feb. 16, 2021)

Abstract

In the Internet era of big data, the emergence of crawlers significantly improves information retrieval efficiency. This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm used to improve the performance of information crawling and classification. Then, the crawler software and clustering algorithm were programmed by the python software. Experiments were carried out using the MATLAB software in the LAN in a laboratory to test the Weibo data between October 1 and October 31. Moreover, a crawler software that adopted the scrapy framework but did not add the clustering algorithm was taken as a control. The results showed that the scrapy framework based crawler software could not achieve the same Weibo information classification as the actual classification whether the clustering algorithm was added or not; the crawler software that was added with the clustering algorithm was closer to the exact proportion in classification and obtained classification results with higher accuracy and lower false alarm rate in a shorter time.

Keywords: Clustering Algorithm; Crawler Software; Network Data; Scrapy Framework

1 Introduction

With the development of computer technology and the birth of the Internet, the speed of information generation has gained an explosive improvement [9]. Especially in recent years, with the popularization of 4G communication technology, the mobile Internet has been fully developed. After combining the mobile Internet and the traditional Internet, the generation and transmission speed of information data further increases.

The advent of the big data era makes people's life more convenient, which is mainly reflected in the fact that users can use more data to assist their different choices and service providers can optimize their services according to big data. However, the emergence of big data not only brings convenience but also brings difficulties. The excellence of big data is reflected in a large number of laws hidden in a large number of data, which can assist the decision-making of individuals or enterprises better. However, due to a large amount of data, data fragments that support different laws are scattered, and the method of human retrieval alone cannot meet the retrieval needs [2]. Crawler technology can replace manual search to retrieve big data and also can carry out preliminary classification of the retrieved data, which is convenient for mining the rules.

In order to crawl deep web pages, Feng *et al.* [13] designed an intelligent crawler with a two-stage framework. In the first stage, with the help of a search engine, the central page search based on the site is performed to avoid visiting a large number of pages. In the second stage, the most relevant links are mined to realize fast site search. The simulation results showed that the crawler could effectively retrieve the deep web interface in large-scale websites and obtained a higher harvest rate than other crawlers. Seyfi [10] proposed a focus crawler that uses specific HTML elements of a page to predict the topic focus of all pages in the current page that have unvisited links and verified the effectiveness of the method through simulations.

Huang *et al.* [5] put forward an extensible GeoWeb crawler framework that could search various GeoWeb resources and verified through simulations that the framework had good extendibility. This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm that was used for improving the performance of information crawling and classification. Then, the crawler software and clustering algorithm were programmed by the python software. Experiments were carried out using the MATLAB software in the LAN in a laboratory to test the Weibo data between October 1 and October 31. Moreover, a crawler software that adopted the scrapy framework but did not add the clustering algorithm was taken as a control.

2 Crawler Software Based on the Scrapy Framework

2.1 The Basic Structure of Crawler Software

The basic framework of the crawler software for network data crawling [6] is shown in Figure 1. In the overall structure, crawler software is divided into an interaction layer, logical business layer, and database layer. The interaction layer is the top layer of the software, responsible for the human-computer interaction with users.

The main content of the interaction layer is the design of the application form, which includes the main page module, task view module, server view module, and client view module. The main page module is responsible for querying the task information list and carry out various operations on the task. The task view module is the module for editing the task information, which can directly edit the task by visualizing the task data. The module is generally nested in the main page module. The server module is responsible for monitoring the user's use of the client. The client module is used by the user to view the software's connection to the server and to receive or release tasks.

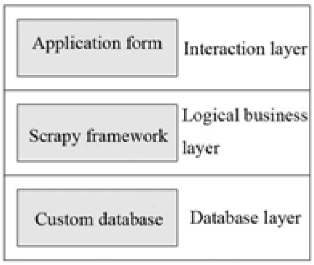

Figure 1: The basic framework of crawler software

Next is the logical business layer, which is mainly composed of a scrapy framework [1]. It is the functional core of the whole crawler software. Its main function in the software is to realize the task description submitted by the interaction layer, generate the corresponding crawler, download the network data in the given URL address, and summarize and count the string.

The last one is the database layer, whose main structure is a custom database. Its main function is to store or delete the network data searched in the URL address in the logical business layer. The user-defined database will be created according to the user's needs. When creating the database, the user only needs to input the necessary information such as database type, name, and account password into the configuration file of the database. The user-defined database will also automatically layer different crawling tasks for the easy query.

2.2 Scrapy Framework

After the Internet has entered the era of big data, the amount of information data has expanded rapidly, which greatly increases the difficulty of information retrieval. The huge amount of data not only increases the difficulty of information retrieval but also brings more high-value hidden rules. In the face of the increasing amount of network information, the emergence of search engines makes information retrieval more convenient. The working principle of a search engine is to crawl information data in the Internet using crawler software and classify the information according to the needed keywords.

As Python is easy to learn and has a large standard library of modules, crawlers are usually written in Python. Scrapy is a crawler framework completely written by Python language [4], and its operation diagram is shown in Figure 2. The crawler framework consists of a scrapy engine, scheduler, crawler, crawler middleware, downloader, download middleware, project pipeline, and the Internet. The scrapy engine is the core of the whole operating framework, which is used for processing the data flow in the operation process. The scheduler is connected with the engine to store the crawling requests from the engine and provide the stored crawling requests to the engine, *i.e.*, a crawling task list.

The downloader is connected with the Internet, downloads the web page information on the Internet according to the task target order given by the scheduler, and transmits it to the crawler. The crawler module that contains the Internet crawling logic and parsing rules for downloaded content is responsible for generating the parsing results and subsequent search requests. The project pipeline will receive the result data from the crawler, clean and verify the data to reduce the interference of bad information, and store them in the database. The crawler middleware and download middleware are the intermediate processing modules between the crawler and engine and between the downloader and engine, respectively [8].

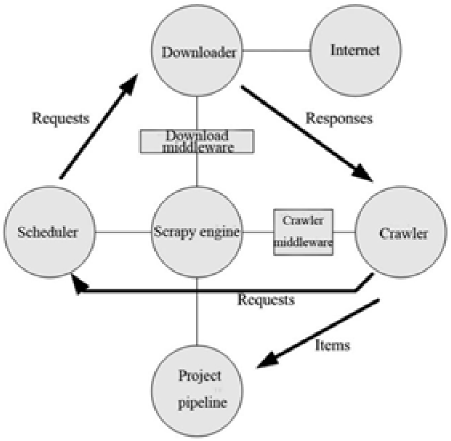

Figure 2: Scrapy operation framework

The basic operation flow of scrapy is as follows:

1) The crawler generates the request according to the crawling logic and then transmits it to the scheduler to get the crawler request list without being processed by the engine.

2) The engine obtains the crawler request from the scheduler and starts to crawl the information, and the request is passed to the downloader through the download middleware.

3) The downloader downloads the web page information from the Internet according to the address given by the crawler request and transmits it to the engine through the download middleware.

4) The engine feeds back the web page information to the crawler through the crawler middleware, and the crawler parses the information according to the set parsing rules.

5) The data obtained after parsing is transferred to the project pipeline by the engine to clean and verify the data.

The above steps cycle until there are no more crawler requests in the scheduler.

2.3 Clustering Algorithm for Data Arrangement

Crawler software usually crawls the network information to obtain the needed information and improve the retrieval efficiency. Facing the huge amount of network information, in order to facilitate storage and subsequent retrieval, crawler software will classify the crawled information.

The traditional crawler software usually classifies the string of the crawled information by word segmentation and divides the information containing the same keywords into one category. This method is relatively simple, and information containing the same keywords is generally relevant on the surface. However, in the era of big data, it is more important to deeply mine the hidden rules in the network information. Classifying by relying on keywords only is likely to divide the information with the same or similar content but different keywords into different topics, which will ultimately affect the effectiveness and comprehensiveness of the retrieval results. Clustering algorithm [14] is an algorithm that divides based on the difference between data, which not only depends on the difference of keywords but also depends on the deep connection of information.

In the crawler software, the crawler of the scrapy framework crawls the information data according to the URL. Before storing the data in the database, the data are classified using the clustering algorithm to facilitate the subsequent accurate storage and retrieval. The data classification flow of the clustering algorithm is shown in Figure 3.

1) Firstly, the crawling data are preprocessed to remove the information noise and segment words. The removal of information noise includes deleting the meaningless characters and the text that cannot express the meaning because of few words. The segmentation of words is to obtain individual words from the text to form the vector features of the information data.

2) Then, Gibbs sampling is performed on the preprocessed information data [7] to reduce the vector feature dimension of the information data and the amount of calculation. The sampling formula is:

$$\begin{aligned} & P(Z_j = k|\overrightarrow{Z_{\neg i}}, \overrightarrow{w}, \alpha\beta) \propto \theta_{mk} \cdot \varphi_{kt} \\ = & \frac{n^k_{m,\neg i} + \alpha_k}{\sum_{k=1}^{K}(n^t_{m,\neg i} + \alpha_K)} \cdot \frac{n^t_{m,\neg i} + \beta_k}{\sum_{t=1}^{K}(n^t_{k,\neg i} + \beta_K)} \end{aligned}$$

where Z_j is the j^{th} word in all information data sets, i is a two-dimensional subscript, which is composed of m and n, representing n words in m information data, K represents for the number of hidden themes, $\overrightarrow{w}$ stands for a word, and α and β are the prior superparameter of information data theme and the prior superparameter of words under the theme respectively, both of which obey the Dirichlet distribution [11]. After repeated sampling to convergence, the theme distribution of every information data (θ) can be obtained, which is taken as the vector feature of information data.

3) According to the input K value, K cluster centers are randomly generated, and then the distance between information data and different cluster centers is calculated according to the distance calculation formula:

$$d(x, Z_j) = \sqrt{\sum_{i=1}^{n}(x_i - Z_{ji})^2},$$

where $d(x, Z_j)$ stands for the distance between data x and cluster center Z_j, x_i stands for the i^{th} dimensional data of x, and Z_{ji} stands for the i^{th} dimensional data of Z_j. Then, according to the distance, the information data are allocated to different cluster centers.

4) After clustering, the cluster center of every kind of data set is recalculated, and then Step 3 is repeated to reclassify.

5) Steps 3 and 4 are repeated until the clustering criterion function reaches the predetermined standard, and its equation expression [12] is:

$$\begin{aligned} J(I) &= \sum_{j=1}^{k}\sum_{i=1}^{n_j}||x_i^j - Z_j(I)||^2 \\ |J(I) - J(I-1)| &< \xi, \end{aligned}$$

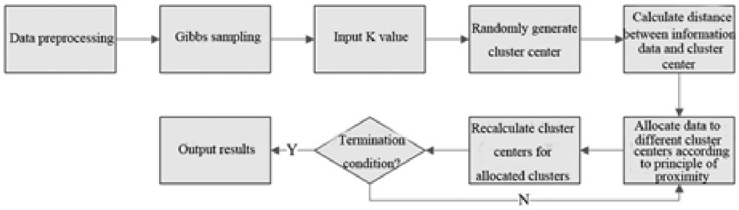
Data preprocessing
Gibbs sampling
Input K value
Randomly generate cluster center
Calculate distance between information data and cluster center
Allocate data to different cluster centers according to principle of proximity
Recalculate cluster centers for allocated clusters
Termination condition?
Y
N
Output results

Figure 3: The calculation flow of the clustering algorithm

where $J(I)$ stands for the square error of the I^{th} clustering results, x_i^j stands for the i^{th} data in j category, $Z_j(I)$ stands for the cluster center of the j category at the I^{th} clustering, and ξ is a threshold for determining whether the iteration terminates or not.

3 Simulation Analysis

3.1 Experimental Environment

In this study, the crawler software and its clustering algorithm were programmed using the python software. The simulation experiment was carried out in a laboratory server using MATLAB [3]. The configuration of the server was Windows 7 operating system, 16G memory, and Core i7 processor.

3.2 Experimental Data

In order to verify the effectiveness of the crawler software in crawling information data after adding the clustering algorithm, this study compared it with the crawler software without the clustering algorithm. In order to ensure the accuracy of the comparison results, it is necessary to know the actual information data of the subject crawled by the crawler software. However, in the real Internet, new information will be generated constantly, and it is nearly impossible to collect complete actual information data. Therefore, the experiment in this study built a LAN in the laboratory.

In the LAN, a server provided website services, and the rest of PC used the crawler software to crawl the website information. The web page data in the server providing website service came from the collectible information of Weibo. The Weibo data between October 1 and October 31 were collected through the application programming interface (API) of Weibo. There were a total of 3000 text messages, including five themes: 5G (520 messages), mobile payment (390 messages), anti-corruption work (650 messages), animation (750 messages), and environmental protection (690 messages). Through the manual review, the theme of Weibo data came from the central idea reflected by each message. The messages might not include the same keywords as the theme name. There was also a connection between different themes, and there was a small amount of Weibo information containing keywords of other theme names.

The above situation of keyword mixing in different themes could be used as the interference of crawler software on crawling information classification storage. In the experiment, crawler software with clustering algorithm and software without clustering algorithm were used to crawl the micro blog information in the laboratory LAN, and then the information after crawling classification was analyzed.

3.3 Experimental Results

Two kinds of crawler software crawled the Weibo information in the LAN of the laboratory and then classified and stored the crawled information. The final results are shown in Figure 4, showing the actual proportion of Weibo information classification. It was seen from Figure 4 that “5G” messages accounted for 17.33%, “mobile payment” messages accounted for 13.00%, “anti-corruption work” messages accounted for 21.67%, “animation” messages accounted for 25.00%, “environmental protection” messages accounted for 23.00%, and the rest of messages accounted for 0% among the actual Weibo information; in the classification of the crawler software without the clustering algorithm, “5G” messages accounted for 16.00%, “mobile payment” messages accounted for 12.00%, “anti-corruption work” messages accounted for 21.07%, “animation” messages accounted for 24.37%, “environmental protection” messages accounted for 22.5%, and the rest of messages accounted for 4.07%.

In the classification of the crawler software that was added with the clustering algorithm, “5G” messages accounted for 17.30%, “mobile payment” messages accounted for 12.93%, “anti-corruption work” messages accounted for 21.57%, “animation” messages accounted for 24.93%, “environmental protection” messages accounted for 22.97%, and the rest of messages accounted for 0.30%. It was seen from Figure 4 that both crawler software could effectively crawl effective information from Weibo and classify information. There were only five classification themes in the actual Weibo information. Although the two kinds of crawler software also classified five themes, there were some other information classification, especially the classification by the crawler software without the clustering algorithm. Only a small part of the infor-

Table 1: Accuracy and false alarm rate of two kinds of crawler software for classification of Weibo crawling information

	Crawler Software **without** the Clustering Algorithm		Crawler Software **with** the Clustering Algorithm	
	Accuracy/%	False alarm rate/%	Accuracy/%	False alarm rate/%
5G	88.7	5.2	98.2	1.7
Mobile payment	89.6	5.6	98.3	1.6
Anti-corruption work	89.4	5.4	97.5	2.3
Animation	88.8	5.2	98.6	1.4
Environmental protection	89.2	5.7	99.1	0.7
Comprehensive evaluation	89.1	5.4	98.3	1.5

mation was classified as other categories by the crawler software added with the clustering algorithm.

On the whole, the classification of the crawling information by the crawler software that was added with the clustering algorithm was very close to the actual Weibo information classification; however, the crawler software without the clustering algorithm classified more information into other categories, and the classification of crawling information was more deviated from the actual information classification.

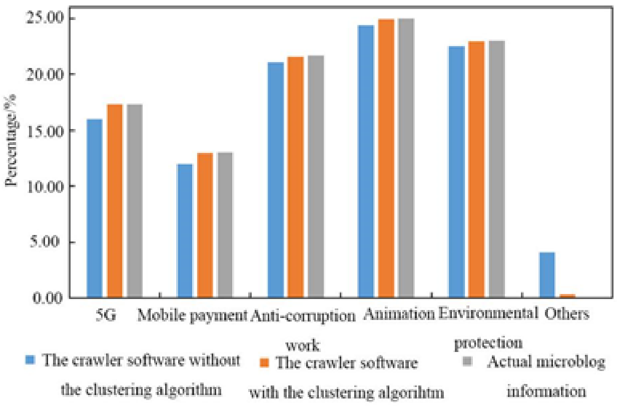

Figure 4: Classification proportion of Weibo crawling information by the two crawlers and the actual proportion

Although the classification proportion of Weibo crawling information shown above also reflected the effect of the two crawler software on information crawling and classification, the classification proportion only evaluated the classification information from the whole but could not reflect whether the different information was classified accurately. Table 1 shows the classification accuracy and false alarm rate of the two crawlers. By comparison, it was found that no matter what kind of classification information, the crawler software with the clustering algorithm had higher accuracy and lower false alarm rate.

When the two kinds of crawler software classified the Weibo information, there were other types that were not identified in the actual Weibo information classification; moreover, the crawler software with the clustering algorithm had higher classification accuracy and lower false alarm rate. The reason was that keyword mixing between different themes interfered with the classification of the two software, especially the crawler software that was not added with the clustering algorithm. The crawler software with the clustering algorithm classified the text information based on the vector features of the information; therefore, the influence caused by fixed keyword mixing was relatively small. As the keyword was also a part of the vector feature, the keyword mixing still impacted the features, leading to classification errors.

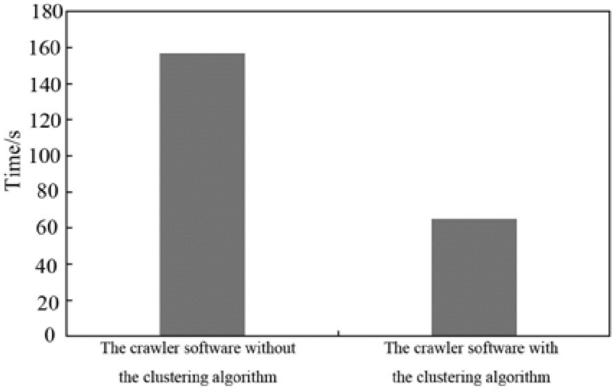

Figure 5: The time required for two kinds of crawler software to crawl and classify Weibo information

As shown in Figure 5, it took 157 s for the crawler software without clustering algorithm to crawl and classify Weibo information and 65 s for the crawler software with the clustering algorithm. The comparison in Figure 5 shows that the scrapy framework based crawler software that was added with the clustering algorithm could classify the crawling information faster in the face of big data Weibo information could classify the crawled information faster when faced with a large amount of Weibo information. Combined with the above results, it was seen that the crawler software that adopted the scrapy framework could effectively crawl the Weibo information and could classify the crawled information faster and more accurately after using the clustering algorithm.

4 Conclusion

This paper briefly introduced the basic structure of crawler software, the scrapy framework, and the clustering algorithm that was used for improving the performance of information classification. Then, the crawler software and clustering algorithm were programmed by the python software, and an experiment was carried out on Weibo data between October 1 and October 31 using the MATLAB software in LAN. The crawler software that adopted the scrapy framework but did not add the clustering algorithm was used as the control. The final experimental results are as follows:

1) In the statistics of the classification proportion of Weibo information, the two crawlers could effectively crawl the effective information from the Weibo and classify the information, but neither of them could make the same proportion as the actual classification; the category except the five themes appeared in the classification of both software, but the classification proportion obtained by the crawler that was added with the clustering algorithm was closer to the actual proportion;

2) The scrapy framework based crawler software had higher accuracy and lower false alarm rate in crawling and classifying Weibo information after being added with the clustering algorithm;

3) The scrapy framework based crawler software spent less time crawling and classifying information after being added with the clustering algorithm.

References

[1] J. Bao, P. Liu, H. Yu, C. Xu, "Incorporating twitter-based human activity information in spatial analysis of crashes in urban areas," *Accident Analysis & Prevention*, vol. 106, pp. 358-369, 2017.

[2] T. Fang, T. Han, C. Zhang, Y. J. Yao, "Research and construction of the online pesticide information center and discovery platform based on web crawler," *Procedia Computer Science*, vol. 166, pp. 9-14, 2020.

[3] A. C. Gabardo, R. Berretta, P. Moscato, "M-Link: a link clustering memetic algorithm for overlapping community detection," *Memetic Computing*, vol. 12, no. 2, pp. 87-99, 2020.

[4] U. R. Hodeghatta, S. Sahney, "Understanding twitter as an e-WOM," *Journal of Systems & Information Technology*, vol. 18, no. 1, pp. 89-115, 2016.

[5] C. Y. Huang, H. Chang, "GeoWeb crawler: an extensible and scalable web crawling framework for discovering geospatial web resources," *International Journal of Geo-Information*, vol. 5, no. 8, pp. 136, 2016.

[6] M. A. Kausar, V. S. Dhaka, S. K. Singh, "Design of web crawler for the client - server technology," *Indian Journal of Science and Technology*, vol. 8, no. 36, 2015.

[7] F. Li, L. L. Dai, Z. Y. Jiang, S. Z. Li, "Single-Pass Clustering Algorithm Based on Storm," *Journal of Physics Conference*, vol. 806, pp. 012017, 2017.

[8] S. H. Peng, P. Y. Liu, J. Han, "A python security analysis framework in integrity verification and vulnerability detection," *Wuhan University Journal of Natural Sciences*, vol. 24, no. 2, pp. 141-148, 2019.

[9] S. Raj, R. Krishna, A. Nayak, "Distributed component-based crawler for AJAX applications," in *Second International Conference on Advances in Electronics, Computers and Communications*, pp. 1-6, 2018.

[10] A. Seyfi, "Analysis and evaluation of the link and content based focused treasure-crawler," *Computer Standards & Interfaces*, vol. 44, pp. 54-62, 2016.

[11] F. F. Wang, B. H. Zhang, S. C. Chai, "Deep auto-encoded clustering algorithm for community detection in complex networks," *Chinese Journal of Electronics*, vol. 28, no. 3, pp. 49-56, 2019.

[12] B. Yuan, T. Jiang, H. Z. Yu, "Emotional classification algorithm of micro-blog text based on the combination of emotional characteristics," *Advanced Materials Research*, vol. 1077, pp. 246-251, 2015.

[13] F. Zhao, J. Zhou, C. Nie, H. Huang, H. Jin, "SmartCrawler: A two-stage crawler for efficiently harvesting deep-web interfaces," *IEEE Transactions on Services Computing*, vol. 9, no. 4, pp. 608-620, 2016.

[14] F. Zhao, Y. Zhu, H. Jin, L. T. Yang, "A personalized hashtag recommendation approach using LDA-based topic model in microblog environment," *Future Generation Computer Systems*, vol. 65, pp. 196-206, 2016.

Biography

Dashan Wang, born in 1980, received the master's degree of engineer from Hubei Agricultural College in 2003. He is a lecturer in Hainan College of Vocation and Technique. He is interested in computer network.

Qingbin Zhang, born in 1989, received the bachelor's degree from Dongbei University of Finance and Economics in 2020. He is a network engineer in Hainan College of Vocation and Technique. He is interested in computer network.

Shaoxian Hong, born in 1977, received the master's degree of education from Henan University in 2001. She is an associate professor in Henan University. She is interested in English teaching theory and research.