Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(5):2057-2063

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

Research of a professional search engine system based on Lucene and Heritrix

Ying Hong[*] and Chao Lv

Computer Information Center, Beijing Institute of Fashion Technology, Beijing, China

ABSTRACT

In order to solve user's problem of searching professional information quickly and correctly, a professional search engine is designed and realized. In the first place, the web pages are collected by the means of extended Heritrix web crawler. The data extracted from web pages based on jsoup is saved to the local. In the second place, Chinese words segmentation, inverted index, index retrieval and improved web page ranking algorithm technology are taken to handle the collected data. At last, a professional search engine is designed and realized. The experimental results show that this professional search engine enhances accuracy and efficiency of web page information retrieval in great degree.

Keywords: professional search engine; Lucene; Heritrix; information retrieval; PageRank; web page ranking; Chinese word segmentation; jsoup

INTRODUCTION

How to obtain the required information accurately in the massive level of internet resources is an important research question in computer science. To some extent, the appearance of general search engines has solved the problem of information retrieval from internet. Although general search engine is more powerful, its retrieval performance is not good in professional field. The ambiguity of Chinese vocabulary cause the results searching by general search engines now often cover the relevant network information in all fields including key words. For the user who wants to search the key information in relevant professional field accurately, the workload of filtering and searching the results is huge certainly.

In this paper, we have analyzed the overall architecture of professional search engine system and designed the function of each module based on Heritrix and Lucene. At last, we have achieved a professional search engine system.

INTRODUCTION

General search engine and professional search engine

The general search engine searches for information on the web with a certain degree of strategy. It presents the information to the user after building indexes, removing duplicates and sorting. Its work includes two parts: information crawling and information retrieval.

The work of information crawling is completed by webpage capture program (spider, also known as network spider) and indexer. Network spider traversals and crawls the webpage information along the hyperlinks in the webpage on the basis of scanning strategy. The indexer processes the webpage information including word segmentation, removing duplicates and sorting. Then it will establish an index database. Information retrieval is work by the searcher and the user interface. When the user input query keywords, the searcher will delete meaningless characters and then find the relevant information from the index database to present to the user on the basis of sorting strategy. The user interface is an interactive interface between search engine and users. Its function is for users to input keywords and view the results. Professional search engine and general search engine have the same working principle basically but two different: (1)When the spider crawled the webpage information, firstly it will filter the information with the help of professional lexicon, and then establish an index database;(2)It needs to consider professional relevance of the search results in ranking algorithms[1].

Lucene

Lucene is a free, open-source information retrieval library written in Java and supported by the Apache Software Foundation. It is suitable for any application which requires full-text indexing and search, and is a popular choice for consumer and business web applications, single-site searching, and enterprise search. It can index text from a range of content types including HTML, PDF, Microsoft Word and Excel, XML [2]. All the source code of Lucene is divided into 7 packages and each package will complete a specific function.

The function of Lucene is more powerful, but the basic work is two main parts: (1) Use algorithm for the Chinese word segmentation on webpage information and then create index. (2) Match the information in the index tree according to the query keywords inputted by the user and then ranks the results properly.

Heritrix

Heritrix is the internet archive's open-source, extensible, web-scale, archival-quality web crawler project [3]. Heritrix has good scalability, so the developer can expand its components to realize their own crawl tactics. When Heritrix was starting, it selects a URI from URI queue and downloads files from remote server. Next, Heritrix will analyze page content and extract new URI to join URI queue for downloading. The architecture of Heritrix is shown in Fig. 1:

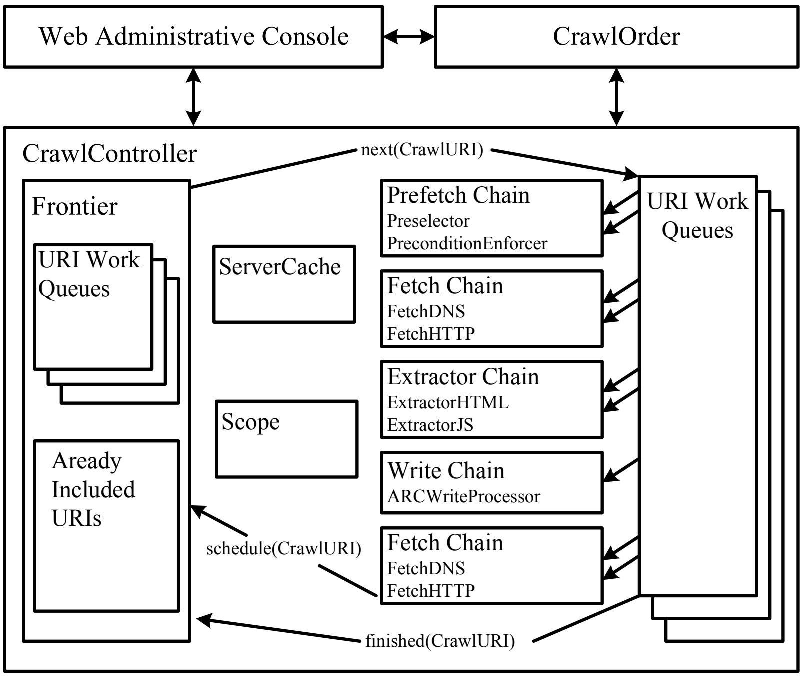

Fig. 1.The architecture of Heritrix

There are some components included in Heritrix:
(1)CrawlOrder. This component is starting point of grasping job. It can be set up with various methods. The simple way is setting by file order.xml.
(2)CrawlController. It is the core component in grasping job. This component controls the beginning and the end of grasping job. It gets URI from component Frontier then transfers it to ToeThread component.
(3)Frontier. The function of this component is providing URI for ToeThread component. It decides which URI will be put in process chain by using specific algorithm.
(4)ToeThread and ToePool. ToeThread is a grasping thread. ToePool is thread pool which manage all grasping threads. Heritrix can grasp web page effectively by using multithreading.
(5)Processor. The function of this component mainly is downloading web pages and extracting URI.

Jsoup
Jsoup is an html parser developed with java. It is a Java library for working with real-world HTML. It provides a very convenient API for extracting and manipulating data using the best of DOM, CSS and jquery-like methods. jsoup implements the WHATWG HTML5 specification, and parses HTML to the same DOM as modern browsers do[4]. Jsoup can scrape and parse HTML from a URL, file, or string. It find and extract data using DOM traversal or CSS selectors.

The architecture of professional search engine system
Considering the characteristic of searching for professional field and the operational efficiency of the system, we can realize the professional search engine system based on Lucene and Heritrix. The architecture of professional search engine system based on components is shown in Fig. 2:

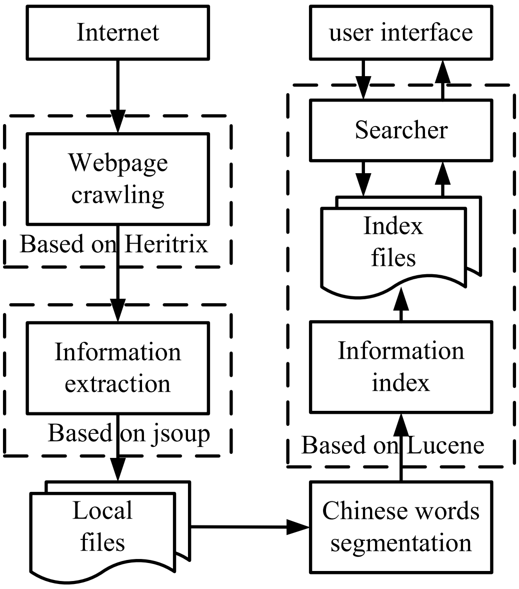

Fig. 2.The architecture of professional search engine system

The function of webpage crawling module is realized by means of expanding Heritrix. Crawling strategy adopted first best algorithm. The gathered data are stored in local files. The function of information extraction module is realized based on jsoup. Information index module is operated based on Lucene. It creates index for documents adopting inverted index technology after Chinese word segmentation. The index files will be stored on local [5]. Searcher module searches information in index files according to key words input by users and returns the results to users after sorting.

Strategy of focused crawling
Focused web crawler filters links which has nothing to theme according to specific web analysis algorithm. It reserves the links related the topic and push them into the URL queue. Then, focused web crawler will select a URL for crawling on the basis of search strategy. This process is repeated until the end. Focused web crawler needs to solve four problems: (1) How to describe the professional topic? (2) How to decide the sequence of URLs which will be selected for crawling? (3) How to judge the correlation between a web page and professional topic? (4) How to improve the coverage of focused web crawler?

Researchers have proposed some strategies of focused crawling and related algorithm. In this paper, we describe the professional topic and adopt best first search algorithm for focused crawling.

Professional topic description
In this paper, we describe professional topic with characteristic words which are selected by consulting the experts in this professional field. Each characteristic words is endowed with a weight used to represent its professional distinction and important degree. The sum of these weights equal 1. By this method, we have a reference document vector which represents the professional topic. Each component of the vector is the weight of every characteristic words and the dimension of vector is the number of characteristic words.

Best first search algorithm

The priority of a URL waiting for crawling is calculated with the correlation degree of web page *p* crawled and the professional topic. The higher the correlation degree is, the higher the priority of a URL which is directed by *p* is. It decides the crawling order of URLS in queues by this method. The calculation formula of correlation degree between web page *p* and professional topic is shown in equations (1):

$$Sim(q,p)=\frac{\sum_{k\in q\cap p} f_{kq}\times f_{kp}}{\sqrt{\left(\sum_{k\in p} f_{kp}^{2}\right)\left(\sum_{k\in q} f_{kq}^{2}\right)}} \tag{1}$$

Among them, *q* expresses the professional topic and *p* is the web page crawled. f_{kq} expresses the occurrence frequency of words *k* appearing in *q*. f_{kp} expresses the occurrence frequency of words *k* appearing in *p*.

Web information extraction

A web page includes many contents such as navigation information, content, copyright information usually. If we store and index the whole web page, the accuracy of retrieval results will be decreased. We must extract the useful content for correctly judging the web page.

In this paper, we adopt jsoup component to extract information and URLs from a web page. The file jsoup-1.7.3.jar can be downloaded on its company website. We need to import it into eclipse. Jsoup is an open source toolkit developed with java. It not only can obtain HTML code of web page according to URL, but also provides many tools to extract useful information from HTML code. We extract title, content and URLs and at last save these to the folder by txt file.

Chinese words segmentation

The Chinese word segmentation machines of lucene named CJKAnalyser is difficult to achieve ideal effects. We must redesign the Chinese word segmentation module of system. At present, there are three mature word segmentation algorithms: based on string matching, based on language understanding and based on statistics. Each of them has its own advantages and disadvantages.

In this paper, we adopted the combination of forward maximum matching algorithm and backward maximum matching algorithm for Chinese word segmentation. This algorithm matches the document positively and backwards and then it compares the results. It can eliminate the ambiguity effectively [6]. Forward maximum matching algorithm is a word segmentation algorithm based on the dictionary. It scans a string from the front to the back. For each word scanning, it finds the longest matching in the dictionary. Backward maximum matching algorithm is similar to forward maximum matching algorithm but it scans a string from the back to the front. The webpage document is segmented to words by two word segmentation algorithms respectively. Then the segmentation results are merged after removing duplicate.
Authoritative dictionary has a direct impact on the effect of segmentation for the maximum matching segmentation algorithm based on dictionary. In this paper, the dictionary we used includes two parts: universal dictionary and professional dictionary.

Design of searcher

The function of searcher module includes key words pretreatment, synonym expansion, query optimization and lucene index retrieval [7]. First, pretreatment module will filter and split the query submitted by user. It removes meaningless character words. Then, searcher module will expand the synonyms of key words. At last, it realizes its search function basis for API which provided by Lucene.

Default web page ranking algorithm of Lucene. Sorting the results of search is very important. Lucene scores results according to its own scoring mechanism [8]. For a given query *q*, the score of document *d* is calculated as shown in equations(3):

$$Score(d)=\sum_{t\in d} tf\left(t\in d\right)*idf\ \_t*boost\left(t.field\in d\right)*lengthNorm\left(t.field\in d\right) \tag{3}$$

The meaning of different parameters as follow:
Score(*d*): the score of document *d*.

tf(*t*∈*d*): frequency of query term *t* in query *q* appearing in document *d*. In Lucene, the value is square root of the real frequency.

idf_t: the number of documents containing query term *t*. Usually *idf_t* is calculated by equations(4):

$$idf\_t = \log_2\left(\frac{numDocs}{docFreq\_t}+1\right)+1 \tag{4}$$

Among them, *numDocs* is the total number of indexed documents. *docFreq_t* is the total number of documents including query term *t*.

boost(*t.field*∈*d*): It is a motivating factor for query term *t* in a field which default value is 1. It increased the importance of this field. At the same time also increased the importance of the document.

lengthNorm(*t.field*∈*d*): It is a factor which reflects the size of document. The longer the document is, the lower the value of the factor and vice versa.

We can find that the location of query in the document is not important in Lucene webpage ranking algorithm. The more the query appears in a document, the higher the score of this document is. The disadvantage of the algorithm is unable to reflect the importance of query position and the important values of web pages.

Improvement of PageRank algorithm

The PageRank algorithm represents important values of web pages by number. Its basic idea to determine the importance of web page is based on "a web page must be a high-quality web page if it is linked from another high-quality web page [9]". Namely, a web page with higher PageRank value will be on the top of the retrieval results. PageRank value of the web page *u* is calculated by equations (5):

$$PR(u) = (1-d) + d\sum_{i=1}^{n}\frac{PR((T_i)}{C(T_i)} \tag{5}$$

There in, *PR*(*u*) is PageRank value of the web page *u*. *PR*(T_i) is PageRank value of T_i which links to *u*. *C*(T_i) is the number of links out of T_i. *d* is damp coefficient, 0< *d* <1, and it is generally taken as 0.85[10].

From Eq. 5, we can find that the PageRank algorithm does not distinguish whether a web page and other web page linking in are in the same site. The PageRank algorithm is improved in this paper as shown in equations (6):

$$PR(u) = (1-d) + d\left(\mu\sum_{T_i\in V_i}\frac{PR((T_i)}{C(T_i)} + (1-\mu)\sum_{T_j\in V_o}\frac{PR((T_j))}{C(T_j)}\right) \tag{6}$$

There in, V_i is the set of web pages which link in *u* and are in the same site with *u*. V_o is the set of web pages which link in *u* but are not in the same site with *u*. μ is weighting factor,0<μ<1 and it is generally taken as 0. 5.

The initial PageRank values of all web pages are set to 1. The PageRank values of web pages will converge to a fixed value by about 20 times iterative calculation with Eq. 6.

Improvement of Lucene web page ranking algorithm based on link analysis

In this paper, we improved Lucene webpage ranking algorithm by the means of introducing PageRank algorithm. It is shown as equations (7):

$$newScore(d) = \alpha * Score(d) + \beta * PageRank(d) \tag{7}$$

There in, α and β are empirical coefficients and the exact values will be determined through many experiments. In our experiments, the empirical coefficients are set to 1.

Analysis of experiment results

We have verified the improved Lucene webpage ranking algorithm based on link analysis algorithm through designing twice comparative experiments. First, we adopt default web page ranking algorithm of Lucene and use 20 random key words to search information. Then, we adopt improved Lucene web page ranking algorithm based on link analysis and do the same search. At last, we calculate the numbers of eligible web pages in the first 100 records retrieved. The results are shown in Fig. 3. Experimental results show that precision ratio is higher when we adopt improved web page ranking algorithm.

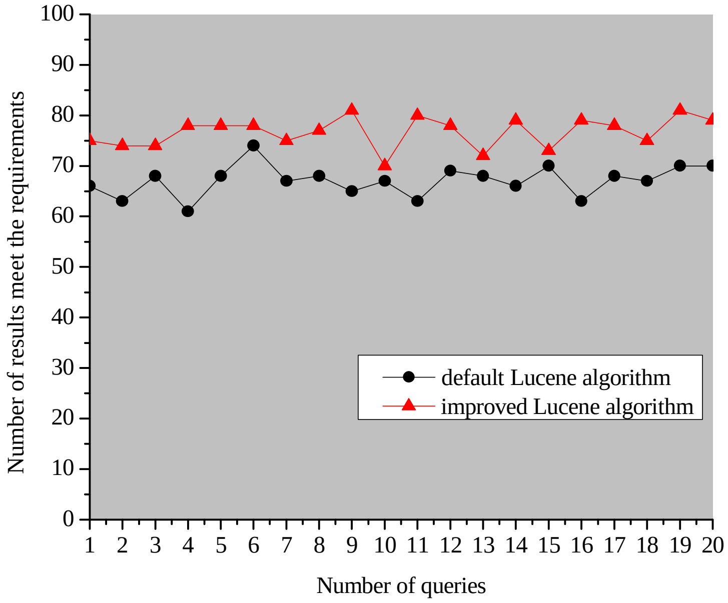

Fig. 3.The experimental results of improved algorithm contrasting with default algorithm

For contrasting this system with general search engines, we input 20 random key words to search in baidu, Google and this system respectively. Then we calculate the numbers of eligible web pages in the first 100 records retrieved. The results are shown in Fig. 4. Experimental results show that this system has a higher precision ratio in professional field retrieval comparing with general search engine.

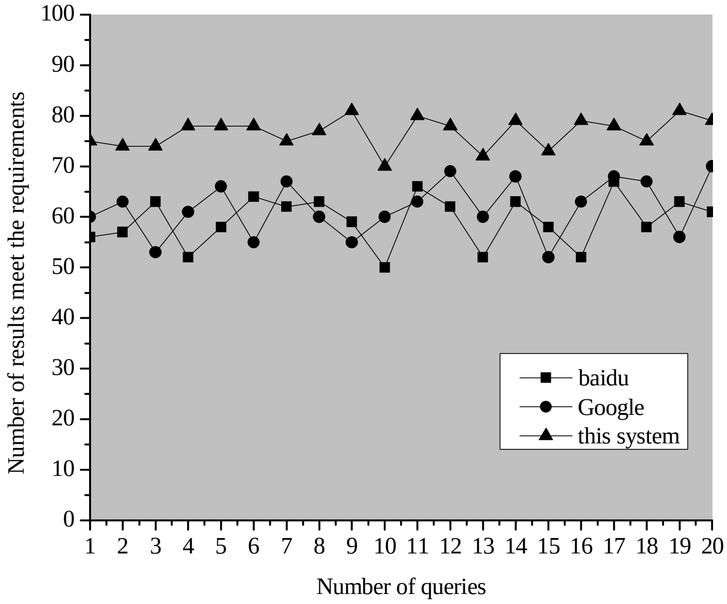

Fig. 4.The experimental results of this system contrasting with general search engines

CONCLUSION

Professional search engine is becoming a research hotspot now in view of its precise search results. In this paper, we have analyzed the overall architecture of a professional search engine system and designed the function of each module after researching deeply in Heritrix and Lucene. We have extended Heritrix for collecting the web page accurately and efficiently. Not only that, we have analyzed default web page ranking algorithm of Lucene and improved it based on link analysis. At last, we have designed a professional search engine system and have realized searching rapidly. Experimental results show that this system has a higher precision in professional field retrieval.

Acknowledgements

This work was financially supported by the Foundation for Beijing teacher team construction-youth with outstanding ability project (No.YETP1414) and the scientific research program of Beijing institute of fashion technology (No.2012A-17).

REFERENCES

[1] Zhao Ke, Lu Peng, Li Yongqiang: Design and Implementation of Search Engine Based on Lucene. Computer Engineering, **2011**, 37(16): 39-41(In Chinese)

[2] Information on http://lucene.apache.org/

[3] Information on http://nutch.apache.org/

[4] Liu Qinchuang: *Journal of Hanshan Normal University*, **2008**, 29(3): 22-25(In Chinese)

[5] Wang Shuo, You Feng, Shan Lan, Zhao Hengyong: *Computer Engineering and Applications*, **2008**, 44(19): 142-145(In Chinese)

[6] Liu Weidong, Lu Ling:. *Computer and Modernization*, **2011**, 191(7): 96-98(In Chinese)

[7] Zhang Xian, Zhou Ya: *Computer Systems and Applications*, **2009**, 18(2): 155-158(In Chinese)

[8] Gu Wenli, Chen Wei, Chen Jiao, Lu Xiaoye: *Computer Systems and Applications*, **2012**, 21(2): 214-217(In Chinese)

[9] Wang Dong, Lei Jingsheng: An Improved Ranking Algorithm Based on PageRank. *Microelectronics and Computer*, **2009**, 26(4): 210-213(In Chinese)

[10] Liang Zhengyou, Pan Tao: Parallel Realization of PageRank Algorithm on Nutch. *Computer Engineering and Design*, **2010**, 31(20): 4354-4356(In Chinese)