International Journal of Future Generation Communication and Networking
Vol. 7, No. 6 (2014), pp. 13-20
http://dx.doi.org/10.14257/ijfgcn.2014.7.6.02

A Focused Crawler Based on Correlation Analysis

Qiuli Qin and Xin Peng

Logistic Technology and Management Lab, School of Economics and Management, Beijing Jiaotong University, Beijing 100044

Abstract

With the rapid development of network and information technology, there is a wealth of huge amounts of data on the internet. But it's a major problem faced by the majority of researchers how to effectively filter out a particular subject or field of information from these data. In this paper, we try to builder a focused crawler based on vector space model and TF-IDF text correlation analysis. We take the seed URL as a collection entrance and fetch web pages from internet. Then analysis page information though technological like web content extraction, page link analysis technology and get the main content of one page. By the correlation analysis method based on VSM and TF-IDF text, we calculation the correlation between pages and the topics what have been defined, so we can achieve the purpose of the focus areas of the web.

Keywords: *Focused Crawler; web crawler; VSM; TF-IDF*

1. Introduction

With the rapid development of network and information technology, the internet has become the carrier of the mass of information. How to find and make use of the useful information to oneself has become a big challenge to everyone. Traditional search engines, such as Baidu and Google, can partly meet the demand of people to retrieval, but there is still a certain lack of:

(1) Traditional search engine index the entire internet web pages, so the retrieval results contain thousands of all things. But the different background, different field users' retrieval requirements naturally differ in thousands ways, which leads to they can't provide professional information for professional users;

(2) Traditional search engines are based on keyword search and cannot be implied from the keywords of semantic retrieval result, which lead to users can't get information they want;

(3) Current search engines are unable to meet the personalized requirements of people. At the same time, due to the lack of comprehensive treatment of the retrieved results, it's somewhat laborious to find information they want from a large number of results.

In order to resolve these problems, focused crawler which crawls specific theme information arises at the historic moment. Focused crawler is a web crawler which only fetches scheduled topics or areas related web pages. When fetching a web page, it determine whether the page is related to the topic pre-determined. If it's relevant, it will be saved; otherwise discarded. The domain knowledge base can be as simple as a theme or a set of keywords, can also be a collection of field information. Compared with the traditional web crawler which pursues the big number and the completeness of features, focused crawler pursues the precise. It aims at a certain class or some kind of theme to quickly and accurately grasp the target information, and increase the recall rate as well as guaranteeing the accuracy.

ISSN: 2233-7857 IJFGCN

This paper researches on the design of the focused crawler based on correlation analysis. The correlation analysis method is based on vector space model of TF - IDF text relevancy calculation method. It is used to extract the main contents from the collected targeting pages, do contrast analysis with the domain knowledge base, and calculate the page relevance. If the relevance rate is high enough, then it will be indentified as related field. The first part of this paper will make research on the basic flow of the topic focused crawler; the second part presents the implementation of the focused crawler in detail; the third part is the experimental analysis, verifying the accuracy and recall rate of this method through the crawler experiment; the fourth part is conclusion and prospect.

2. The Process of Focused Crawler

Web crawler is a web robot, grabbing public network pages from internet, and it's an important part of the search engine. Traditional web crawler finds other web pages by the links in web pages. It begins from a page of a website (usually the home page) and read the contents of the page, to find other links in a web page, and then via these links for a web page. so the cycle continuously, until finish all web scraping this site so far. If regard the internet as a web site, network spiders can use this principle to crawl down all web pages on the internet.

For researchers in the specific field, if he wants to get focus areas related information from the internet by this way, it's just like looking for a needle in a haystack. Topics focused crawler can solve the problem by fetching the data according to the particular subject. This paper focuses on the process of the topic focused crawler, which is shown in Figure 1.

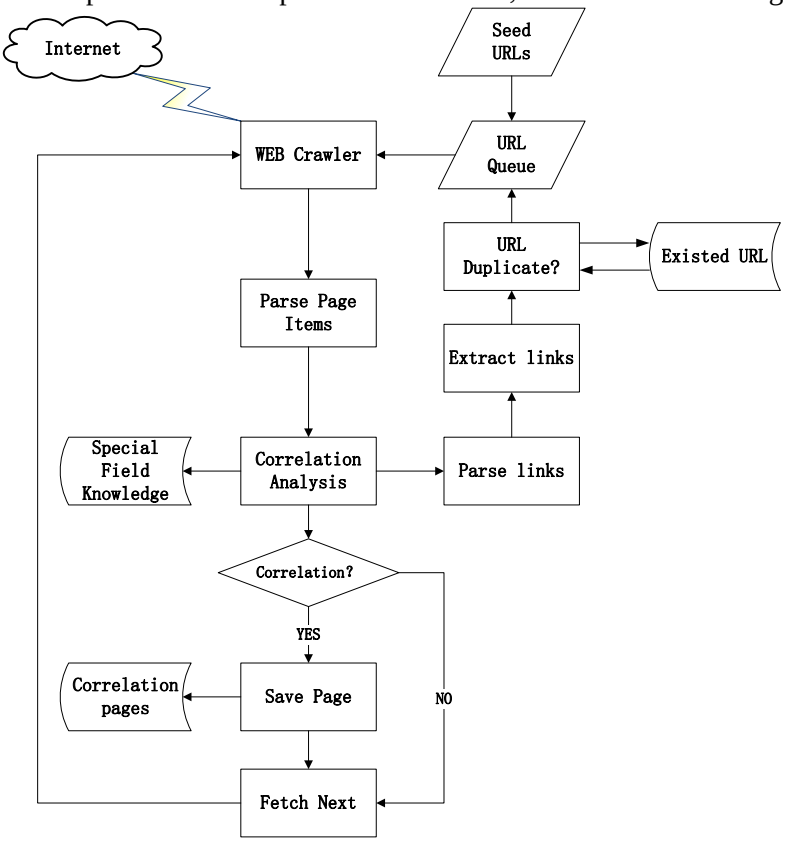

Figure 1. The Process of Focused Crawler

In a focused crawler, there is a URLs queue waiting for fetched. The initial of this queue is the seed URLs. After the crawler starts, it gets a URL from queue, and then downloads the corresponding web page. Do pretreatment to the web page, and parse main elements, such as title, content, pubtime, etc. Based on these elements, analysis whether it's correlated and page links. If a web page is correlated, it will be stored in the database of correlation pages; otherwise it will be discard the page. For pages link analysis results, compare them with existed URL library, and do URL repetitive inspection, and then push URLs which isn't duplicate into the collection queue. This is a whole process dealing with a web page. The crawler loop execution, until finish processing all URLs in the collection queue.

Unlike traditional web crawler, after scraping to web pages, focused crawler does topic relevance determination depending on the web page content. If the web page is not relevant, it will be discarded and will not be saved. This can reduce the irrelevant data in the database, and can improve the effective utilization rate of hardware. At the same time, researchers were able to get the data they need quickly. Meanwhile, the page links extracted from a related page will get a high weight, and will be priority processing. This is taking into account information page on the same subject appears in certain aggregation, if the current page is within the scope of the regulations of acquisition, then the other links in this page is likely to also within the scope of the collection. Such as in the internet, for example, in the website of this news "新浪微博寻找新大陆：2012 成失去的关键一年". Many links in the page are related to internet news. This strategy can guarantee the acquisition as soon as possible, instead of mixing of potentially relevant and irrelevant data, which will influence efficiency of the crawler.

3. Key Technologies

In the implementation of topic focused crawler, there are several key technologies.

3.1. Web Page Pretreatment

Web page pretreatment is purpose to comb clean HTML content and creates page DOM tree. Web page cleaning includes several parts, as following:

(1) Eliminating script code which influence analysis, such as the content of the script node and the action script in normal web page nodes;

(2) Remove style code in the page, such as the content of the style node and style attribute of normal web page nodes;

(3) To eliminate commented code, that is comments in your web page, the general format is "<! -- comment content -- - >";

(4) Rejecting other irrelevant code, such as banner ads, copyright statement and so on.

After cleaning, then build the DOM tree. Full name of DOM is the Document Object Model, which is recommended as the extensible markup language standard programming interface by the W3C. It is a platform and language-neutral API (application program interface), and can access the programs and scripts to dynamically update the content, structure and style of WWW documents (currently, HTML and XML documents is defined by indicating that some). Figure 2 is a DOM tree that is built from a simple web page.

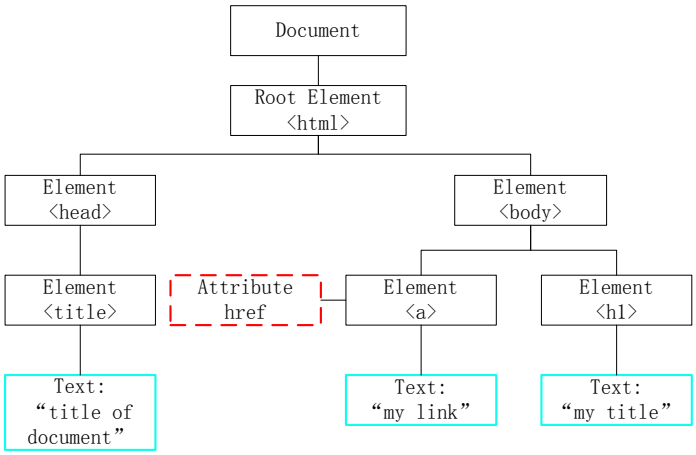

Figure 2. DOM Tree

Through the DOM tree, we can easily find out the link nodes, text nodes and their superior nodes. It's a help for further analysis. This paper uses the DOMParser to parse HTML. Considering there may be non-standard nodes in some pages, such as lack of common ending tag, etc., DOMParser do supplementary treatment for this kind of label, that can guarantee the DOM tree is complete and specification.

3.2. Web Page Analysis

Web page analysis aims to extract the text and links in the web page. The text will be used to analyze the correlation, while the links can be used as the URL seeds in the next grasping phase of the crawler. The following parts describe these in detail.

(1) Extracting the whole text from the web pages

It means that extracting the body text from the web pages, for example the news content in the news web pages and the blog paper contents form the blog post. Let's put Sina as an example, comparing Sina home page with a news page, we can easily find that the whole text exists in the news page (what we should note here is that the text refers to the non-link text which should be separated from the link text, the link node - the text corresponding to the A node is referred as anchor text), while in the home page the whole text does not or seldom exists. From the DOM tree structure, there are many long text paragraphs nodes in News page, such as the P-node, SPAN node and others; While in the Home page, there are many nodes - A link node.

Specific extraction process is as follows: first, traversing up the DOM tree and counting the TextLength of all general text except the anchor text in each node; second, calculating the ratio of each node in the total text of the web page by formula (1), if the ratio is greater than a certain threshold (the threshold is generally set to 0.3), then it will be identified as a possible text node. At the same time, through calculating the number of child nodes contained to a general node by formula (2) to determine the direct parent node of a web page text, and the threshold is set to 0.5.

$$NodeRatio = TextLength / \sum_i TextLer \quad (1)$$

$$UnlinkNodeRatio = UnlinkNodeNumber / \sum_j UnlinkNodeNum \quad (2)$$

Calculation in the first step is to get possible text nodes, while based on the analysis results of the first step, the second step is to confirm whether the node got in the first step is the text node or not by calculating the ratio of the number of child nodes contained by a general text to the total number of the general text node in a web page. Because in a web page, the main

text is divided into several segments, each segment is a paragraph node, and these paragraph nodes have their direct parent nodes. After getting the text nodes in the second step, you can fetch all the child nodes' texts to grasp the body text of the web page. The article may contain several short paragraphs, such as paragraph headings or relatively brief paragraphs, and these nodes may be identified as non-content nodes due to its calculation results are smaller than the specified threshold in the first step, and as a result it will lead to the incompleteness of the text parsing, and that's the reason why you get all the child nodes' texts instead of just getting the node in the first step. However, according to the result of the second step, you can avoid this problem.

(2) Extract the links in the web page

Compared with extracting text, extracting the links in the web page is relatively simple. First, traversing up the DOM tree to get all 'A' node whose 'href' attribute is not null and is not javascript. Then getting the anchor text and 'href' attributes of each node and making unified conversion to the link addresses according to the 'href' attributes and the current URL addresses. During the process of traversal, you can complete eliminating the repetition of page links to guarantee each link is the only link in the list. In the end you can rule out navigation links and advertising links according to the content and length of anchor text nodes.

3.3. The TF - IDF Text Relevancy Analysis based on the VSM

VSM (Vector Space Model) is an algebraic Model applied in the information filtering, information retrieval, index and assessment. Since it was proposed by Salton and others in the 60s, it has been successfully applied to the famous SMART text retrieval system. VSM presents each document as a vector, and obtain the degree of similarity between different documents by calculating the Angle between vectors. The model is put forward based on the following thought: the semantics of articles is expressed by words. If you take each word in the article as a vector, then you can compare the documents and queries to determine their degrees of similarity. Here, in this paper, the field theme articles or keyword is regarded as a kind of query, building query vectors, and determining whether the article is related to this field or not by calculating the cosine of the Angle between an article and the query vector. Figure 3 represents the basic idea of the vector space model applied in this article, including mapping the field of thematic as a basic vector, and also includes four sample document vectors. Through individually calculating the vector spaces between the vector S with vector D1, D2, D3 and D4 to determine whether the document belongs to this field or not.

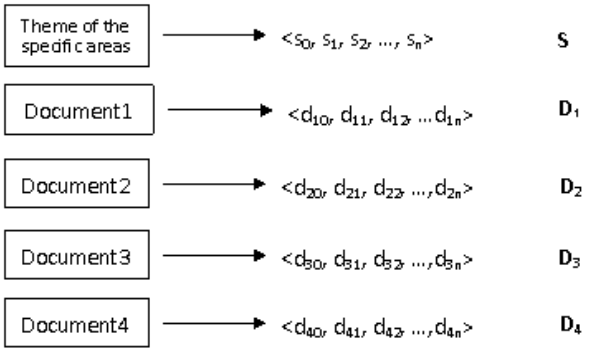

Figure 3. VSM

For every different word in the document collection, VSM records it as a component in the vector. For example, word "a" appears twice in document D1, then the corresponding value of vector component of word "a" is 2. Each word in the document are treated fairly here, and the importance of each word in expressing the theme of document has not been taken into consideration. However, in fact some words are indeed more important than others, so you should put a weight value to each word here. The specific weight value of each word is determined by the frequency of the data set. For a given word, its weight value is calculated by using the IDF (Inverse Document Frequency, Inverse Document Frequency). Define the following variables:

t: the number of different words in document collections

tf_{ij}: the number of times word "t_j" apprears in Document D_i, that's term frequency

df_j: the number of documents which contain the word "t_j"

idf_j: $\lg\frac{d}{df_j}$, here the word "d" represents the number of all documents

In every document, the weight value of each word is determined by the frequency it appears in the whole document collection or in a specific document, and then it can be calculated and assigned. In the process of calculation, the main consideration is the value of these two variables -TF and IDF. Since there are lots of specific calculation formulas, this paper uses a common formula:

$$w_{ij} = \frac{(\lg tf_{ij} + 1.0) \times idf_j}{\sum_{j=1}^{t}[(\lg tf_{ij} + 1.0) \times idf_j]^2} \quad (3)$$

This formula can effectively avoid the consequence of the high frequency matching words overwhelming other matching words. Using lg(tf)+1.0 can narrow the scope of the word frequency, and with that we can accurately calculate the appropriate weights to words in different degree of importance.

As a result, this paper uses these two kinds of methods-TF - IDF and VSM to initialize the field vocabulary database and calculate the initial semantic similarity between words in vocabulary database; it computes the initial semantic similarity between the text based on the initial semantic similarity; According to the initial semantic similarity between words and text, it does not stop alternating iterative calculating between the semantic similarity of each text and the semantic similarity of words until convergence. After that, it constructs the ultimate meaning of similar matrix of all vocabularies according to the convergence results of iterative calculation;

According to described ultimate meaning similarity matrix, it transforms the original text's text word frequency vector into a new text word frequency vector, and finally calculates the centered text relevancy of the described text.

4. Experiment

4.1. Evaluation Index

In the traditional information retrieval, the basic evaluation index of a system including: recall and precision. Recall is ratio of the number of relevant documents in retrieve results and the number of all relevant documents in the document library; Precision is ratio of the number of relevant documents in retrieve results and the total number of documents in retrieves results. For a topic focused crawler, the main evaluation index including: the total rate and the accurate rate. Similar to recall ratio and precision ratio, the total rate refers to the relevant page number and collected all the related web page in internet the ratio of the number; The accurate rate refers to the web page, collected the real relevant page number and the crawler determine the ratio of number of relevant pages. The full rate can determine the

theme of the crawler coverage, and the accurate rate can determine the efficiency and accuracy of the collected for the crawler.

Refer to these indicators, determine the evaluation index of this experiment: recall ratio and accuracy ratio, specific calculation is as follows:

$$\text{recall} = \frac{\text{The number of collection of related web pages}}{\text{Actual number of relevant pages collection scope}} \quad (5)$$

$$\text{accuracy} = \frac{\text{The number of actual related web pages}}{\text{The number of collection of related web pages}} \quad (6)$$

4.2. Experimental Process

This experiment chooses the electronic commerce field as the target domain. Retrieve "e-commerce" in baidu news, and get the first page of article 20 of the news as seed URLs. The depth of the crawl is setted to 3. Finally we collect 1147 related web pages. Finally, through artificial selection, we get 868 web pages which are really relevant. Within the scope of collection, there are 944 relevant web pages.

4.3. Results Analysis

According to the experimental formula of index, can be concluded that the results are as follows:

Table 1. Experiment Results

recall	91.95%
accuracy	75.68%

Based on the experimental results, it can be seen that the recall rate of this method is 91.95%, which means that it can collect most of the targeting web page; while its accuracy rate is 75.68%, which means that about 75% of the crawler determined relevant pages is proved to be correct.

5. Conclusion

This article discusses the topic focused crawler based on correlation analysis technology. It puts the TF - IDF text relevancy analysis method which is based on the vector space model (VSM) into the traditional web crawler frame, which provides a complete set of method for fast and efficiently grasping the realms information. It is helpful to the actual scientific research work.

With the advent of the era of big data, more and more resources are hidden in the internet. How to quickly find the pin we want from the vast sea of the internet has become a common challenge for us, which on the other hand provides a develop space for the theme focused crawler. At the same time, How to make the focused crawler better understand the web page subjects and how to quickly and accurately collect the realms data, all those questions have become the widely concern and research topics for us.

References

[1] Y. Zhao, “Comparative Study of Services of Common and Semantic Search Engine”, Information Science, no. 2, **(2010)**, pp. 255-270.

[2] L. Zhou and L. Lin, "Survey on the Research of Focused Crawling Technique. Computer Applications, no. 9, **(2005)**, pp. 1965-1969.
[3] D. Zhou and Z. Li, "Survey of High-performance Web Crawler", Computer Science, no. 8, **(2009)**, pp. 26-29.
[4] M. Diligenti, F. M. Coetzee, S. Lawrence, C. L. Giles and M. Gori, "Focused Crawling using Context Graphs, 26th International Conference on Very Large Databases, VLDB 2000 Cairo, Egypt, **(2000)**, pp. 527-534.
[5] Q. Jiang, Z. Gong and Y. Xin, "Design and Implementation of BBS Information Extraction System Based on HTML Parser", Techniques of Automation and Applications, no. 1, **(2012)**, pp. 32-37.
[6] M. Zhu and S. Luo, "Research of a Focused Crawler to Specific Topic Based on Heritrix", Computer Technology and Development, no. 2, **(2012)**, pp. 65-68.
[7] Q. Wang, S. Tang, D. Yang and T. Tang, "DOM-Based Automatic Extraction of Topical Information from Web Pages", Journal of Computer Research and Development, **v**ol. 10, **(2004)**, pp. 1786-1792.
[8] S. Liu, L. Xia and N. Xu, "Search Strategy and Achieve of the Topic Search Engine Spider", Computer Systems & Applications**,** no. 3, **(2010)**, pp. 49-52.
[9] J. Wei, D. Yang and X. Liao, "Focused Crawler Based on Improved Algorithm of Web Content Similarity", Computer and Modernization, no. 9, **(2011)**, pp. 1-4.
[10] D. A. Grossman and O. Frieder, "Information Retrieval: Algorithms and Heuristics (2nd Edition)", Springer, New York **(2004)**.

