ISSN No. 0976-5697

Volume 3, No. 7, Nov-Dec 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

# New Generation Focused Crawler

Jitasha Mishra*
Computer Science and Engineering
Gandhi Institute of technology and management
Bhubaneswar, India
sibu_124@yahoo.co.in

Debashis Hati
Computer Science and Engineering
Gandhi Engineering College
Bhubaneswar, India
d_hati@yahoo.com

Amritesh Kumar
Computer Science and Engineering
Gandhi Institute of technology and management
Bhubaneswar, India
amritesh.kiit@gmail.com

Lizashree Mishra
Computer Science and Engineering
Gandhi Institute of technology and management
Bhubaneswar, India
lizashreemishra@gmail.com

***Abstract:*** *Vertical* search engines use focused crawlers as their key component and develops some specific algorithms to select web pages relevant to some pre-defined set of topics. Therefore, to effectively build up a semantic pattern for specific topics it is extremely important for such search engines. Crawlers are software which can traverse the internet and retrieve web pages by hyperlinks. A focused crawler traverses the web selecting out relevant pages to a predefined topic and neglecting those out for prioritizing the URL queue. While surfing the internet it is difficult to deal with method that analyzes the reference-information among the pages, relevant pages and to predict which links lead to quality pages. In our proposed work we calculate the link score based on page rank and average relevancy score of parent pages (because we know that the parent page is always related to child page which means that for detailed information any author prefers the child page). After finding out the link score, we compare the link score with some threshold value. If link score is greater than or equal to threshold value, then it is relevant link. Otherwise, it is discarded. Focused crawler first fetches that link which has greater value compared to all link scores and threshold

***Keywords:*** vertical search engine, focused crawler, page rank, vector space model

## I. INTRODUCTION

The world-wide Web can be modeled as a very large graph with nodes representing pages and edges representing hyperlinks. Thanks to dynamically generated content, the Web graph is infinitely large. Page content and hyperlinks change continually. Any centralized Web search service must first fetch a large number of Web pages over the Internet using a Web crawler, and then subject the local copies to indexing and other analysis. At anytime during its execution, a Web crawler has a set of pages that have been fetched, and a frontier of unexplored hyperlinks encountered on fetched pages. Given finite network resources, it is critical for the crawler to choose carefully the subset of frontier hyperlinks it should fetch next. Depending on the application and user group, it may be beneficial to preferentially acquire pages that are highly linked, pages that pertain to specific topics, pages that are likely to mention specific structured information, pages that score highly with respect to queries submitted frequently to the search engine, pages that change frequently, and so on. Focused Web crawling is a generic term for employing hyperlink and text mining techniques to prioritize the crawl frontier to maximize the harvest of qualified or preferred pages, while minimizing communication and computation effort on other pages. The network resources thus saved may be used, for example, to monitor crawled pages more aggressively for changes. Focused Web crawling is commonly used to build vertical search services catering to one or few topical interests [1].

In this paper, our proposed work is to calculate the link score based on page rank and average relevancy score of parent pages (because we know that the parent page is always related to child page which means that for detailed information any author prefers the child page). After finding out the link score, we compare the link score with some threshold value. If link score is greater than or equal to threshold value, then it is relevant link. Otherwise, it is discarded. Focused crawler first fetches that link which has greater value compared to all link scores and threshold.

## II. RELATED WORK

Many different Web analysis algorithms have been proposed in previous studies. In general, they can be categorized into two types: content-based Web analysis algorithms and link-based Web analysis algorithms. Content-based analysis algorithms analyze the actual HTML content of a Web page to obtain relevance information about the page itself. For example, key words or phrases can be extracted from the body text by using document indexing techniques to determine whether the page is relevant to a target domain. Web pages also can be compared to *Standard*

*Documents* that are already known to be relevant to the target domain using the *Vector Space Model* [2]. The Vector Space Model has been used in many existing focused crawlers [3,4,5]. Focused crawlers are programs designed to selectively retrieve web pages relevant to a specific domain for the use of domain- specific search engines and digital libraries. Unlike the simple crawlers behind most general search engines which collect any reachable Web pages in breadth-first order, focused crawlers try to "predict" whether or not a target URL is pointing to a relevant and high-quality Web page before actually fetching the page. In addition, focused crawlers visit URLs in an optimal order such that URLs pointing to relevant and high-quality Web pages are visited first, and URLs that point to low-quality or irrelevant pages are never visited.

There has been much research on algorithms designed to determine the quality of Web pages. However, most focused crawlers use local search algorithms such as *best- first search* to determine the order in which the target URLs are visited [6]. A focused crawler is a program used for searching information related to some interested topics from the Internet. The main property of focused crawling is that the crawler does not need to collect all web pages, but selects and retrieves relevant pages only. Because the crawler is only a computer program, it cannot determine how relevant a web page is [7]. In order to find pages of a particular type or on a particular topic, focused crawlers aim to identify links that are likely to lead to target documents, and avoid links to off topic branches. However the concept of prioritizing unvisited URLs on the crawl frontier for specific searching goals is not new, and Fish-Search and Shark-Search were some of the earliest algorithms for crawling for pages with keywords specified in the query [8]. In Fish-Search, the system is query driven. Starting from a set of seed pages, it considers only those pages that have content matching a given query (expressed as a keyword query or a regular expression) and their neighborhoods (pages pointed to by these matched pages).

Shark- Search is a modification of Fish-search which differs in two ways: a child inherits a discounted value of the score of its parent, and this score is combined with a value based on the anchor text that occurs around the link in the Web page. There are many approaches to select the strategy for focused crawler. Many researchers have written their approaches based on link analysis. For example, Effective Focused Crawling based on content and link structure analysis has been proposed for link analysis based on URL score, anchor score and relevance score and HAWK: A Focused Crawler with Content and Link Analysis. Some have written their approaches based on page rank value. For example, An Application of Improved Page Rank in Focused Crawler based on To-page rank value and an Improvement of Page Rank for Focused Crawler based on T page rank. Some have written based on ontology. For example, A Survey in Semantic Web Technologies-Inspired Focused Crawlers and A Transport Service Ontology-based Focused Crawler based on ontology. Some have developed based on meta search and content block partition "A Framework of a Hybrid Focused Web Crawler." Some have developed rule based focused crawler. For example, Design of an Enhanced Rule based Focused Crawler and URL rule based focused crawler.

A working process of a focused crawler is composed of two main steps. The first step is to determine the starting URLs and specify user interest. The crawler is unable to traverse the Internet without starting URLs. The second step in a focused crawling process is the crawling method. In theoretical point of view, a focused crawler smartly selects a direction to traverse the Internet. A clever route selection method of the crawler is to arrange URLs so that the most relevant ones can be located in the first part of the queue. The queue will then be sorted by relevancy in descending order [9]. The performance and efficiency of a focused crawler is mainly determined by the ordering strategy that determines the order of page retrieval.

## III. PROPOSED ARCHITECTURE

Seed URLs are extracted by threesearches.com. Now by using link extractor tool, we find out the all forward links of seed link because the seed page link is most topic relevant link for query. Focused crawler fetches the web page of seed URLs. Frontier is initialized by seed URLs. It contains only unvisited URLs. It uses the priority queue. A URL which has higher URLs score is given higher priority. The higher priority URL is input to the web page downloader. Web page downloader is used to take input URLs which has higher priority from frontier and downloads the web page from internet. After fetching all forward link web pages of particular seed URL we calculate the page rank and relevancy score of all back link of child link of seed URL. If the average relevancy score is more than 0.5 then this link will be sent to the frontier to fetch the web pages else it will be rejected.

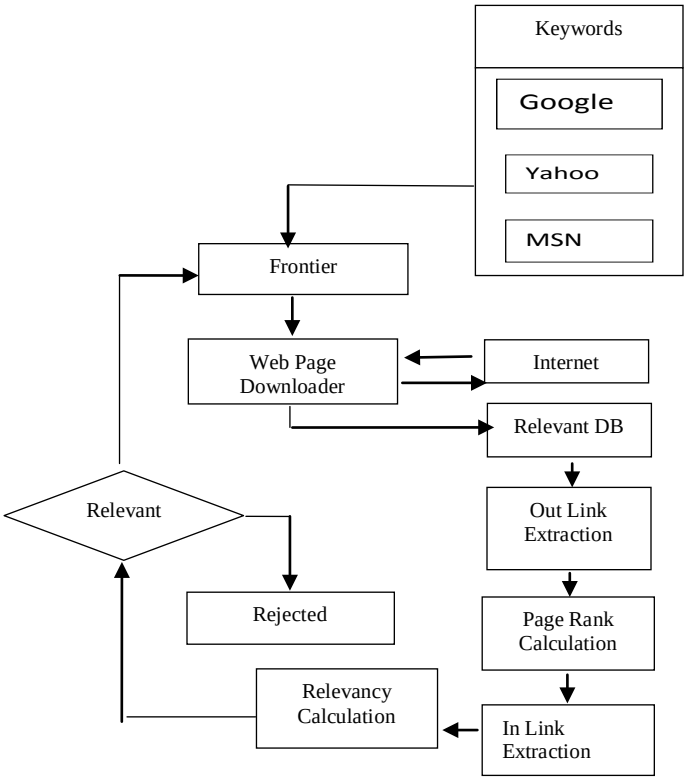


Figure 1. architecture of focused crawler

## IV. PROPOSED WORK

### *A. Seed URL Extraction:*

In our proposed approach, seed URLs are extracted by one search engine known as threesearches.com. We put a query in this search engine and it shows the result of three most popular search engines like Google, Yahoo, and MSN search. We take resulting URLs which are common in all the three search engines. URLs which are common in all three search engines like Google, Yahoo, and MSN search, we assume that those common search result URLs are most relevant for this query and thus these URLs are the seed URLs, and a URL which is common in three search engines result by experiment belongs to relevant category seed URLs. We assume also that a resulting URL which is common in three search engines result is not most relevant for topics but it is relevant for topics and we are putting it also in seed URLs category. For example, we put a query "computer books" on a threesearches.com and common result of all three search engines like Google, Yahoo and MSN search are extracted. Here, two outputs are www.freecomputerbooks.com and www.computer-book.us.

### *B. Topic Specific Weight Table Construction:*

Weight table defines the crawling target. The topic name is sent as a query to the Google Web search engine and the first 7 results are retrieved. The retrieved pages are parsed, stop words such as "the" and "is" are eliminated. Words are stemmed using the porter stemming algorithm and the term frequency (tf) of each word is calculated. The 10 words with most occurrences are recorded and used to build the topic vector. The topic vector would be a vector containing 10 elements representing the term weights.
The term weights = {t1 t2 …… ti……t10} are computed as:.

$$t_i = n_i / n_{\max} \qquad (1)$$

Table 1 Topic specific weight table

| Terms | Weight |
|---|---|
| Book | 1 |
| Free | 0.894259882 |
| Program | 0.459214501 |
| Computer | 0.380664665 |
| Web | 0.25679758 |
| Ebook | 0.25679758 |
| Site | 0.250755287 |
| Linux | 0.223564954 |
| Java | 0.208459214 |
| Post | 0.187311178 |

where $n_i$ is the term occurrences in the web page and $n_{max}$ is the frequency of the term with most occurrences.

### *C. Page Rank Calculation:*

As Page Rank is a link analysis algorithm that assigns a numerical weighting to each element of a hyperlinked set of documents. The rank value indicates an importance of a particular page. Here we calculate the rank of a page with the help of one web site, named www.prchecker.info. We assume threshold value for page rank as 0.2.

### *D. Relevancy Calculation:*

When we calculate the page rank of a web page sometimes we find that some irrelevant link also appears in first 10 to 20 results because these links have more numerical weight or we can say these links have more page rank value. But it is irrelevant for some specific topics. So to remove this limitation, our proposed work is based on page rank with relevancy calculation, that calculates the relevancy of the link with topic keywords. Firstly, we find out all the back link of webpages by using the website www.backlinkwatch.com. Once the back link is found we open the individual link and open its source codes. Then within the source codes we search where the particular link is present after obtaining the link we note the whole paragraph within and . Those descriptions within the paragraph with the link altogether are compared to find the similarity between the topic keyword and the selected paragraph with the help of vector space model. We assume threshold value for relevancy calculation as 0.5.

$$R(t,c) = \frac{\sum_{i=1}^{n} wkt_i * wkc_i}{\sqrt{(\sum_{i=1}^{n} (wkt_i * wkt_i) * \sum_{i=1}^{n} (wkc_i * wkc_i))}} \qquad (2)$$

Where,
$Wkt_i$= weight of terms in topic specific weight table
$Wkc_i$= weight of terms in backlinks of a link

## V. PROPOSED ALGORITHM

Step 1: Extract page from seed URLs
/*with the help of web page downloader we extracts the web pages from internet and store it in Relevant DB*/
Step 2: for seed_page =1 to total_seed page
Step 3: Extract all child links of each seed page.
/*By using link extractor tool we find out all out links of each seed pages.*/
Step 4: Calculate page rank of each child links of Seed web pages.
/* Here we are using www.prchecker.info web site to calculate the page rank of each child links*/
Step 5: If page_rank of link>= threshold_value_1
/*threshold _value_1= 0.2*/
Step 6: valid page rank.
Step 7: Extract all in links of each child links which has valid page rank.
/*Here we are using one website www.backlinkwatch.com to find out the all inlinks of each link.*/
Step 8: Calculate Relevancy Score.
Step 9: if relevancy_score>=threshold_value_2
/*threshold_value_2=0.5*/
Step 10: Links are relevant and store it in frontier
Step 11: Else
Step 12: Links are rejected.
Step 13:Stop.

We have implemented this algorithm by using AppletContext Interface, URL class, getAppletContext() and show Document () methods in JAVA programming.

## VI. RESULT AND ANALYSIS

When we type a query say "computer books" in some popular search engine, we get different links in top 20 results by page rank. We select any one of the link say http://www.ieee.org/publications_standards/publications/books/index.html then we find the backlinks of the particular link. Now we calculate the relevancy score of all inlinks by considering the terms within the  and  of the html codes with the topic specific table. http://www.ieee.org/publications_standards/publications/books/index.html

There are 13 backlinks of the selected link with their relevancy score, they follows :-

http://www.cvel.clemson.edu/emc/info/emc-books.html

Relevancy Value= 0

http://www.mtt.org/benefits-of-membership.html

Relevancy Value= $R(t,c) = \frac{1.064199395}{\sqrt{2.633697906}}$

$R(t,c) = \frac{1.064199395}{1.622867187}$

$R(t,c) = 0.65575261$

http://www.r10sac.org/resources/student/discounts.html

Relevancy Value= $R(t,c) = 0.0$

http://www.math.iit.edu/~kaul/Journals.html

Relevancy Value= $R(t,c) = 0.0$

http://www.cvel.clemson.edu/emc/info/si-books.html

Relevancy Value= $R(t,c) = 0.0$

http://mtt.org/benefits-of-membership.html

Relevancy Value= $R(t,c) = 0.65575261$

http://www.cvel.clemson.edu/modeling/info/books.html

Relevancy Value= $R(t,c) = 0.0$

http://www.acml egypt.com/ACML%20Publishers/IEEE.htm

Relevancy Value= $R(t,c) = 0.0$

http://www.cvel.clemson.edu/auto/info/books.html

Relevancy Value= $R(t,c) = 0.0$

http://www.ieeevbit.org/ieeevbit/discounts/

Relevancy Value= $R(t,c) = 0.0$

http://ieeevbit.org/ieeevbit/discounts/

Relevancy Value= $R(t,c) = 0.0$

http://www.ieee.org.pr/membership-services/publications standards/

Relevancy Value= $R(t,c) = 0.635157651$

http://ieee-tcet.com/index.php?option=com_content&view=section &layout=blog&id=16&Itemid=67

Relevancy Value= $R(t,c) = 0.635157651$

Now we calculate the average of relevancy score of all those links:

$$R_{avg}(t,c) = \frac{0 + 0.65575261 + 0 + 0 + 0 + 0.65575261 + 0 + 0 + 0 + 0 + 0 + 0.635157651 + 0.635157651}{13}$$

$$R_{avg}(t,c) = \frac{2.581820522}{13}$$

$$R_{avg}(t,c) = 0.198601578$$

This URL has been selected by some popular search engine in top 20 result for topic "computer book". But according to our proposed approach this URL is not relevant for topic "computer book", because relevance score is very less $R_{avg}(t,c) = 0.198601578$. So, it is irrelevant link. This will be rejected.

## VII. CONCLUSION

Focused crawling methods are the important members of the search engine family. But one of the key problems of vertical search engines is to develop an effective algorithm for the topic-specific search and the similarity measurement among topics. One approach for solving this problem is to analyze the retrieved URLs and the URL patterns and their formal representation are discovered. In this paper we propose to calculate the link score based on page rank and average relevancy score of parent pages (because we know that the parent page is always related to child page which means that for detailed information any author prefers the child page). After finding out the link score, we compare the link score with some threshold value. If link score is greater than or equal to threshold value, then it is relevant link. Otherwise, it is discarded.

## VIII. REFERENCE


[1] S. Chakrabarti, M. van den Berg and B. Dom. "Focused crawling: a new approach to topic-specific Web resource discovery", 8th International WWW Conference, May 1999.

[2] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," inProc. of the ACM-SIAM Symposium on Discrete Algorithms, San Francisco, California, USA, 1998.

[3] D. Bergmark, "Collection Synthesis," in Proc. of JCDL 2002, Portland, Oregon, USA, 2002.

[4] J. Dean, and M. R. Henzinger, "Finding Related Pages in the World Wide Web," in Proc. of the 8th International WWW Conference, Toronto, Canada, 1999

[5] M. Kitsuregawa, M. Toyoda, and I. Pramudiono, "WEB Community Mining and WEB Log Mining:Commodity Cluster Based Execution," in Proc. of the 13th Australasian Database Conference, Melbourne, Australia, 2002.

[6] J.Qin, Y.Zhou, M.Chau, "Building Domain-Specific Web Collections for Scientific Digital Libraries: A Meta-Search Enhanced Focused Crawling Method", IEEE Conference on Digital Libraries,2004.

[7] X. Zhang, T. Zhou, Z.Yu and D.Chen, "URL Rule Based Focused Crawlers", IEEE International Conference on e-Business Engineering,2008.

[8] A. Pal, D. S. Tomar and S.C. Shrivastava. "Effective Focused Crawling Based on Content and Link Structure Analysis",

(IJCSIS)International Journal of Computer Science and Information Security,Vol. 2, No. 1,June 2009.

[9] M. Yuvarani, N. Ch. S. N. Iyengar and A. Kannan, "LSCrawler:A Framework for an Enhanced Focused Web Crawler based on Link Semantics" in Proceedings of the 2006 IEEE/WIC/ACM International Conferenceon WebIntelligence.

.