Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 11; April-June, 2015 pp. 1-6

http://www.krishisanskriti.org/acsit.html

Focused Web Crawler

Dvijesh Bhatt[1], Daiwat Amit Vyas[2] and Sharnil Pandya[3]

[1,2,3]Institute of Technology, Nirma University
E-mail: [1]dvijesh.bhatt@nirmauni.ac.in, [2]daiwat.vyas@nirmauni.ac.in,
[3]sharnil.pandya@nirmauni.ac.in

Abstract—*With the rapid development and increase in global data on World Wide Web and with increased and rapid growth in web users across the globe, an acute need has arisen to improve and modify or design search algorithms that helps in effectively and efficiently searching the specific required data from the huge repository available. Various search engines use different web crawlers for obtaining search results efficiently. Some search engines use focused web crawler that collects different web pages that usually satisfy some specific property, by effectively prioritizing the crawler frontier and managing the exploration process for hyperlink. A focused web crawler analyzes its crawl boundary to locate the links that are likely to be most relevant for the crawl, and avoids irrelevant regions of the web. This leads to significant savings in hardware and network resources, and helps keep the crawl more up-to-date.The process of focused web crawler is to nurture a collection set of web documents that are focused on some topical subspaces. It identifies the next most important and relevant link to follow by relying on probabilistic models for effectively predicting the relevancy of the document. Researchers across have proposed various algorithms for improving efficiency of focused web crawler. We try to investigate various types of crawlers with their pros and cons.Major focus area is focused web crawler. Future directions for improving efficiency of focused web crawler have been discussed. This will provide a base reference for anyone who wishes in researching or using concept of focused webcrawler in their research work that he/she wishes to carry out. The performance of a focused webcrawler depends on the richness of links in the specific topic being searched by the user, and it usually relies on a general web search engine for providing starting points for searching.*

Keywords*: Focused Web Crawler, algorithms, World Wide Web, probabilistic models.*

1. INTRODUCTION

Innovations in the field of web technology and data mining has had a significant impact on the way web based technologies are being developed. Internet has been the most useful technology of modern times and has become the largest knowledge base and data repository. Internet has various diversified uses like in communication, research, financial transactions, entertainment, crowdsourcing, and politics and is responsible for the professional as well as the personal development of individuals be he/she be a technical person or a non-technical person. Every person is so acquainted with online resources that somehow or the other is dependent on online resources for his/her day to day activities.

Search engines [6]are the most basic tools used for searching over the internet. Web search engines are usually equippedwith multiple powerful web page search algorithms. But with the explosive growth of the World Wide Web,searching information on the web is becoming an increasingly difficult task. All this poses an unprecedented scaling challenges for the general purpose crawlers and search engines. Major challenges like, to make users available with the fastest possible access to the requested information in a most precise manner, making lighter web interfaces, etc are being addressed by researchers across the globe.

Web Crawlers are one of the main components ofweb search engines i.e. systems that assemble a corpus of web pages, indexthem, and allow users to issue queries against the index and find the webpages that match the queries fired by the users.Web crawling is the process by which system gather pages from the Web resources, inorder to index them and support a search engine that serves the user queries. The primary objective of crawlingis to quickly, effectively and efficiently gather as many useful web pages as possible, together with the link structure that interconnects them and provide the search results to the user requesting it. A crawler must possess features like robustness, scalability, etc.

The first generation of crawlers on which most of the search engines are based, rely heavily on the traditional graph algorithms like the breadth-first search and the depth-first search to index the web. In the NetCraft Web Server survey, the Web is measured in the number of Websites which from a small number in August 1995 increased over 1 billion in April 2014. Due to the vast expansion of the Web and the inherently limited resources in a search engine, no single search engine is able to index more than one-third of the entire Web. This is the primary reason for general purpose web crawlers having poor performance.

The basic purpose of enhancement in the search results specific to some keywords can be achieved through focused web crawler. [3] With the exponential increase in the number of Websites, more emphasis is made in the implementation of focused web crawler. It is a crawling technique that

dynamically browses the Internet by choosing specific modes that maximize the probability of discovering relevant pages, given a specific search query by user. Predicting the relevance of the document before seeing its contents i.e. relying on the parent pages only, is one of the central characteristic of focused crawling because it can save significant amount of bandwidth resources. Instead of collecting and indexing all accessible web documents to be able to answer to all ad-hoc queries, a focused web crawler analyzes its crawl boundary to find the links that are likely to be most relevant for the crawl, and avoids irrelevant regions of the web making it more focused on some specific keywords. Adopting to this it helps significant in savings in hardware and network resources, and helps keep the crawl more up-to-date.

In this paper, we study a focused web crawler[1, 12] which seeks, acquires, indexes and maintains pages on a specific set of topics that represent a relatively narrow segment of the web.

The flow of topics in paper is as mentioned. In the next section, the classification [2] and types of crawlers are described and all the types of crawlers are discussed in brief. In chapter 3 various challenges in web crawling are discussed.In chapter 4, basic overview of the focused web crawler is included. This includes the basic functionality of the focused web crawler and the principle behind working of focused web crawler is discussed. In chapter 5, the architecture of focused web crawler is drilled upon. The various components of the focused web crawler and the functionality and working of those components is discussed. In section 6, some algorithms which are discussed to improve the efficiency of the focused web crawler are explained. Many algorithms [5, 18] to better the performance of the focused web crawler are discussed. Out of those, some algorithms are explained in that section. In the last chapter, the paper is concluded and future work is outlined.

2. RELATED WORK

The World Wide Web is experiencing an exponential growth[6], both in size and the number of users accessing. The quantity and variety of documentation available, poses the problem of discovering information relevant to a specific topic of interest from users perspective. The instruments developed to ease information recovering on the Internet suffer from various limitations. Web directories cannot realize exhaustive taxonomies and have a high maintenance cost due to the need for human classification of new documents.

Web crawlers are used by web search engines and a few other sites to update their web content or indexes of other sites' web content. A Web crawler is an Internet bot or a robotic crawler that systematically browses the World Wide Web, typically for the purpose of web indexing, which helps in faster access of information. A Web crawler has many names like Spider, Robot, Web agent, Wanderer, worm etc.

The crawlers can be divided into two parts: - Universal crawlers and Preferential crawlers. Further the preferential crawlers can be divided into two parts: - the focused web crawler[2]and the topical crawlers [4, 14]. The topical crawlers are of types: - Adaptive topical crawlers and Static crawlers. The evolutionary crawlers, reinforcement learning crawlers, etc. are examples of adaptive topical crawlers and the best-first search, PageRank algorithms, etc. are examples of static crawlers.

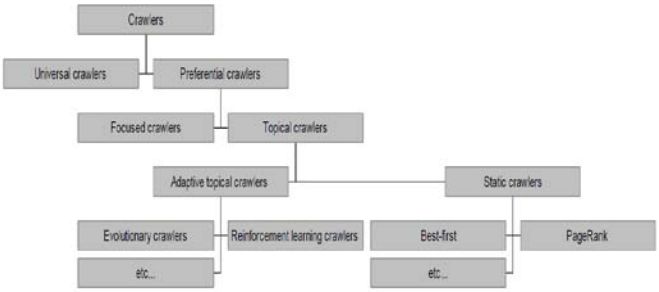

Fig. 1: Classification of Web Crawlers

The Universal crawlers support universal search engines. It browses the World Wide Web in a methodical, automated manner and creates the index of the documents that it accesses. The Universal crawler first downloads the first website. It then goes through the HTML and finds the link tag and retrieves the outside link. When it finds a link tag, it adds the link to the list of links it plans to crawl. Thus, as the universal crawler crawls all the pages found, huge cost of crawl is incurred over many queries from the users. The universal crawler is comparatively expensive.

The preferential crawlers are the topic based crawlers. They are selective in case of web pages. They are built to retrieve pages within a certain topic. Focused web crawlers and topical crawlers are types of preferential crawlers that search related to a specific topic and only download a predefined subset of web pages from the entire web. The algorithms for the topical and focused web crawlers started with the earlier breadth-first search and the depth-first search. Now, we can see a variety of algorithms. There is De Bra's fish search[8], Shark search which is a more aggressive variant of the De Bra's fish search. Another algorithms using concept of topical crawlers are the link structure analysis, the page rank algorithm[11, 19] and the hits algorithm. Several machine learning algorithms are also used in focused web crawlers.

The adaptive topical crawlers[8, 16, and 17] and the static crawlers [9]are type of topical crawlers. If a focused web crawler includes learning methods in order to adapt its behavior during the crawl to the particular environment and its relationships with the given input parameters, e.g. the set of retrieved pages and the user-defined topic, the crawler is named adaptive. Static crawlers are simple crawlers not adapting to the environment they are provided.

The examples of the adaptive and the static crawlers are shown in the Figure1.

3. CHALLENGES IN WEB CRAWLING

The major issues in web crawling faced globally are as mentioned below:

- Collaborative Web Crawling
- Crawling the large repository on web
- Crawling Multimedia Data
- Execution time
- Scale: Millions of pages on web
- No central control of web pages
- Large number of web pages emerging daily

Various mechanisms and alternatives have been designed to address the above mentioned challenges. The rate at which the data on web is booming makes web crawling a more challenging and uphill task. The web crawling algorithms need to constantly match up with the demanding data from users and growing data on web.

4. OVERVIEW OF FOCUSED WEB CRAWLER

Tremendous advances in World Wide Web has created many scaling challenges for the general purpose crawlers and search engines. So in this paper we have described the working of a crawler called the focused web crawler that can solve the problem of information retrieval from the huge content of the web more effectively and efficiently.

Focusedweb crawler is a hypertext system that seeks, acquires, indexes and maintains pages on a specific set of topics that represent a relatively narrow segment of the web. It entails a very small investment in the hardware and network resources and yet achieves respectable coverage at a rapid rate, simply because there is relatively little to do. Thus, web content can be managed by a distributed team of focused web crawlers, each specializing in one or a few topics[10]. Each focused web crawler will be far more nimble in detecting changes to pages within its focus than a crawler that is crawling the entire web. The focused web crawler is guided by a classifier which learns to recognize relevance from examples embedded in a topic taxonomy, and a distiller which identifies topical vantage points on the web.

A simple structure of crawler is shown in Fig. 2. Crawling starts with a set of seed URLs and URLs to be fetched are stored in a queue structure, called “URL queue”. Then multiple threads executes simultaneously. Each thread gets the URL from the queue and fetches the corresponding web pages from the server. Later, this page is parsed to extract links and these links are appended to the URL queue to be fetched later. A real life crawler is much more complex than this structure to consider issues like politeness policy i.e. do not request many web pages from the same server at the same time.

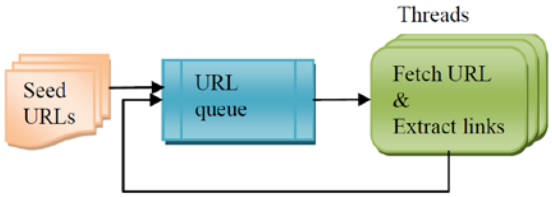

Fig. 2 : Structure of Crawler

Focusedweb crawlers use vertical search engines to crawl web pages specific to a target topic. Fig. 3 represents the structure of the focused web crawler. The only difference compared to the general crawler is the topic classifier which makes it more precise [21]. Each fetched page is classified to predefined target topic(s). If the page is predicted to be on-topic, then its links are extracted and are appended into the URL queue. Otherwise the crawling process does not proceed from this page. This type of focusedweb crawler is called “full-page” focused web crawler since it classifies the full page content. In other words, the context of all the links on the page is the full page content itself.

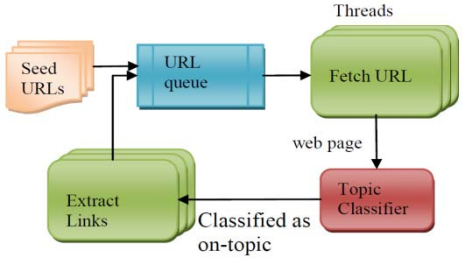

Fig. 3: Structure of Focused Web Crawler

5. ARCHITECTURE OF FOCUSED WEB CRAWLER

Fig. 4 shows the architectureof focused webcrawler. This type of web crawler makes indexing more effectives ultimately helping us in achieving the basic requirement of faster and more relevant retrieval of information from the large repository of World Wide Web. Many search engines have started using this technique to give users a more rich experience while accessing web content ultimately increasing their hit counts. The various components of the system, their input and output, their functionality and the innovative aspects are shown below.

Seed detector: - The functionality of the Seed detector is to determine the seed URLs for the specific keyword by retrieving the first n URLs. The seed pages are detected and assigned a priority based on the PageRank algorithm or the hits algorithm or algorithm similar to that.

Crawler Manager: - The Crawler Manager is a significant component of the system next to the Hypertext Analyzer. The component downloads the documents from the global network. The URLs in the URL repository are fetched and

assigned to the buffer in the Crawler Manager. The URL buffer is a priority queue. Based on the size of the URL buffer, the Crawler Manager dynamically generates instance for the crawlers, which will download the document. For more efficiency, crawler manager can create a crawler pool. The manager is also responsible for limiting the speed of the crawlers and balancing the load among them. This is achieved by monitoring the crawlers.

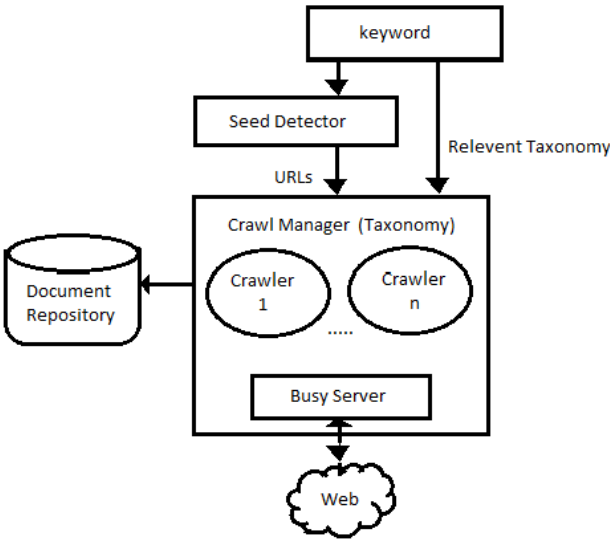

Fig. 4: Architecture of Focused Web Crawler

Crawler: - The crawler is a multi-thread java program, which is capable of downloading the web pages from the web and storing the documents in the document repository. Each crawler has its own queue, which holds the list of URLs to be crawled. The crawler fetches the URL from the queue. The same or the other crawlers would have sent a request to the same server. If so, sending the request to the same server will result in overloading the server. The server is busy in completing the request that have come from the crawlers that have sent request and awaiting for the response. The server is made synchronized. If the request for the URL has not been sent previously, the request is forwarded to the HTTP module. This ensures that the crawler doesn't overload any server.

HTTP Protocol Module: - HTTP Protocol Module, sends the request for the document whose URL has been received from the queue. Upon receiving the document, the URL of the document downloaded is stored in the URL fetched along with the timestamp and the document is stored in the document repository. Storing the downloaded URL avoids redundant crawling, hence making the crawler more efficient.

Link Extractor: - The link extractor extracts the link from the documents present in the document repository. The component checks for the URL being in the URL fetched. If not found, the surrounding text preceding and succeeding the hyperlink, the heading or sub-heading under which the link is present, are extracted.

Hypertext Analyzer: - The Hypertext Analyzer[13] receives the keywords from the Link Extractor and finds the relevancy of the terms with the search keyword referring the Taxonomy Hierarchy.

A focused web crawler has the following advantages ascompared to the other crawlers:

a) The focused web crawler steadily & easily acquires the relevant pages by focusing on specific keywords, while other crawler quickly loses its way, even though they start from the same seed set.

b) It can discover valuable Web pages that are many links away from the seed set, and on the other hand can acquire millions of Web pages that may lie within same radius. This helps in having a high quality collections of Web documents on specific topicsi.e focused on some keywords etc..

c) It can also identify regions of the Web that are dynamic or grow as compared to that are relatively static.

6. ALGORITHMS TO INCREASE EFFICIENCY

Following are some algorithms that can be implemented over the focusedweb crawler to improve the efficiency and the performance of the crawler.

Best-First Search: - The Best-First Search algorithm can be implemented to search for the best URL which explores the URL queue to find the most promising URL. The crawling order can be determined by the BFS algorithm. It is based on a heuristic function $f(n)$. The value of the function is evaluated for each URL present in the queue and the URL with the most optimal value of the function $f(n)$ is given the highest priority and priority list can be formed and the URL with the highest priority can be fetched next. The optimal URL can be fetched using the BestFirst Search algorithm but the memory and the time consumed are very high and so more enhanced algorithms have been proposed.

Fish-Search: - The FishSearch algorithm[8] can be implemented for dynamic search. When the relevant information is found, the search agents continue to look for more information and in the absence of information, they stop. The key principle of the algorithm is as follows: It takes as input a seed URL and a search query. It then dynamically builds the priority list of the next URLs. At each step, the first URL is popped from the list and is processed. As each document's text becomes available, it is analyzed by a scoring component evaluating whether it is relevant or irrelevant to the search query (*1-0* value) and, based on that score, a heuristic decides whether to pursue the exploration in that direction or not: Whenever a document source is fetched, it is scanned for links. The URLs pointed to by these links (denoted as "children") are each assigned a depth value. If the parent is relevant, the depth of the children is set to some predefined value. Otherwise, the depth of the children is set to be one less

than the depth of the parent. When the depth reaches zero, the direction is dropped and none of its children is inserted into the list. The algorithm is helpful in forming the priority table but the limitation is that there is very low differentiation among the priority of the URLs. Many documents have the same priority. And also the scoring capability is problematic as it is difficult to assign a more precise potential score to documents which have not yet been fetched.

Shark-Search: - The Shark-Search [15]is the improved version of the Fish-Search algorithm. The Fish-Search did a binary evaluation of the URL to be analyzed and so the actual relevance cannot be obtained. In Shark-Search, a similarity engine is called which evaluates the relevance of the documents to a given query. Such an engine analyzes two documents dynamically and returns a "fuzzy" score, i.e., a score *between 0* and *1* (*0* for no similarity whatsoever, *1* for perfect "conceptual" match) rather than a binary value. The similarity algorithm can be seen as orthogonal to the fish-search algorithm. The potential score of the URL that is extracted can be refined by the meta-information contained in the links to the documents.

Genetic Algorithm: - The Genetic algorithm [7, 20]is used to improve the quality of search results obtained from focused crawling. It is an adaptive and heuristic method for increasing optimization and improving search results. It exploits several techniques inspired by biological evolution such as inheritance, selection, cross-over and mutation. It has four phases. In the first phase, the parameters like population size, generation size, cross-over rate or the probability of cross-over and mutation rate or probability of mutation rate are fixed. Initial URLs are fetched by the crawler. On the basis of Jaccard's similarity function, a fitness value is assigned to the Web page. The higher the fitness value, the more is the page similar to the domain lexicon. The Jaccard's function, based on links, is the ratio of the number of intersection links and union links between the two Web pages. The more number of common links, the higher is the Jaccard's score. After fitness values are calculated, the pages with better fitness values are selected by a random number generator. Some relevant pages are selected and the rest are discarded. Then, all outbound-links are extracted from the survived pagesand a cross-over operation is performed to select the most promising URLs. Cross-over of a URL is the sum of the fitness of all pages that contain the URL. Based on cross-over value, the URL's are sorted and put into crawling queue. The mutation operation is aimed at giving the crawler, the ability to explore multiple suitable Web communities. Random keywords from the lexicon are extracted and are run as query in well-known search engines. Top results from the search-engines and results from the cross-over phase are combined to give more optimal results.

Some challenges to deal with while using focused web crawling are as mentioned below:

Missing Relevant Pages: One issue with focused web crawlers is that they may miss relevant pages by only crawling pages that are expected to give immediate benefit.

Maintaining Freshness of Database: Many HTML pages consist of information that gets updated on daily, weekly or monthly basis. The crawler has to download these pages and updates them into the database to provide up-to-date information to the users. The crawling processbecomes slow and puts pressure on the Internet traffic if such pages are large in number. Thus, a major issue is to develop a strategy that manages these pages.

Absence of Particular Context: The focused web crawler uses the best strategy to download the most relevant pages based on some criteria. The crawler focuses on a particular topic but in the absence of a particular context, it may download large number of irrelevant pages. Thus the challenge is to develop focused crawling techniques that focus on particular context also.

Some specific mechanisms like handling freshness of database, context, making sure relevant pages are retrieved and support of indexing algorithms, we can to a very good extent improve the performance of focused web crawling

7. CONCLUSION

In this paper, we discussed about focusedweb crawler, the functionality and methodology of itsworking, various components and the various algorithms that can be implemented with the focusedweb crawler so that its efficiency can be improved. The focusedweb crawler is a system that learns the specialization from the examples and then explores the Web, guided by a relevance and popularity rating mechanism. If filters at the data acquisition level, rather than as a post-processing step. Thus, focusedweb crawler proves to have better performance than general Web crawlers. We can implement algorithms to improve the results obtained from focusedweb crawler.New algorithms can be developed, which can be applied to the focusedweb crawler, to increase the optimality of the Web page results.

REFERENCES

[1] S. Chakrabarti, M.H. Van den Berg, and B.E. Dom, "Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery," Computer Networks, vol. 31, nos. 11–16, pp. 1623–1640, 1999.

[2] KamilÇalışkan, RifatOzcan, "Comparing classification methods for link context based Focused crawlers",Department of Computer Engineering, TurgutOzal University, Ankara, Turkey.

[3] G. Pant and P. Srinivasan, "Learning to Crawl: Comparing Classification Schemes," ACM Trans. Information Systems, vol. 23, no. 4, 2005.

[4] G. Pant and P. Srinivasan, "Link contexts in classifier guided topical crawlers," Knowledge and Data Engineering, IEEE Transactions on , vol.18, no.1, pp.107-122, Jan. 2006.

[5] M.Yuvarani, N.Ch.S.N. Iyengar, A.Kannan, "LSCrawler: A Framework for an Enhanced Focused Web Crawler based on Link Semantics", Anna University, Chennai, India.

[6] "Google", http://www.google.com, 2014.

[7] Song Zheng, "Genetic and Ant Algorithms Based Focused Crawler Design", Shenzhen Securities Information Co., Ltd Shenzhen, China.

[8] Paul De Bra and LiciaCalvi."Creating Adaptive Hyper documents for and on the Web" in Proceedings of the AACE Web Net Conference, Toronto, 1997: 149-155.

[9] Xueming Li, Minling Xing, Jiapei Zhang, "A Comprehensive Prediction Method of Visit Priority for Focused Crawler", Chongqing University, Chongqing, China.

[10] I. S. Altingovde, O. Ulusoy, "Exploiting interclass rules for focused crawling," Intelligent Systems IEEE, vol. 19, pp. 66-73, 2004.

[11] Yuan Fu-yong, Yin Chun-xia, and Liu Jian, "Improvement of page rank for focused crawler," SNPD 2007: 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007.

[12] Li Sheng-tao, "Focused Web Crawling Technology," The Chinese academy of sciences institute of computing technology, master's thesis, 2002.

[13] Jiang Peng and Song Ji-hua, "A Method of Text Classifier for Focused Crawler," Journal of Chinese Information Processing, vol. 26, pp. 92-96 Nov. 2010.

[14] Peng Tao, "Research on Topical Crawling Technique for Topic-Specific Search Engine," Doctor Degree thesis of Jilin University, 2007.

[15] HersovIcim, JacovIm, and Maarek Y S, "The shark-search algorithm: an application: tailored Web site mapping,"Proc of the 7thInternational World Wide Web Conference. Brisbane: [s. n.], 1998, pp. 65-74.

[16] Hai Dong, FarookhKhadeerHussain, "Self-Adaptive Semantic Focused Crawler for Mining Services Information Discovery", Member, IEEE.

[17] A. Micarelli and F. Gasparetti, "Adaptive Focused Crawling", The Adaptive Web, 231 – 262, 2007.

[18] DuyguTaylan, MitatPoyraz, SelimAkyokuş and Murat Can Ganiz, "Intelligent Focused Crawler: Learning which Links to Crawl", Doğuş University, Istanbul, Turkey.

[19] F. Yuan, C. Yin and Y. Zhang "An application of Improved PageRank in focused Crawler", Fourth International Conference on Fuzzy Systems and Knowledge Discovery.China,vol. 2, pp. 331-335, August 2007.

[20] BanuWirawanYohanes, Handoko, HartantoKusumaWardana, "Focused Crawler Optimization Using Genetic Algorithm", Universitas Kristen SatyaWacana, Salatiga.

[21] Huo Ling Yu, Liu Bingwu, Yan Fang, "Similarity Computation of Web Pages of Focused Crawler", Beijing Wuzi University, China.