Volume 3, Issue 4, April 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Web Crawlers: Taxonomy, Issues & Challenges

Dr Rajender Nath
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra, India.

Khyati Chopra
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra, India.

Abstract— with increase in the size of Web, the search engine relies on Web Crawlers to build and maintain the index of billions of pages for efficient searching. The creation and maintenance of Web indices is done by Web crawlers, the crawlers recursively traverses and downloads Web pages on behalf of search engines. The exponential growth of Web poses many challenges for crawlers.This paper makes an attempt to classify all the existing crawlers on certain parameters and also identifies the various challenges to web crawlers.

Keywords— WWW, URL, Mobile Crawler, Mobile Agents, Web Crawler.

I. INTRODUCTION

World Wide Web [1] is a collection of text documents, images, multimedia and other resources, which are linked by URLs and hyperlinks, usually accessed by web servers. According to the estimation WWW contains more than 2000 billion visible pages on web [2]. Due to large number of pages on web, the search engine depends upon web crawlers to create and maintain indices for the web pages. A web crawler is a program which, giving one or more than one seed URLs, downloads the web pages associated with these URLs, extracts any hyperlinks present in them, and iteratively continues to download the web pages identified by these hyperlinks. Web crawlers are short software codes also called wanderers, automatic indexers, Web robots, Web spiders, ants, bots, Web scutters. In order to download a document, Crawler picks up its initial URL (seed URL) and depending on the host protocol, thus downloads the document from the Server. The search engine relies on massive collections of web pages that are acquired with the help of Crawlers, which traverses Web by following URLs and hyperlinks and storing downloaded pages in a depository that is later indexed for efficient execution of user queries [2].

II. RELATED WORK

Matthew Gray [3] wrote the first Crawler, the World Wide Web Wanderer, which was used from 1993 to 1996. Distributed Crawlers developed from 1993 until 1997, were of type standalone [12]. In 1998, Google introduced its first distributed crawler, which had distinct centralized processes for each task and each central node was a bottleneck. After some time, AltaVista search engine introduced a crawling module named as Mercator [13], which was scalable, for searching the entire Web and extensible. UbiCrawler [15] a distributed crawler by P. Boldi, B. Codenotti, M. Santini and S. Vigna, with multiple crawling agents, each of which run on a different computer. Number of threads run in parallel by each agent, who had several TCP connections open at a time. Some agents executes on remote machines to improve downloading speed [16]. IPMicra [17], by Odysseas Papapetrou and George Samaras, a location-aware distributed crawling method, which utilized an IP address hierarchy, crawl links in a near optimal location aware manner. IPMicra, was an extension of UCYMicra, allows crawling the links in a near optimal location aware manner. The basis for IPMicra was IP address hierarchy tree, which is created using information from the four Regional Internet Registries. The hierarchy was then used to delegate the web sites to near migrating crawlers. SAN model of an UbiCrawler [18] presented an improved UbiCrawler-based architecture for a distributed Web crawler. The proposed model focused to improve the crawler's accuracy. The Web crawler consists of distinct agents; each agent was with finite number of threads. Each thread selected a URL from its queue, downloaded it, converted it to HTML, analysed it to find new URLs, and then inserted the URLs of its own site into its queue and send the remaining URLs to its parent agent. By using consistent hashing, parent agent selected the URLs that were sent to other agents. Focused crawling was introduced by [9] in 1999, which indicate crawling of topic-specific web pages. To save hardware and network resource consumption, a focused crawler analysed the crawled pages to find links that were most relevant to topic and ignored the irrelevant clusters of the web.

Chakrabarti,
scheme, a frequency change estimator was also used, to calculate the probability of change in page. The purposed scheme save the bandwidth if the probability of change in page was less than or equal to 10% or greater than or equal to 80%, by not sending information about these pages in the OLDDATABASE File. It was shown that the Modified mobile crawler reduces the network load and saves bandwidth by compressing the modified pages at remote site before sending them to Berg and Dom [9] described focused web crawler with three components: a classifier to evaluate the web page relevant to the chosen topic, a distiller to find hub pages inside the relevant Web regions, and a reconfigurable crawler that was governed by the classifier and distiller.

In [10] Novel Mobile Crawler System introduced by Rajender Nath and Satinder Bal , to find out change in the page three attributes: last modified date, number of keywords and number of URLs were used. The aim was to preserve the bandwidth and reduce load on the network. From the experimental work it was found that Mobile Crawler reduced 37 % load on the network. The proposed scheme reduced the load on the network to one fourth by making use of compression. Modified Mobile Crawler introduced by Rajender Nath and Satinder Bal [11] used three modules: OLDDATABASE file, COMPARATOR Program and ANALYZER Program, to find out whether the crawled page was modified or not modified. In the purposedthe search engine, thus reducing the amount of data transferred over the network. Learnable Focused Meta Crawling introduced by Mukesh Kumar and Renu Vig [19], was based upon Tf -Idf (Term frequency Inverse document frequency) semantics and hub score learning. Four consecutive runs of the proposed crawler were made, to study the effect of learning. The results were plotted as graph between the precision value and the number of pages downloaded by the various crawling phases.

III. GENERAL ARCHITECTURE OF WEB CRAWLER

The general architecture of a crawler based search engine is shown in Fig. 1.

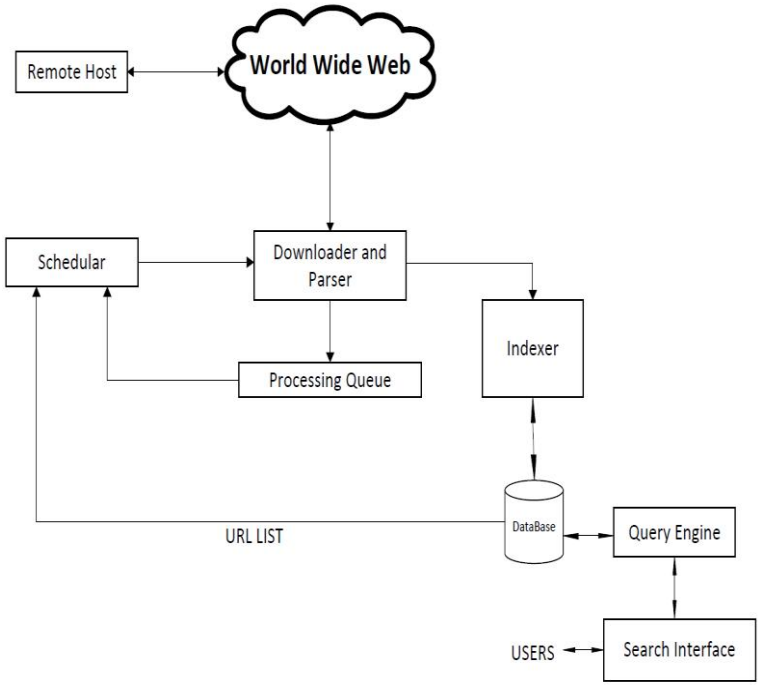

Fig. 1: General Architecture of Web Crawler

Web crawlers recursively traverse and download web pages (Using GET and POST commands) for search engines to create and maintain the web indices. The need for maintaining the up-to-date pages causes a crawler to revisit the websites again and again. In general, it starts with a list of URLs to visit, called the seed URLs. As the Crawler traverses these URLs, it identifies all hyperlinks in the page and adds them to the list of URLs to be visited, called the crawl frontier. URLs from the crawl frontier are visited one by one and searching of the input pattern is done whenever text content is extracted from the page source of the web page.

The basic working of a web-crawler can be summarised as follows:

1) Select a starting seed URL or URLs
2) Add it to the Processing queue
3) Now pick the URL from the Processing queue
4) Fetch the web-page corresponding to that URL
5) Parse that web-page to find new URL links
6) Add all the newly found URLs into the Processing queue

Go to step (2) and repeat while the Processing queue is not empty

IV. PROPOSED TAXONOMY OF WEB CRAWLER

Proposed Taxonomy classifies the Web Crawlers on basis of following parameters:

A. Coverage Area
B. Mobility
C. Domain Specific
D. Load Distribution

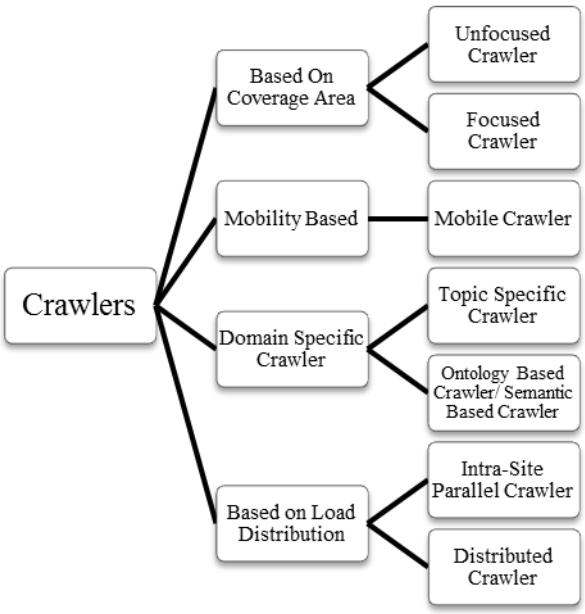

Fig. 2: Proposed Taxonomy of Web Crawler

A. Based on Coverage Area

Depending upon the Web area covered by Crawlers, there are two different classes of crawlers known as (1) Unfocused and (2) Focused.

1) Unfocused Crawler

The purpose of Unfocused Crawlers is to search over the entire Web to construct their index. As a result, they deal with the laborious job of creating, refreshing, and maintaining a database of large dimensions.

2) Focused Crawler

Focused Crawler limits its function upon a semantic Web zone by selectively retrieving pages to predefined topic and avoiding irrelevant web regions to eliminate irrelevant data.

B. Mobility Based Crawler

In order to filter out the irrelevant data at the source site, where the data resides, Crawlers are transported to the site of the source. Different classes of mobility based crawler are

1) Mobile Crawler

Mobile Crawler crawl the Web using Mobile Agents. A mobile agent is an automatic-independent program act on behalf of its owner. Mobile Crawlers are transported to the remote site where they filter out any unwanted data locally before transferring it back to the search engine. These migrating crawlers remained in the remote systems and perform constant monitoring of all the web documents assigned to them for changes.

Mobile Crawler reduces the network load caused by the traditional crawlers by compressing the retrieved data at remote host and reducing the amount of data transferred over the network.

C. Domain Specific Crawler

This type of crawlers traverses the web according to specific domains. Major classes of Domain Specific Crawler are:

1) Topic Specific Crawler

A Topic specific Crawler is a program used for searching information related to some specific topic from the web. The main property of topic specific Crawling is crawler does not need to collect all web pages, but selects and retrieves only relevant pages. It starts with a topic vector, and for each URL, the relevance is computed for the web page in the selected domain. If it is found to be important, it gets added to the URL list else, gets discarded.

2) Ontologies Based Crawler / Semantic Based Crawler

These crawlers make use of semantics, which helps to download only relevant pages. Semantics are provided by Ontologies. Ontologies provides a common vocabulary of an area, defines meaning of terms and relationships between them. Ontology based crawler crawl the web focusing on pages relevant to given ontologies.

D. Based on Load Distribution

In order to increase the coverage and decreases the bandwidth usage, crawlers distribute and localize the load. Depending upon load distribution two major classes of Crawler are Intra – site Parallel Crawler and Distributed Crawler.

1) Intra – site Parallel Crawler

In intra site Parallel Crawler all crawling processes run on the same local network and communicate through a high speed interconnection.

2) Distributed Crawler

Distributed Crawler is a crawler when crawling processes run geographically distant location connected by the internet.

The distributed crawler involves several crawlers crawl information simultaneously on different computers.

V. ISSUES IN WEB CRAWLER

Till date the Crawlers that have been designed yet, involves following challenges. The premises underlying the crawler are the following:

1) Crawling processes needs to communicate to each other, in order to improve the quality of the downloaded pages. This communication increases as the number of crawling processes increases, resulting in communication overhead.
2) At the same time of maximizing the coverage rate of web resources, Crawlers also download a large amount of useless or redundant information, thus quality gets affected.
3) It has to have a good crawling strategy, i.e., a strategy for deciding which page to download next.
4) It needs to have a highly optimized system architecture that can download large number of pages per second while being robust against crashes.
5) Due to large coverage area a crawler rarely refresh its crawls.
6) A large amount of web pages are written in JavaScript or Ajax. It's impossible to extract new URLs by tag matching because these link URLs are generated by JavaScript functions.
7) A crawler does not have good ranking functions or support advanced query capabilities that need more processing power.
8) The crawler also has to decide how frequently to revisit pages it has already seen, in order to keep its crawler informed of changes on the Web.
9) When multiple processes run in parallel to download pages, different processes download the same page multiple times. It may be the case that one process is not aware that the other process has already downloaded the page. Such multiple downloads should be minimized to save network bandwidth and increase the crawler's effectiveness.
10) Security: The issue which needs to be addressed is security. Migration the crawler and executing the code at remote site sometime causes severe security problems because a mobile crawler might contain harmful code.

Above mentioned issue (1), (4), (5) are addressed by Mobility based Mobile crawlers. Focused Crawler addresses the issues (2) & (3). Ontology based domain specific crawler addresses the problem associated with (g). Some issues still need to be addressed for effective crawling like security.

VI. CONCLUSION

A web crawler is a way for the search engines and other users to regularly ensure that their databases are up to date. Web crawlers are a central part of search engines. This paper has described the evolution of web crawlers
The proposed taxonomy of Web crawlers, classifies the existing crawlers, based on four parameters: coverage area, mobility, topic- domain and load distribution. Furthermore, this paper also discussed the issues addressed by Crawler.

REFERENCES

[1] http://en.wikipedia.org/wiki/Web_crawler.
[2] Anbukodi S and Muthu Manickam K., *Reducing Web Crawler Overhead using Mobile Crawler*, In the Proceedings of ICETECT 2011, pp. 926-932.
[3] Gray M., *Internet Growth and Statistics: Credits and background*, http://www.mit.edu/people/mkgray/net/background.html.
[4] Marios D. Dikaiakos, Athena Stassopoulou and Loizos Papageorgiou, *An investigation of web crawler behaviour: characterization and metrics*, Science Direct Computer Communications, 2005, pp. 880–897.
[5] Qingzhao Tan and Prasenjit Mitra, *Clustering-Based Incremental Web Crawling*, ACM Transactions on Information Systems, Vol. 28, No. 4, Article 17, November 2010.
[6] F. Douglis, A. Feldmann, B. Krishnamurthy and J. Mogul, *Rate of change and other metrics: a live study of the World Wide Web*, In Proceedings of the USENIX Symposium on Internet Technologies and Systems on USENIX Symposium on Internet Technologies and Systems (USITS). USENIX Association, Berkeley, CA, 1997, pp. 147–158.
[7] J.L. Wolf, M.S. Squillante, P. S Yu, J. Sethuraman, and L. Ozsen, *Optimal crawling strategies for Web search engines*, In Proceedings of the 11th International Conference on World Wide Web (WWW). ACM, New York, NY, USA, 2002, pp. 136–147.
[8] J Cho and H. Garcia-Molina, *Synchronizing a database to improve freshness* SIGMOD Record 29, 2, 2000, pp. 117–128.
[9] Chakrabarti, Soumen, Martin van den Berg, and Byron Dom., *Focused crawling: a new approach to topic-specific Web resource discovery*, Elsevier, 1999.
[10] Rajender Nath and Satinder Bal, *A Novel Mobile Crawler System Based on Filtering off Non-Modified Pages for Reducing Load on the Network*, published in The International Arab Journal of Information Technology, Vol. 8, No. 3, July 2011.
[11] Rajender Nath and Satinder Bal, *Reduction in Bandwidth Usage for Crawling Using Mobile Crawlers*, published in International Journal of Computer Science and Knowledge Engineering, January-December 2007, pp. 51-61, ISSN: 0973-6735.

[12] S. Brin and L. Page, *The Anatomy of a Large-Scale Hyper Textual Web Search Engine*, In Proc. of the 7th World Wide Web Conference, 1998, pp. 107-117.

[13] D. Gomes and M.J. Silva, *The Viuva Negra Crawler*, Software, Practice and Experience, 2008, Volume 38, No. 2.

[14] J. Cho and H. Garcia Molina, *Parallel Crawlers*, In Proc. Of the 11th International Conference on World Wide Web, ACM, Hawaii, 2002, pp. 124–135.

[15] P. Boldi, B. Codenotti, M. Santini and S. Vigna, *UbiCrawler: a Scalable Fully Distributed Web Crawler*, Software, Practice and Experience, Vol. 34, No. 8, 2004, pp. 711-726.

[16] D. Daly, D. Deavours, J. Doyle, A. Stillman and P.Webster, *Mobius: An Extensible Framework for Performance and Dependability Modelling*, Mobius, 1999.

[17] Odysseas Papapetrou and George Samaras, *Distributed Location Aware Web Crawling*, 2004, ACM, New York, USA.

[18] Mitra Nasri, Saeed Shariati and Mohsen Sharifi, *Availability and Accuracy of Distributed Web Crawlers: A Model-Based Evaluation*, Second UKSIM European Symposium on Computer Modeling and Simulation, IEEE, 2008.

[19] Mukesh Kumar and Renu Vig, *Learnable Focused Meta Crawling Through Web*, In 2nd International Conference on Communication, Computing & Security ICCCS 2012.

[20] Bing Zhou Bo Xiao, Zhiqing Lin and Chuang Zhang, *A Distributed Vertical Crawler Using Crawling-Period Based Strategy*, IEEE, 2010.

[21] Mitra Nasri, Saeed Shariati and Mohsen Sharifi, *Availability and Accuracy of Distributed Web Crawlers: A Model-Based Evaluation*, Second UKSIM European Symposium on Computer Modeling and Simulation, IEEE, 2008.

[22] R. Baeza-Yates, C. Castillo, F. Junqueira, V. Plachouras and F. Silvestri, *Challenges on Distributed Web Retrieval*, IEEE 23rd International Conference on Data Engineering ICDE, 2007.

