Web Page Load Reduction using Page Change Detection

Naresh Kumar*
Assistant Professor, MSIT, Janakpuri,
New Delhi, India.

***Abstract:*-World Wide Web is the assortment of hyper-linked documents in hypertext mark-up language format. In the wake of the growing and dynamic nature of the internet, it has become a challenge to traverse all the URLs offered within the web documents. The list of URLs is extremely immense and so it's troublesome to refresh it quickly as 40% of web pages undergo changes every day. Due to this, a lot of the network resources, particularly bandwidth is consumed by the web crawlers to keep the repository updated at the search engine end. Therefore, this paper proposes Percentage-based Web-page Modified Crawler for focused load reduction on the network that looks up for web pages and retrieves them from the webs that are associated with a particular domain solely and skips unsuitable domains. It ignores the minor changes in update of the previous web document and focuses only on such network resources which have at least 30% difference from the earlier existing document. It downloads solely those pages that are modified more than a fixed percentage of change and avoid the pages which don't seem to be changed at all or have undergone only few changes from the last crawl. It has been found that the crawler which has been proposed can lower down the load significantly on the network. The novelty in achieving the load reduction is going to be a major contribution in the working of web crawlers.**

Keywords: URLs, web documents, bandwidth, repository, search engine, domain, network resources, web crawlers

1. INTRODUCTION

World Wide Web is the system of assorted inter-linked hypertext documents which are expanding rapidly from a few thousand pages in 1993 to more than a billion of pages at present [2]. In the present era of cloud computing and convergence of technologies, it has become all the more important for WWW to perform its functions in a more effective and efficient manner. Networks are becoming complicated day after day and scientists are increasingly using a search engine to locate research of interest: some rarely used libraries, locating research articles primarily on-line; scientific editor using a search engine to locate potential reviewers [1].

The statistics [1] are given in Table 1 for the search engine coverage with respect to estimated web-size for top 10 engines.

S.No.	Seach Engine	Coverage (in %)
1.	Northern Light	16%
2.	Snap	15.5%
3.	Alta Vista	15.5%
4.	HotBot	11.3%
5.	Microsoft	8.5%
6.	Infoseek	8%
7.	Google	7.8%
8.	Yahoo	7.4%
9.	Excite	5.6%
10.	Lycos	2.5%

Table 1: Search Engine Coverage

This implies that none of the search engines can cover more than 16% of the entire web. Many of the above search engines are running multiple processes in parallel, which are referred as parallel crawler [4]. It is also interesting to note that only 6% of the web servers have scientific/ educational content (defined here as a university, college and research lab servers) [1]. In yet another study [5], its indicated that 40% of the content available on the web are dynamically changing. But since many of the search engines like Google make extensive use of 'popularity' based accessibility, the trend of popular pages become more popular and new, unlike pages becoming hardly visible despite having its content modified by more than 30%, is an unfair technique which biases the accessibility of information on the web.

Therefore, due to the presence of a large number of pages on the web, the search engines have to depend on crawlers to retrieve relevant pages from the cluster of several billions of web pages. Whereas, a crawler (also called wanderer, spider, walker, etc.) downloads and stores the web pages using the hyperlink of the document; a mobile crawler, on the other hand, fetches webpage and download only those pages that are changed since the last crawl [3]. Currently, web crawlers in vogue have indexed billions of web pages and around 40% of internet traffic is due to them. Besides that, the utilization of bandwidth is also due to web crawlers which download the relevant web pages for various search engines [6]. Various suggestions [4][6][8][10]were given to use mobile crawlers in order to keep the repositories updated. In a study, it has been found out that not most of the web pages undergo considerable changes [8]. Therefore, this paper proposes an alternate approach by using mobile crawlers, frequency and amount of change. These crawlers go the remote site and examine the web pages. It ignores those pages that are not modified or if the modifications are under a certain level. Therefore,the crawlers send solely the considerably modified or never crawled web pages to the search engine for indexing.

The rest of the paper is organized as follows: Section 2 discusses the related work done on the topic. Section 3 describes the limitations of the current crawling techniques. Section 4 describes the architecture of the proposed mobile crawler system. Section 5 outlines the URL devolution approach and Section 6 delineate the working of the proposed crawler system. Section 7 concludes the paper. Section 8 describes the future work required to enhance the proposed approach.

2. RELATED LITERATURE

In [6], last modified date, ASCII count, number of keywords were considered in order to check the change a web page has undergone. Their purpose was to determine the webpage without downloading it at the search engine end, preserve the bandwidth and reduce load on the network. It was found that the proposed scheme reduced the load to half without compression and reduced much more if the compression is done. It therefore preserved the network bandwidth. In order to measure web page detection, ASCII count was taken into consideration. The current ASCII count of the web page is calculated and then it is compared to the previous ASCII count. Only the web pages with change in ASCII counts were accepted. Also, the web pages whose last modified date was not found were directly downloaded without ASCII value comparison process. Experimentally, it was found that the average load on the network decreased from 800kbps to 480kbps. Furthermore, this load was found to be approximately 170kbps when the crawler sent the pages to the search engine after using the compression tools. In [11], a number of web sites which list products for sale was examined to see the nature of web page changes through a study of the evolution of some 12,000 web pages from around 20 websites for five months. It focused in categorizing and measuring the quantity of structural changes that web pages undergo. It was found that the structure of web pages tends to change regularly with occasional drastic changes [11]. Focused Structure Parallel Crawler by [9] proposed the application of link unconstrained web page importance metric in order to control the priority rule in the crawl frontier by a modest weighted architecture for structured focused parallel web crawler in which clickstream-based prioritizing is used to assign credit to URL in crawl frontier.

The focused crawlers recorded by [1][4][6][7] have the following limitations:

- The appropriateness of a web page can only be examined after downloading it to the search engine.
- The downloading of the web pages' results in increased bandwidth consumption and increased load on network.
- In appropriate web pages are downloaded and bandwidth and time is spent in processing them.

3. PROBLEM FORMULATION

Many techniques[3][8][9][10]of mobile crawling were considered in order to overcome the provided limitations in the previous section.

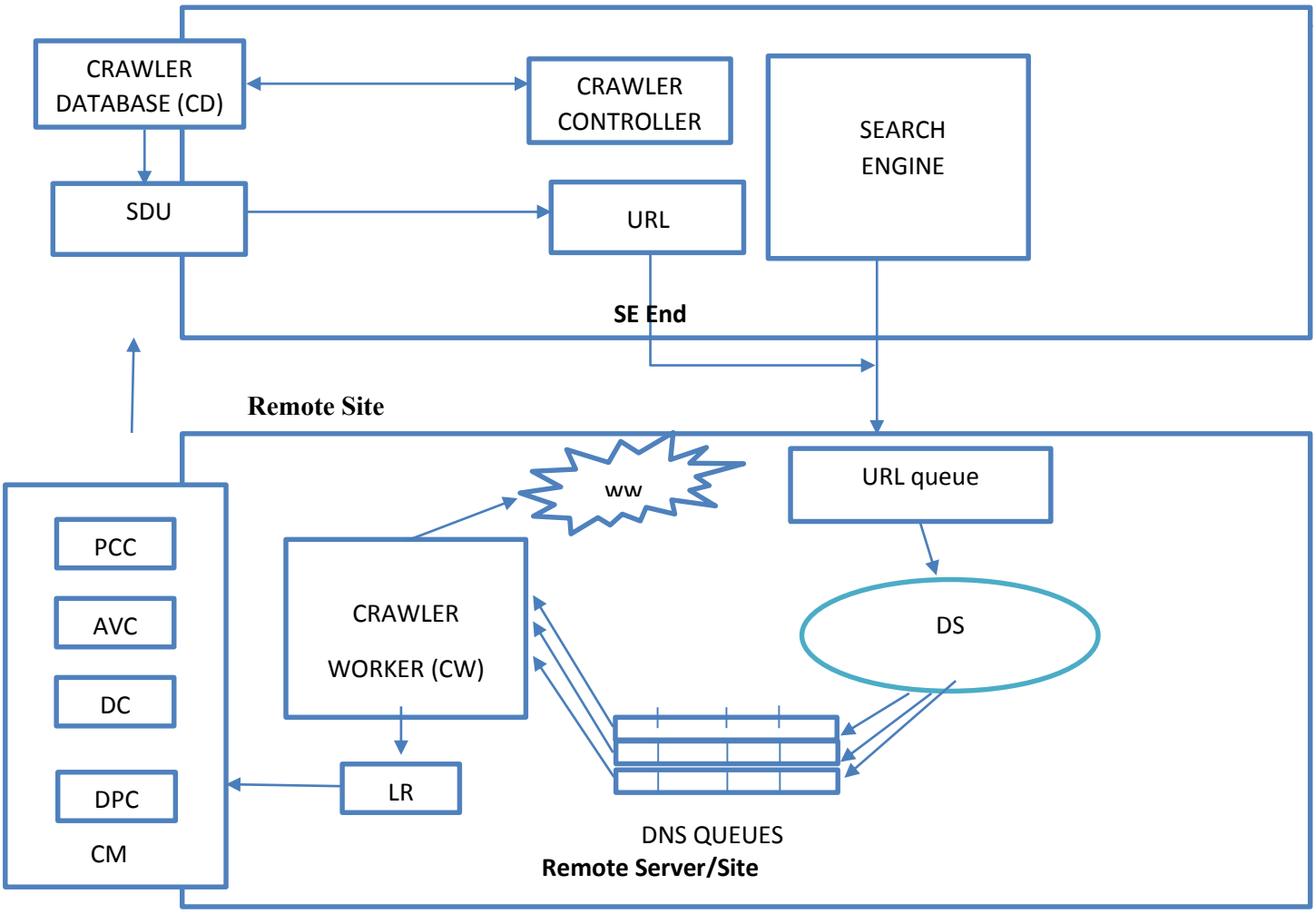
But, these techniques have a few limitations as well:

- Residency of mobile crawlers on remote site consume a lot of memory space which may lead to availability of lower memory since remote sites also need memory.
- There is a possibility that these remote sites do not allow these mobile agents to stay at the remote site due to security measures.
- The requirement for bandwidth reduction is still present in order to obtain an efficient system which can make the process smoother.
- Bandwidth is wasted in processing web pages that have undergone only few changes which does not affect the researcher at the cost of considering all the pages related to the search.

To overcome these limitations, this paper proposes PWCC that uses the concept of filtering web pages in context to the amount of change they have gone through since the last crawl.

4. PROPOSED ARCHITECTURE

In Percentage Web-page Change Crawler (PWCC), the mobile crawlers reach the remote sites from the search engine where they download and process the web pages. In the filtering procedure, these mobile crawlers ignore the unmodified webpages and the webpages that have undergone only minor changes as compared to the previous web document and focuses only on those webpages that have undergone at least 30% change from the earlier existing document. Since some changes and adjustments minor changes in the layout, date and time changes, logging functionality, which don't affect the document much and downloads only such webpages which have undergone a considerable change since the last crawl or have been crawled for the first time. The propose crawler is shown in Fig. 1. The main components of the proposed crawlers are CC, DS, VC, CD, PA, LR, CW, PP and CU.



4.1 Similarmodules:

The following modules bear a resemblance to previously stated modules in [6] and [9].

a) **Crawler Controller (CC):** Working of this module is similar to that of 'Crawler Manager' in [6]. Crawler Controller also delegates the URL to the mobile crawlers with the help of the CM Module.

b) **Domain Selector (DS):** Domain Selector module is similar to "DND" module in [6].The DS is responsible for selecting particular domain for URLs coming from the URL queue and forwarding it to the specified DNS queue.

c) **Version Checker(VC):** The working of VC is similar to the module 'FCE' which is explained in [6].

d) **Crawler Database (CD):**Crawler Database (CD) is similar to 'SD' in [6]. This module contains the given fields:

i. **URL Name:**It is similar to the 'Name of URL' in [6].

ii. **Main URL:** It is similar tothe 'Parent URL' in [6]. For example: say the USER entered https://www.accenture.com, then this URL is called the Main URL whereas https://www.accenture.com/in-en/new-applied-now is the URL Name.

iii. **ASCII value:** It is similar to the 'ASCII count' in [6].

iv. **Last Change Date:** It is similar to 'Last Modified Date' in [6].

v. **Frequency Change:** It is similar to 'Frequency of change' in [6].

vi. **File Locations:**It is similar to 'File Path' in [6].

vii. **Percentage Change:** It refers to the amount of change in percentage undergone by the web page.

e) Crawler Controller: The Crawler Controller (CC) resembles 'ODF' in [6]. The Statistical Database module is sent along the

f) respective mobile crawler on the remote server with 'URL Name' and 'ASCII value'.Therefore, the comparator unit checks the provided parameters and only if the HTML page passes all the parameters, it sends the given web page for further processing.

g) Page Analyser (PA): The working of PA is similar to that of 'AM' in [6]. This module sends the value of the 'ASCII value' with the spiders. A replica of this 'ASCII value' is kept by the remote site for future help to these agents.

h) Link Releaser (LR): Link Releaser is responsible for taking out links from the HTML pages coming the search engine from the crawler worker. It releases all the links to the comparator unit by taking them out from the HTML pages.

i) Crawler Worker (CW): The Crawler Worker is similar to 'CH' in [6]. The page is only processed if the 'ASCII value' of a given web page is different from its previous 'ASCII value'.

j) Page Provider (PP): Page provider fetches the web page with the URL of the crawling handler. This web page is then sent to the controller for further processing. Similarly, a new web page is taken with the help of a new URL. Also, the web crawler has the ability to run multiple processes at a single time by sharing the load between different processes. Each crawl worker (CW) is independent i.e. each crawl does not have to depend on other as the crawl worker gets seed URL from an independent DNS queue.

4.2 Comparator Unit (CU): The comparator unit is responsible for performing several tasks. This unit works on both sides i.e. search engine side and remote site. The several actions which the comparator unit performs are listed below:

a) *Download Period Checking(DPC):* This module examines the downloading period of a web page. If it has attained the downloading period, then the URL of the given web page is sent to the Page Analyser(PA). Whereas, if it has not attained the downloading period, the URL of the web page is rejected.

b) *Date Checking(DC)*: Along with this comparison, the 'Last Change Date' and 'Frequency Change' is checked to see if the difference of the existing date and the 'Last Change Date' is less than or equal to the value of 'Frequency Change'. If the difference is less than or equal to the value of 'Frequency Change', the web page is not processed. Whereas, if the difference between the two is found to be exceeding 'Frequency Change', the HTML page is processed.

c) *ASCII Value Checking(ACC)*: The page is only processed if the 'ASCII value' of a given web page is different from its previous 'ASCII value'.

d) *Percentage Change Checking(PCC)*: Percentage change checking is explained in Fig. 2. It is done at the remote site for the web pages which have been crawled earlier. In this the amount of change a web page has undergone is calculated in percentage. If the amount of change is found to be equal or less than 30%, the web page is not processed whereas if the percentage change is found to be greater than 30%, the web page is processed.

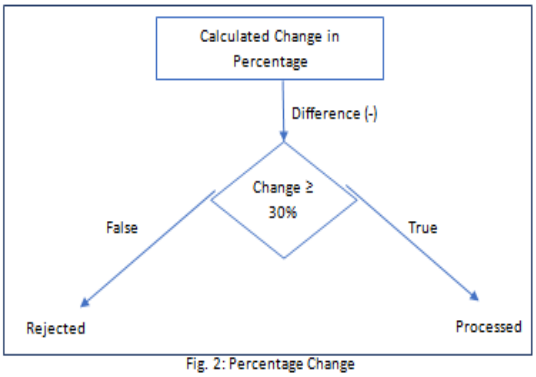


Fig. 2: Percentage Change

5.URL Devolution Approach: The URL Devolution Approach is same as 'The URL Delegation Approach' used [9].

6.WORKING OF THE PWCC

The working of the PWCC is explained in Fig. 3. The mobile crawlers used in PWCC takes the SDU, the CU and the PA with it on the remote site where the web pages are to be crawled. The mobile crawler accesses the web page one after another whose URLs are provided in the SDU and calculates the ASCII value. It then compares the current ASCII value with the old ASCII value. The page is only processed if the ASCII value of a given web page is different from its previous 'ASCII value'. Then, it checks for the percentage change in a web page. If the amount of change is found to be equal or less than 30%, the web page is not processed whereas if the percentage change is found to be greater than 30%, the web page is processed. But, the pages which were never crawled before are directly sent

to the search engine without any comparison of ASCII value and checking of the percentage change.

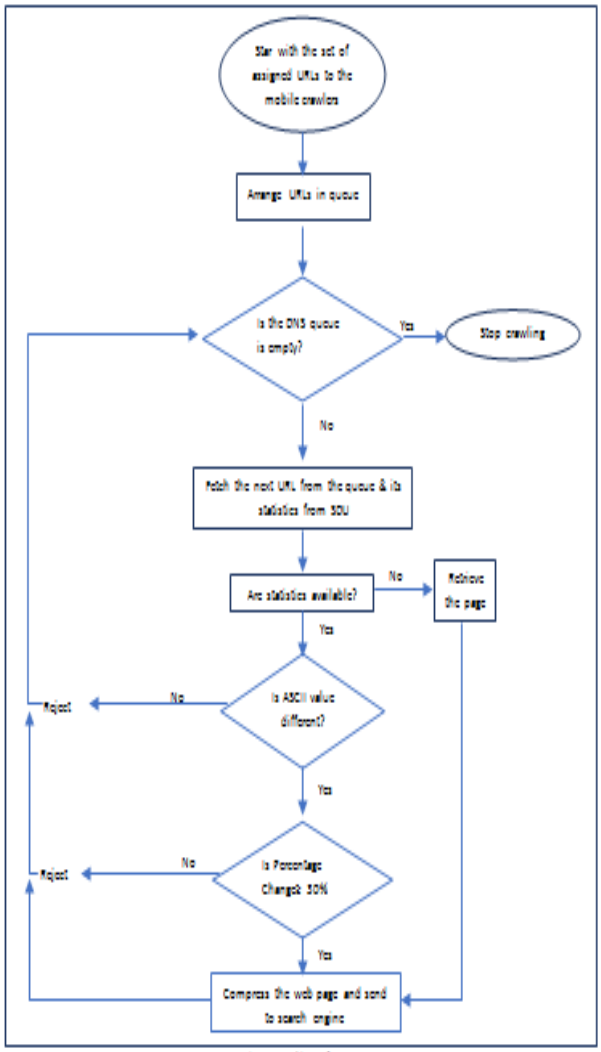


Fig. 3: Working of PWCC

7. CONCLUSION AND FUTURE WORK

The study of literature from 1999 was performed and the limitations of the techniques proposed were identified. Therefore, the solution to these problems has been provided by using a different technique in order to overcome the limitation. The benefits of using the proposed technique are as follows:

a) It lowers down bandwidth consumption which is very significant for an efficient system.
b) It saves the bandwidth in processing the pages which were not supposed to be crawled before.
c) It doesn't waste bandwidth on pages which have undergone only few changes.

For the proposed model to be successful, execution is very essential. Therefore, work is in progress to justify the proposed model with the help of real world implementation.

REFERENCES

[1] Steve Lawrence and C. Lee Giles, "Accessibility of information on the web", in Commentary, Nature, Vol. 400, pp. 107-107, 8 July,1999
[2] Shkapenyuk V. and Suel T., "Design and Implementation of A High Performance Distributed Web Crawler", in Proceedings of the 18th International Conference on Data Engineering, San Jose, California. IEEE CS Press, pp. 357-368, 2002.
[3] Jan Fiedler, and Joachim Hammer, "Using the Web Efficiently: Mobile Crawlers", in Seventeenth Annual International Conference of the Association of Management (AoM/IAoM) on Computer Science, Maximilian Press Publishers, San Diego, CA, pp. 324-329, August 1999.
[4] Nidhi Tyagi & Deepti Gupta," A Novel Architecture for Domain Specific Parallel Crawler ", in Nidhi Tyagi et. al. / Indian Journal of Computer Science and Engineering Vol 1 issue 1, pp. 44-53, 2010.
[5] Cho J. and Garcia-Molina H., "Estimating Frequency of Change", in ACM Transactions on Internet Technology (TOIT), vol. 3, no. 3, pp. 256-290, 2003.
[6] Rajender Nath and Naresh Kumar, "A Novel Parallel Domain Focused Crawler for Reduction in Load on the Network" in International Journal of Computational Engineering Research Vol. 2 issue 7, pp. 77-84, November,2012.
[7] Joach im Hammer, Jan Fiedler, "Using Mobile Crawlers to Search the Web Efficiently", in International Journal of Computer and Information Science, 1:1, pp. 36-58, 2000.
[8] Odysseas Papapetrou and George Samaras," Minimizing the Network Distance in Distributed Web Crawling", R. Meersman, Z. Tari (Eds.): CoopIS/DOA/ODBASE 2004, LNCS 3290, pp. 581–596, 2004.
[9] F. Ahmadi - Abkenari, Ali Selamat, "A Click stream-based Focused Trend Parallel Web Crawler", in International Journal of Computer Applications (0975 – 8887), Volume 9– No.5, November 2010.
[10] Rajender Nath and Satinder Bal," A Novel Mobile Crawler System Based on Filtering off Non-Modified Pages for Reducing Load on the Network ", in The International Arab Journal of Information Technology, Vol. 8, No. 3, July 2011
[11] Mira Dontcheva, Steven M. Drucker, David Salesin and Michael F. Cohen, "Changes in Webpage Structure over Time" UW CSE Technical Report 2007-04-02