A Users Search History based Approach to Manage Revisit Frequency of an Incremental Crawler

Yadu Nagar
Assistant Professor
IIMT Engineering College, Meerut, India

Niraj Singhal
Assistant Professor
Shobhit University, Meerut, India

ABSTRACT

With the tremendous growth of the Internet, World Wide Web has become a huge source of hyperlinked information contained in hypertext documents. Search engines use web crawlers to collect these documents from web for the purpose of storage and indexing. An incremental crawler visits the web for updating its collection. There is a need to regulate the frequency of the crawler to visit web sites and provide latest information to the user. In this paper a novel approach to manage the revisiting frequency of an incremental crawler based on the users search history is being proposed.

Keywords

Search engine, incremental crawler, page revisit frequency, hit count, user's search history.

1. INTRODUCTION

Internet [12] is a global system of interconnected computer networks that use the standard internet Protocol (IP) suite to serve billions of computers worldwide. The World Wide Web (or Web) [1, 2, 5, 8] is a large repository of text, documents, images, multimedia and vast amount of other information. Web is a huge source of interlinked hypertext documents [13] lying on different websites. Hyperlinks allow the user to connect to the interconnected links in back and forth manner. Because the web contains very large number of web pages, search engine depends upon crawlers for gathering all the required web pages on the web.

A crawler [6, 7] is a program that automatically retrieves and stores web pages and it creates the local collection of web pages. A crawler starts by taking initial set of URLs into the queue, where all the retrieved URLs are kept and also prioritized. The crawler takes URL form this queue, downloads the web pages, extracts all the URLs present in the downloaded page, and place the new URLs in the queue. Crawler repeats this process until it crawls a desirable no of web pages. These collected web pages are later used by search engine to answer the user query. An incremental crawler [6] updates the web pages of their local collection in incremental manner. It does not periodically refresh the collection, but improves the "newness" of the local collection and fetches new pages in the local collection in more appropriate manner.

Providing relevant information to the users and fulfill their needs is the primary goal of search engines, therefore finding the right content from Web considering the user's interests and needs have become increasingly important. When a user makes a query from search engine it generally returns a large number of pages in response to user queries. This result-list contains many relevant and irrelevant pages according to user's query. As user impose more number of relevant pages in the search result-list. To assist the users to navigate in the result list, various ranking methods are applied on the search results. In this paper a new methodology has been proposed that manages the revisiting frequency of an incremental crawler based on the user's search history.

2. RELATED WORK

A search engine [2, 4] is a coordinated set of programs that is able to read every searchable page on the web creates an index of the information it finds, compare this information with a user's search request and finally return results to the user. This technology grants quick and easy access to the knowledge they seek, by categorizing web pages according to their relevancy in regard with user's request. Based on the application for which search engines are needed [11] they can be categorized as follows:-

- Primary search engines scan entire sections of the www and produce their results from databases of web page content, automatically created by computers.
- Subject guides like indexes in the back of a book cover fewer resources and topics but provide more focus and guidance.
- Finance-oriented search engines facilitate searches for specific information about companies.
- Image search engines help us search the www for images of all kinds.
- Business and Services search engines essentially National yellow page directories.
- Job search engines either provides potential employers access to resumes of people interested in working for them or provide prospective employees with information on job availability.
- News search engines search newspaper and news website archives for selected information.
- Specialized search engines search specialized databases, allow users to enter search terms in a particularly easy way.
- People search engines search for names, addresses, telephone numbers, and e-mail addresses.

Search engines (figure 1) operate as a link between users and web documents. Without search engines, this vast source of information in web pages remain hidden. It is searchable database which collects information from web pages, index it and stores results in a huge database from where it can be searched quickly. A general search engine has three parts crawler, indexer and query engine. A web crawler is a module that searches the web pages from the web world, on the search engine's behalf and follows links to reach other linked pages. URL queue list is used to add new Urls extracted from the downloaded pages and it also feeds Urls to crawler for downloading. The indexer indexes the uncommon words

from each page and records the URL where each word has occurred. The result is stored in a large table containing URLs, priority to pages in a repository where a given word has occurred. Query engine is responsible for receiving and filling search requests from users. It relies on the index and on the repository.

Crawl Module crawls a page and saves/updates the page in the Local Collection based on the request from Update Module. The Crawl Module also extracts all links/ URLs on the crawled page and adds in ALL_URLs.

While designing the incremental crawler two issues to be addressed are [6] maintain the local collection with fresh

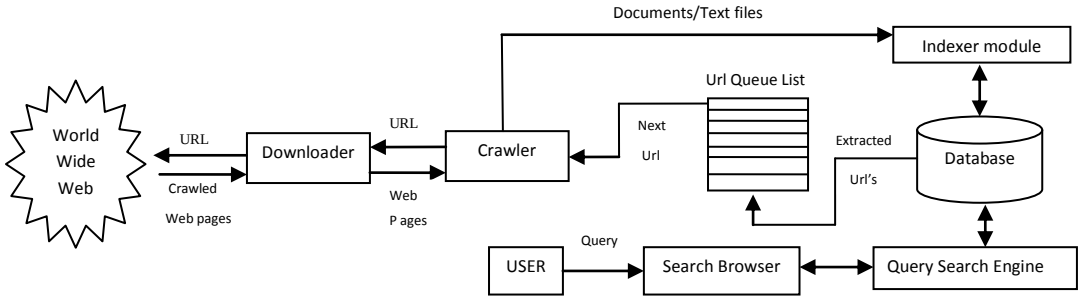

Figure 1. Architecture of a typical web search engine

Various types of crawlers used in search engines are parallel crawler, distributed crawler, focused crawler, hidden crawler, and incremental crawler. An incremental crawler [6] updates the web pages of its local collection in incrementally manner. It improves the newness of the local collection in more appropriate manner. The incremental crawler refreshes existing pages and replaces less important existing pages with more important new pages. It crawls the web sites continuously, refreshes local collection, and provides fresh information to the user. For good freshness the crawler needs to revisit and select the page that will increase the freshness most significantly.

pages and improve quality by keeping relevant pages in the local collection. To maintain the freshness of local collection, Revisit Frequency Calculator is used to find the appropriate revisit frequency of the crawling so that crawler can update its local collection with fresh documents. The web crawler should be improving the quality of the local collection by replacing less relevant pages with more relevant pages. It is essential because pages are continuously created and removed, and it may possible that some of the pages that were created may be more relevant than existing pages in the local collection. That's why the crawler needs to replace less relevant existing pages with more relevant new web pages.

Having a look to the literature on revisit frequency of crawler

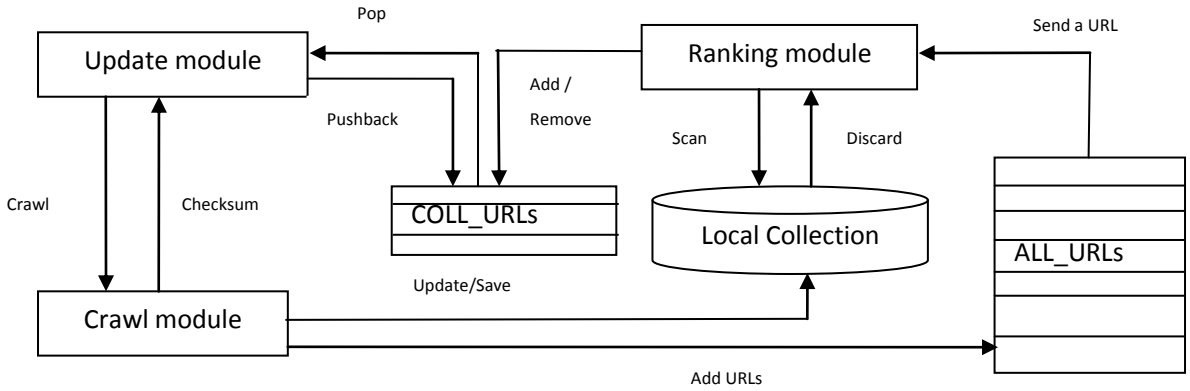

Figure 2. Architecture of an Incremental web crawler

The incremental crawler consists of three main data structures *ALL_URLs* (set of all URLs accessed/to be accessed), *COLL_URLs* (set of all URLs in the Local Collection) and *Local Collection* (collection of downloaded pages related to URLs in COLL_URLs) and three main software modules *Ranking Module*, *Update Module* and *Crawl Module*. The Ranking Module continuously scans through ALL_URLs and the Local Collection to take the right refinement decision. Update Module selects a URL from COLL_URLs, downloads the page and if changed updates the Local Collection. The

it is found that, Brin and Page [8] developed PageRank algorithm at Stanford University based on the hyper link structure. PageRank algorithm is used by the famous search engine, Google. PageRank algorithm is the most frequently used algorithm for ranking billions of web pages. The PageRank algorithm is based on the concepts that if a page surrounds important links towards it then the links of this page near the other page are also to be believed as imperative pages. The Page Rank imitate on the back link in deciding the rank score.

Another approach [1] proposed a novel mechanism and a novel architecture for incremental crawler to regulate the revisiting frequency and focused on the documents that contain dynamic information which gets changed on daily, weekly, monthly or yearly basis. It needs to refresh the search engine side storage so that latest information is made available to the user. Singhal et al [13] proposes an alternate approach for optimizing the frequency of migrants for visiting web sites based on user's interest. It adjusts the revisit frequency by dynamically assigning a priority of revisiting to a site by computing the priority based on previous experience that how many times the crawler founds changes in content in 'n' visits and the interest of the users shown in the websites.

Dixit et al [12] proposed an efficient approach for optimizing the frequency of visits to sites. It adjusts the frequency of visit by dynamically assigning a priority to a site. In approach [14] personalization is used with which web access or the contents of a Web page are modified to better fit the desires of the user. This may involve actually creating Web pages that are unique per user or using the desires of a user to determine what Web documents to retrieve. In it, a technique which rank the web pages based on the user's interest based on TF-IDF measure is used. In next section, an alternate novel approach to manage the revisit frequency of an incremental crawler based on the user's search history is being proposed.

3. PROPOSED WORK

Since web is extremely large in size and the web documents changes dynamically, calculating the importance of web pages and change of frequency become very difficult. Relevant web pages represent certain importance value of an individual page on the web which is a key factor for web search. Various search engines components such as the crawler, indexer, and ranker are generally guided by this measure. Page Rank is an excellent way to prioritize the results of web keyword searches and provide relevant results. A Crawler is a program that automatically retrieves and stores web pages but it adds excessive traffic to the already congested Internet. So the frequency of visits to web sites should be optimized to reduce this traffic by calculating Hits or visits to particular website.

The websites for which users show more interest should be crawled at a faster rate as compared to those that are less or rarely surfed by the users. The proposed architecture adjusts the revisit frequency of websites by considering the interest shown by users. It also helps in maintaining the freshness of the repository and provides relevant web pages to the users. The calculation of the number of user visits or Hits Count helps in finding the importance of the web pages which is efficiently managed by the Rank Updater so that those particular pages are ranked high. Moreover the architecture is suitable for the applications where search is according to the interest of the user.

In proposed architecture of incremental crawler (see figure 3) the user sends query to the search engine and it sends queried Url/hyperlink information to the database. When a web page is accessed a script will load on the client site from web server. This script will monitor or count the clicks as well as keyboard event to occur on the web page accessed by user. When an event occurs and if that event will happen over hyperlink then it will send a message to log file database with information of current web page and hyperlink.

On server side log file database records the webpage-id, hyperlinks of that page and hit count of hyperlinks. Hit count will incremented every time a hit occur on hyperlink. Now Rank Updater module is used to increases the rank of those Urls which reaches the threshold value of hit count. The database or log files are accessed by crawler at the time of crawling the web. This Hit count information will be stored in log file database and this information is used to calculate the Rank of different web pages and High rank information is sent to web server.

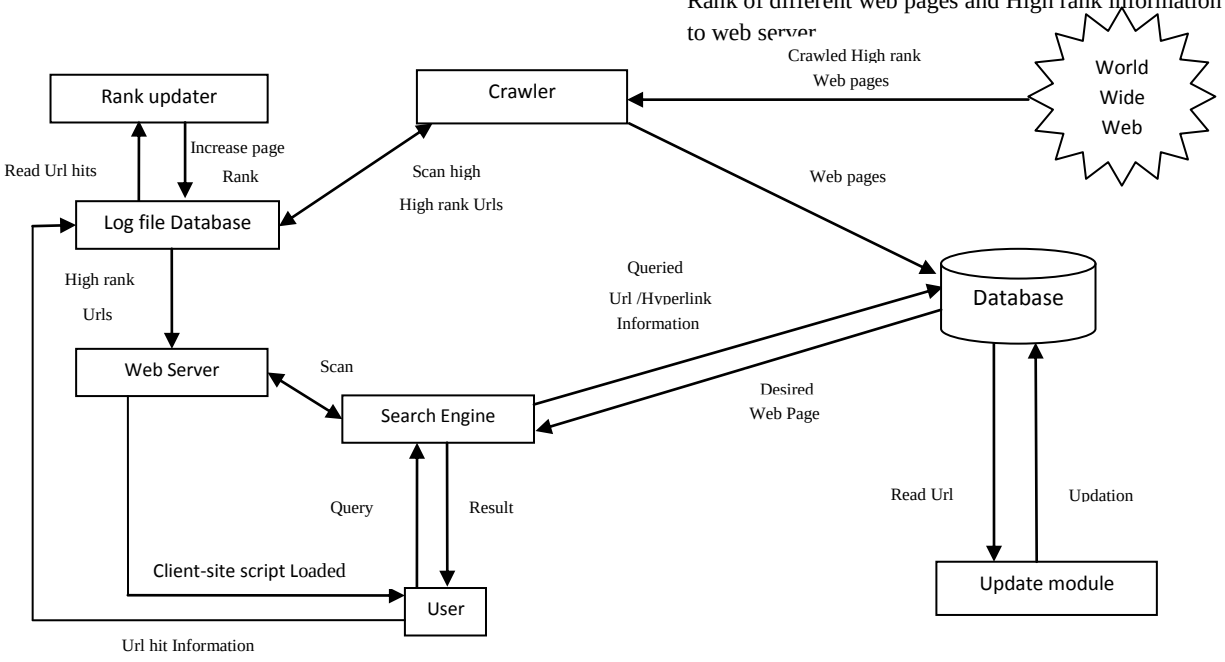

Figure 3. Architecture of proposed Incremental Crawler

The Page Revisit Frequency of Crawler is decided from here to visit those Urls whose rank is high in the log file database made up of user's search history .And finally from the Web, high rank pages are downloaded by crawler according to the

user's search history and are saved into the database. The database is updated by the Update Module time to time to save the newly arrived information and change the already saved information in the database so that user gets the latest information every time. Database provides the desired web pages to the search engine and from there final result is provided to the user. The algorithm for proposed incremental web crawler is as follows:

```
Incremental Crawler ( )
{
    User sends the query to search engine.
    Search engine send Url/Hyperlink to the database.
    Client-side script () starts
    Crawler use the log file database at the time of crawling
    to firstly download those Urls whose rank is high
    according to the user' search history.
    Downloaded web pages are saved into the database.
    Update module is further used for any updation required
    in saved web pages.
    Desired web pages are then sent to the search engine.
    Finally Searched results are provided to the user.
}
```

The algorithm for client side script is as follows:

```
Client-side script ( )
{
    Check the click and keyboard event.
    If event occurs over a hyperlink
    Then send a message about the current Url/hyperlink hit
    information to the web server.
    Hit_count ()
}
```

The algorithm for computation of Hit count is as follows:

```
Hit_count ()
{
    Firstly record web page_id, hyperlinks of that page and
    hit count of hyperlinks.
    And the Hit count(C) of every hyperlink will be
    incremented as the demand of web pages increases
    Save Hit count(C) of that hyperlink in the search
    engine's log file database.
    Rank Updater ()
}
```

The algorithm for Rank updater module is as follows:

```
Rank Updater ()
{
    While (Hit count(C) <Threshold value of hit i.e.100000)
    Rank of that URL will be increased by 0.001
}
```

4. CONCLUSION

As Compared to the previous proposed architecture and working of incremental crawler our proposed architecture helps us to manage the revisiting frequency of an incremental crawler based on the users search history which was not included earlier and is one of the major factor now a days as everybody wants to search the results of their choice with latest information about what they search. The proposed architecture manages the process of revisiting of a web site with a view to maintain fairly fresh documents at the search engine's database. The computation of the no. of hit count of user helps in finding the importance of the web pages. It also helps in improving their rank by the use of Rank Updater so that appropriate chance to each type of website to be crawled at a fast rate. In this way user can have desired web pages based on their search history maintained by the web server.

5. REFERENCES

[1] Niraj Singhal, Ashutosh Dixit and A. K. Sharma, "Design of A Priority Based Frequency Regulated Incremental Crawler", Published in International Journal of Computer Applications (IJCA), Volume 1–No. 1, Article 8, Harvard Press US 2010. ISSN: 0975–8887, pp 47-52, Feb 2010.

[2] A. K. Sharma, J. P. Gupta and D. P. Agarwal, " A novel approach towards management of Volatile Information" Journal of CSI, Vol. 33 No. 1, pp 18-27, Sept 2003.

[3] Alexandros Ntoulas, Junghoo Cho and Christopher Olston,"What's new on the Web ? The Evolution of the Web from a Search Engine perspective", In Proceedings of the World-Wide Web Conference (WWW), May 2004.

[4] Arvind Arasu, Junghoo Cho, Hector Garcia-Molina, Andreas Paepcke and Sriram Raghavan "Searching the Web" ACM Transactions on Internet Technology, 1(1): August 2001.

[5] Brian E. Brewington and George Cybenko. "How dynamic is the web.", In Proceedings of the Ninth International World-Wide Web Conference, Amsterdam, Netherlands, May 2000.

[6] Junghoo Cho and Hector Garcia-Molina, "The evolution of the web and implications for an incremental crawler", In Proceedings of the 26th International Conference on Very Large Databases, 2000.

[7] Junghoo Cho and Hector Garcia-Molina, "Estimating frequency of change", 2000, submitted to VLDB, Research track,2000.

[8] Sergey Brin and Lawrence Page, "The anatomy of a large scale hyper textual Web search engine". Proceedings of the Seventh International World Wide Web Conference, pp 107-117, April 1998.

[9] F. Douglis, A. Feldmann, and B. Krishnamurthy," 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, Monterey, California, Dec. 1997.

[10] D. Fetterly, M. Manasse, M. Najork, and J. L. Wiener," A large-scale study of the evolution of web pages", In Proceedings of the Twelfth International World Wide Web Conference, Budapest, Hungary, May 2003.

[11] Niraj Singhal and Ashutosh Dixit," Need of Search Engines and Role of a Web Crawler",National Conference on Recent Trends in Computers and IT (RTCIT-09),Samalkha, Haryana, 24th-25th April 2009.

[12] Ashutosh Dixit, Harish Kumar and A.K Sharma, "Self Adjusting Refresh Time Based Architecture For Incremental Web Crawler", International Journal of Computer Science and Network Security (IJCSNS), Vol 8, No12, Dec 2008.

[13] Niraj Singhal, Ashutosh Dixit and R. P. Agarwal, A. K. Sharma, "Regulating Frequency of a Migrating Web Crawler based on Users Interest", published in

International Journal of Engineering and Technology (IJET), Vol. 4, No. 4, Aug-Sep 2012, ISSN : 0975-4024, pp. 246-253.

[14] Arun Kumar Singh and Niraj Singhal, "A Novel Page Rank Algorithm for Web Mining based on User's Interest", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Vol. 2, Issue 9, pp. 395-400, September 2012.