ISSN 2229-5518

Comparitive analysis of various Ranking Algorithms for web Page Retrieval

Roopali Jindal[1], Ashok Kumar[2], Dr. Dheerendra Singh[3], Dr. R.N Rajotia[4]

***Abstract*—** The World Wide Web consists of millions of interconnected web pages that provide information to the user present in any part of the world. The World Wide Web is expanding and growing in size and the complexity of the web pages. That is why it is necessary to retrieve the best or the web pages that are more relevant in terms of information for the query entered by the user in the search engine. To extract the relevant web pages, the search engine requires a ranking module that applies a ranking algorithm on the web to fetch the web pages in order of the importance of the information entered by the user in the query. The ranking algorithm is much efficient to rank the surface web, i.e. the web pages that can be indexed by the search engine, as well as the hidden web, i.e. the web pages that cannot be indexed by the search engine. This paper compares the ranking algorithms that are very efficient to find the future scope and the methodology to calculate the relevancy of the web page.

Keywords— Search Engine, Crawler, Page Rank, Hidden Web, Architecture

1 Introduction

The World Wide Web (WWW) consists of two types of web pages: surface web (or visible web), deep web (or the hidden web or the invisible web). Deep web is the information on the web that is not indexed by the search engines. As the size of the web is increasing day-by-day, similarly the size of hidden web is also increasing. So, the amount of valuable information stored on the web in database is accessible only after the user enters a query through a search interface. For example, when the user enters a query the search engine does not have searchable results because the pages are not indexed by the search engine [1]. That information of the web pages that is not indexed is called as the hidden web pages.

2SEarch engine

A. Introduction

The search engines crawls the web and returns millions of web pages as result of user query. The search engine is a computer program that searches for the particular keywords entered by the user and returns a list of documents in which they were found. The resulted url list generated by the search engine is either first searched in its own local database and if the desired web pages is not found there then it fetches from WWW [2, 3]. The search engine consists of following components:

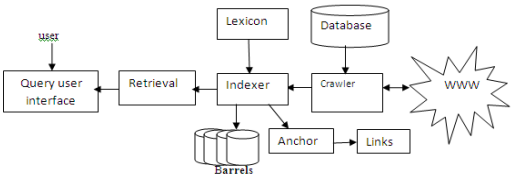

Fig.1. Architecture of search engine

The crawler/Spider module: To find the information on the internet where hundreds of millions of web pages exist, a typical search engine employs special software called spiders or crawlers to build lists of the words found on the websites. A web crawler is a program that automatically traverses the web by downloading documents and following links from page to page [3]. Search engines use web crawler to gather data for indexing the web pages. Crawlers are the automated programs that follow the links found on the web pages. The browser i.e. Internet Explorer, sends HTTP requests (hypertext transfer protocol), the most common protocol on the web which is used to retrieve the web pages and to download and show them on the user's computer system. A search engine assigns some URLs as the initial URL's for every web crawler. The work flow of a web crawler can be described as; a web crawler starts working with choosing the initial URLs. The web crawler issue a request (i.e GET request) to retrieve the web page. It then extracts all of the URLs (current URL's) in the web pages. The web crawler adds them to the queued URLs.

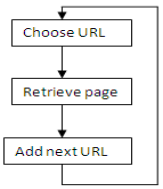

Fig.2. Crawler

The repository/database module- The repository or database has a very large collection of data objects. Each web page retrieved by the crawler is compressed and then stored in the repository with a unique ID associated with the URL and a note is taken of the length of each page [3].

The indexer/link analysis module- The information is present in the database in large amount so the information of web pages is to be stored in the most relevant order. It has a significant impact on web search as indexer takes a collection of data or documents and builds a searchable index. There could be a number of indexes built on the content of the pages so that the crawler can index the information required by the user. The indexer parses out all the links in every web page and stores important information about them in an

anchors file. Anchor file contains enough information to determine where each link points from and to, and the text of the link [4]. Each search engine creates its own custom dictionary or lexicon, which has to include every new 'term' discovered after a crawl. The lexicon stores every word so that it is aware of the words entered by the user in the query. Link analysis identifies the popularity and information how an anchor text can significantly enhance the presentation of pages pointed to by links.

The retrieval/ranking module- The retrieval means to find the files related with the query term. It calculates the scores for the documents using a ranking algorithm. This module is the core component of any search engine. Ranking algorithms are designed for efficient performance optimization to bring the relevant web pages. The retrieval module works on the output files of the indexer. It accepts the user queries from the query interface, executes the part of the query over its part of index and returns sorted results back to the query interface. The indexer performs a number of functions. It reads the repository, un-compress the documents, and parses them. The hits record the number of clicks on the web pages that is fetched by search engine using matching technique with user query. The indexer distributes the hits into a set of barrels that creates a partially sorted forward index which lists the URL's to the user in an ordered manner according to the unique doc Id assigned to each web URL. Page ranking methods are applied, which arrange the documents in order of their relevance, importance and rank score order the web page [4]. Some algorithms rely only on the link structure of the documents i.e. their rank scores, whereas some algorithms look for the content or data in the documents. Page rank algorithm assigns numerical weight to hyperlinked documents indexed by a search engine.

The user query interface- The user enters a query related to the information required by the user to the graphical user interface provided by the search engine. Most web interfaces are very simple; applications may use forms to make the user submit a query.

3 Ranking algorithm

There are various page ranking algorithms for the web page that allow indexing the web page according to the relevancy.

A. PAGE RANK ALGORITHM

Page rank algorithm is proposed by Larry page and Brin which is patented by Stanford University. It is the ranking algorithm used by Google's search engine to compute a page rank of the web page.

We compute the page rank algorithm by assuming a small universe of four web pages; A, B, C and D. The links from a page to itself or multiple outbound links from one single page to another single web page are ignored. Page Rank is initialized to the same value for all the web pages present in the web. In the Page Rank, we assume the sum of Page Rank over all pages equal to the total number of pages on the web at that time. We assume a probability distribution between 0 and 1 for all the web pages. The Page Rank transferred from a given page to the other web page of its outbound links in the next iteration is divided equally among all the outbound links of the given web page.

Let us suppose, the page B has a link to pages C and A, page C has a link to page A and page D have links to all three pages. We assume the initial value for each web page as 0.25.

To compute the page rank of A:

If the only links in the system were from pages B, C, and D to A, each outgoing link would transfer 0.25 to web page A to compute the Page Rank of A in the next iteration.

PR (A) = PR (B) + PR (C) + PR (D)

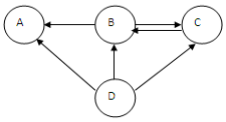

Fig.3. Web pages with backlinks

PR (A) = PR (B)/2 + PR(C)/1 +PR (D)/3

Thus, upon the next iteration, page B would transfer half of its existing value or 0.125 to page A , because page B has 2 back-links; to page A and to page C; and the other half or 0.125 to page C. And page C would transfer all of its existing value, which is 0.25, to the only page it links to the web page A. As shown in figure3 that web page D has three outbound links. The web page D would transfer one third of its existing value or 0.083 (0.25/3=0.083) values to web page A. At the completion of this iteration, page A will have a Page Rank of 0.458.

PR (A) = 0.25/2 + 0.25/1 + 0.25/3 = 0.458

Thus, the Page Rank calculated by an outbound link which is equal to the web page's own Page Rank score divided by the number of outbound links L( ).

PR (A) = PR (B)/L (B) + PR(C)/L(C) +PR(D)/ /L(D)

In the general case, the Page Rank value for any page u can be expressed as:

PR (u) = ∑VϵB(u) PR (v)/L (v)

The Page Rank value for a page u depends on the Page Rank value for each page v contained in the set B(u),the set which consists of all the web page linking to page u, divided by the number L(v) of links from page v.

Dampening factor- The Page Rank algorithm holds that an imaginary surfer who is randomly clicking on the links will eventually stop clicking and the probability that the person will continue is called as the damping factor d.

PR (A) = (1-d)/N + d(PR(B)/L(B) + PR(C)/L(C) + PR(d)/L(D) + …)

Thus, the equation is as follows [5]:

PR (pi) = (1 – d)/N + d ∑ pjϵM(pi) Lu(PR(pj))/ L(pj)

where p1 , p2 , p3, …. pn are the pages under consideration, M(pi) is the set of the web pages that link to page pi, L(pj) is the number of outbound links on the

page pj, and N is the total number of pages, N =total no. of web pages.

B. WEIGHTED PAGE RANK ALGORITHM

The weighted page rank algorithm is proposed by Wenpu Xing, Ali Ghorbani. This algorithm is an extension of the page rank algorithm. In this algorithm, unlike page rank algorithm, both the in-links and out-links are considered to determine the page rank of any page on the web. The in-links and out-links are computed as [6,10,11],

Win(v,u) is the weight of link(v, u) calculated based on the number of in links of page u and the number of in links of all reference pages of page v.

Win (v, u) = Iu /∑ pϵR(v)Ip

where Iu and Ip represent the number of in links of page u and page p respectively. R (v) denotes the reference page list of page v.

Wout (v, u) is the weight of link(v, u) calculated based on the number of out links of page u and the number of out links of all reference pages of page v.

Wout (v, u) = Ou / ∑pϵR(v)Op

Where, Ou and Op are the number of out-links of the web page u and page p respectively, R (v) denotes the reference page list of page v.

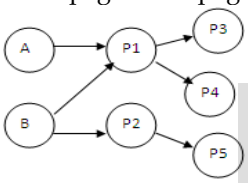

Fig.4. Links on web pages

In the above example, we assume web pages as page A,B , P1,P2,P3,P4,P5. Now, we compute in links and out links as:

Win (A, p1) = IP1 /IP1+IP2 = 2/2+1 = 2/3

Wout (A, p1) = OP1 /OP1+OP2 = 2/2+1 = 2/3

The weighted page rank (WPR) computes the page rank of any web page by modifying the page rank algorithm and is formulated as in the following equation:

PR (u) = (1 – d) +d ∑vϵB(u) PR (v)Win (v,u) Wout (v,u)

Here, d is a dampening factor , u represents a web page, B(u) is the set of pages that point to u, Win (u,v) and WPRout(v,u) are the rank scores of the in-links from page u and v, and the outgoing links from v to u, respectively, L(u) is the number of visits of link which is pointing page u from v.

C. Page ranking based on number of Visits of links of web page

This is a improved version of page ranking algorithm, proposed by Gyanendra Kumar, Neelam Duhan and A.K Sharma. The original page rank algorithm considers the number of out-links from a web page and computes a rank score for each page. In this algorithm we assign more rank value to the outgoing links which is mostly visited by the users. So, here a page rank value is calculated on the number of visits (Lu) of the inbound links [7].

The page ranking based on the number of visits of links on web page is formulated by the following equation,

PR (pi) = (1 – d) + d ∑pjϵM(pi) Lu* (PR (pj))/ L(pj)

Here, PR (pi) is the page rank of page I, d is the dampening factor, L (u) is the number of the visits on the link, L(pj) is the number of outbound links on page pj.

In this algorithm, the computation of the number of visits on the links on the page rank algorithm increases the relevancy of web page and also search space is reduced.

D. Weighted page rank algorithm based on number of visits of links of web page

This ranking algorithm is proposed by Neelam Tyagi and Simple Sharma. In the algorithm, we assign more rank value to the outgoing links which is mostly visited by the users and to the links which receives higher popularity from the number of in-links. The user's browsing behavior on the in-links is determined by calculating the number of visits on in-links. The number of visits on the link is represented by L (u).

The modified weighted page ranking algorithm is given as the following [8],

PR (u) = (1 – d) +d ∑vϵB (u) L(u) * PR(v)Win (v,u) Wout (v,u)

Here, d is a dampening factor, u represents a web page, B(u) is the set of pages that point to u, Win (u, v) and WPRout(v,u) are the rank scores of the in-links from page u and v, and the outgoing links from v to u, respectively, L(u) is the number of visits of link which is pointing page u from v, L(u) denotes the number of visits on the in-links.

This ranking algorithm increases the relevancy of the web page than page rank algorithm by calculating the number of links on the web page by running a client side script to calculate the number of visits or clicks on any link.

E. Improved time aware incremental page ranking using personalized link structure analysis

This ranking algorithm is proposed by Shail K Dinkar, Hemant Kumar.Time Aware incremental page rank algorithm analyzes the structure of a link to find out the variation of time spent by a user on a link. There may be a possibility that a user visits a particular page on a day X(let the page rank value be x) for duration t1 and the same user visits the same page on day Y for duration t2.So, this algorithm proposes the due change in visiting time of the same particular page that will cause the change in the page rank value of the page(let it be y) [9].So, the proposed algorithm calculates the page rank of the web page by measuring the change in visiting time of the same user on the same day which predicts the link structure and the user behavior for a particular web page. A parameter D is used to calculate the difference between the visiting time of the same user on the same link at different time.

The parameter D is used to find the time spent by a user on a particular page in a day and is calculated by the following equation,

D = |T(i)previous – T(i)current| / H*M*S

Where H= 24 hours. M= 60 minutes, S= 60 seconds.

The ranking value of a page is computed by the following equation,

PR (i) = (1-d) +d * ∑Ejϵ S (PR (j) / L (j)) / T(i) + D

F. Content Based Hidden web ranking algorithm

This ranking algorithm(CHWRA) is proposed by Neha Batra, Ashok Kumar, Dheerendra singh and RN Rajotia [12]. The proposed ranking algorithm consists of following factors,"Term Weightage (TW)" technique has been given the highest priority, and then is the "Visitor Count (VC)" with the second highest priority, after that, user feedback "Like/Dislike (FB)" is taken into consideration and in the end the "PageRank (PR)" algorithm is considered. Content based hidden web ranking algorithm gives better ranked results and arrange webpages according to their relevancy.

The equation is as following:

Rank of URL A =TW+VC+ FB+ PR

Where, VC= Visitor count counts number of visits on the web page,

TW= Term weightage calculates document length, document frequency and term frequency of a keyword in the document.

FB= User's feedback in the form of likes and dislikes affects the page rank value of a webpage.

PR= Page rank of a web page as described in page rank algorithm.

4 COMPARISON OF PAGE RANKING ALGORITHM

The comparison of various page ranking algorithms is given in a tabular form below,

S.No	Ranking Algorithm	Proposed By	Conclusion
1	Page Rank	Larry Page, Brin	This algorithm computes the score for pages on out-links
2	Weighted Page rank	Wenpu Xing, Ali Ghorbani	Increases relevancy of webpage by computing weights on outlinks and inlinks
3	Page ranking based on number of visits of links of webpage	Gyanendra Kumar, Neelam Duhan, AK Sharma	The relevancy of webpage is increased and the search space is reduced
4	Weighted page rank algorithm based on number of visits of links of webpage	Neelam tyagi, Simple Sharma	Increases the relevancy of the webpage by calculating number of links
5	Improved time aware incremental page ranking using personalized link structure analysis	Shail K Dinkar, Hemant Kumar	Inspite of having less incoming links to the newer web page, this algorithm provides appropriate result.
6	Content based hidden web ranking algorithm	Neha Batra, Ashok Kumar, Dheerendra Singh and RN Rajotia	This algorithm gives better ranked results

Table1. Comparison between page ranking algorithms

5 CONCLUSION AND FUTURE WORK

The comparative study on page ranking algorithm describes various parameter on the different ranking algorithm such as number of visiting links, time stamp, that are used to compute the rank score of a web page to arrange web pages according to the relevancy. It is observed that CHWRA retrieves better ranked results from the previous ranking algorithms. It removes the drawbacks of other discussed ranking algorithms by including some factors like term weightage, visitor count and user's feedback. The existing ranking algorithm can be extended with a ranking factor, e.g. time spent on the web page by any user, which may increase the relevancy of the web pages retrieved from the World Wide Web and further provides much more relevant ranked results.

REFERENCES

[1] George Valkanas, Alexandros Ntoulas, Dimitrios Gunopulos; " Rank-Aware Crawling of Hidden Web sites " , http://cgi.di.uoa.gr/~gvalk/pubs/rachweb.pdf

[2] How search engine works , Mike Grehan, Search engine watch; http://www.mentevisual.com.ec/pdf/search-engines.pdf

[3] Knut , Mathias , Yngve ,"Multi tier architecture for web search engine", IEEE 2003; Proceedings of the First Latin American Web Congress (LA-WEB 2003) 0-7695-2058-8/03 $17.00 © 2003 IEEE

[4] Monika Peshave , KamyarDezhgosha , " How a search engine works and a web crawler application "; http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.133.6059&rep=rep1&type=pdf

[5] Larry Page; "Page ranking algorithm" ; Stanford university; http://google.stanford.edu

[6] Wenpu Xing, Ali Ghorbani; " Weighted Page Rank Algorithm"; Proceedings of the Second Annual Conference on Communication Networks and Services Research (CNSR'04) 0-7695-2096-0/04 $20.00 © 2004 IEEE

[7] Gyanendra Kumar, Neelam Duhan , A.K Sharma; "Page ranking based on number of Visits of links of web page" ; International Conference on Computer & Communication Technology (ICCCT)-2011

[8] Neelam Tyagi , Simple Sharma ; "Weighted page rank algorithm based on number of visits of links of web page" ; International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012

[9] Shail K Dinkar, Hemant Kumar; "Improved time aware incremental page ranking using personalized link structure analysis"; IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 2, November 2012, ISSN (Online): 1694-0814

[10] NeelamDuhan, A. K. Sharma, Komal Kumar Bhatia; "Page ranking algorithms",IEEE International Advance Computing Conference (IACC 2009)

[11] Dilip kumar sharrna, AK sharma ,"A Comparative Analysis of Web Page Ranking Algorithms"; International Journal on Computer Science and Engineering Vol. 02, No. 08, 2010, 2670-2676

[12] Batra, Neha.; Kumar, Ashok.; Singh, Dheerendra.; Rajotia, R.N., "Content based hidden web ranking algorithm(CHWRA)," Advance Computing Conference (IACC), 2014 IEEE International , vol., no., pp.586,589, 21-22 Feb. 2014 doi: 10.1109/IAdCC.2014.6779390

IJSER