Query Independent Time Dependent Page Ranking Algorithm for Web Information Retrieval

L Smitha,
G Narayanamma Institute of Technology and science, Hyderabad, India.

S Sameen Fatima
Osmania University, Hyderabad, India.
Dept of Computer science and Engineering
University College of Engineering,

ABSTRACT.

With the remarkable growth of information obtainable to end users through the web, search engines come to play ever a more significant role. The search engines sometimes give disappointing search results for lack of any classification of search. If we can somehow find the preference of user about the search result and rank pages according to that preference, the result will be more accurate to the user. In this paper page ranking algorithm is being proposed based on the notion of query independent constrained ranked retrieval, which, given a query and a time constraint, produces the best possible ranked list within the specified time limit. The proposed algorithm is based on the constraint that makes Query independency feature where keywords are given less weights and the hyperlinks used with time selection decisions are used. We record the visited time of the page using Log files it means we use the time factor to get better precision of the ranking. Experiments on different test collections show that this algorithm is able to satisfy imposed time constraints, and being able to deliver more effective results, especially under tight time constraints. The proposed approach mainly consists of three steps: select some web pages based on user's demand, measure their damping factor, and give different weightage to each page depending upon how much time user spending on the web page. The results of our simulation studies show that algorithm performs better than the conventional PageRank algorithm in terms of returning larger number of relevant pages to a given query.

Keywords

Search engine, Random surfer, Hub, Authority, Markov chain, PageRank**.**

1. Introduction.

Web is huge and expending day by day and users generally rely on search engine to discover the web. In such a situation it is the responsibility of service provider to provide proper, significant and quality information to the internet user against their query submitted to the search engine. It is a challenging issue for service provider to provide proper, appropriate and quality information to the internet user by using the web page contents and hyperlink between the web pages. Figure 1 [2] shows a working of a typical search engine, which shows the flow graph for a searched query by a web user. In the present paper a web page ranking algorithm is being proposed based on the notion of query independent constrained ranked retrieval, which, given a query and a time constraint, produces the best possible ranked list within the specified time limit. Logically, more time should translate into better results, but the ranking algorithm should always produce some results

In the last years, with the gigantic growth of the Web, we assisted to an explosion of information accessible to Internet users. however, at the same time, it has become ever more critical for end users to search this huge repository and find needed resources by simply follow the hyperlink network as foreseen by Berners-Lee and Fischetti [1]. Today, search engines constitute the most helpful tools for organizing information and extracting knowledge from the Web [3]. However, it is not uncommon that even the most renowned search engines return result sets including many pages that are useless for the user [4]. This is due to the fact that the very basic relevance criterions underlying their information retrieval strategies rely on the presence of query keywords within the returned pages. It is to observe that statistical algorithms are applied to "tune" the result and, more importantly, approaches based on the concept of relevance feedback are used in order to maximize the satisfaction of user's needs. Nevertheless, in some cases, this is not enough.

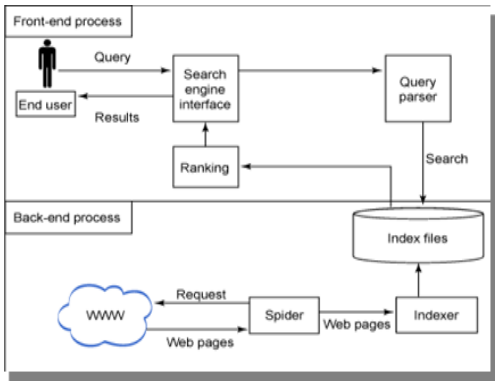


Figure1. Search engine architecture

Ranking is the central part of many applications including document retrieval, recommender systems, advertising and so on. Many ranking algorithms have been proposed previously, including HITS Algorithm ranks the web page by processing in links and out links of the web pages. In this algorithm a web page is named as authority if the web page is pointed by many hyper links and a web page is named as HUB if the page point to various hyperlinks. An Illustration of HUB and AUTHORITY are shown in figure 2.

1.1HISTORY

The AltaVista Search Engine implements HITS[10] is a link based algorithm ignoring textual content . In which ranking of the web page is decided by analyzing their

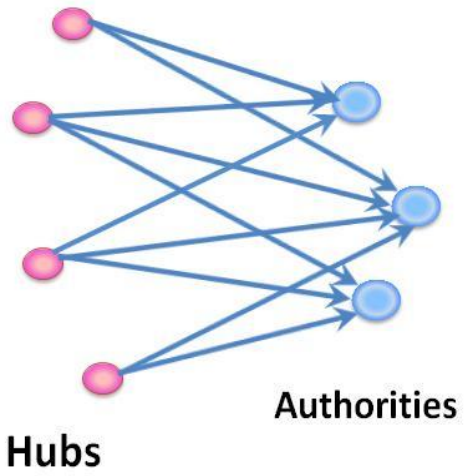


Figure 2. HUB and AUTHORITY

textual contents against a given query. After collection of the web pages, the HITS algorithm concentrates on the structure of the web only, neglecting their textual contents. TagRank Algorithm [11] for ranking the web page based on social annotations. This algorithm calculates the heat of the tags by using time factor of the new data source tag and the annotations behavior of the web users. Relation Based Algorithm [12] proposed this algorithm for the ranking the web page for semantic web search engine. Various search engines are presented for better information extraction by using relations of the semantic web. This algorithm proposes a relation based page rank algorithm for semantic web search engine that depends on information extracted from the queries of the users and annotated resources. Results are very encouraging on the parameter of time complexity and accuracy. Weighted Page Rank Algorithm [14] decides the rank score based on the popularity of the pages by taking into consideration the importance of both the in-links and out-links of the pages. This algorithm provides high value of rank to the more popular pages and does not equally divide the rank of a page among it's out-link pages. Every out-link page is given a rank value based on its popularity. Popularity of a page is decided by observing its number of in links and out links. The length of anchor text seems to be the best attributes in this algorithm. Relative position, which reveals that physical position does not always in synchronism with logical position, is not so result oriented. Distance Rank Algorithm [15] an intelligent ranking algorithm, which is based on reinforcement learning algorithm. In this algorithm, the distance between pages is considered as a punishment factor. In this algorithm the ranking is done on the basis of the shortest logarithmic distance more quickly with the use of distance based solution.

1.2 Types of ranking algorithms.

Query dependent: Rank a small subset of pages related to a specific query where HITS (Kleinberg 98) was proposed as query dependent

Query independent: Rank the whole Web (Brin and Page) in the year 1998

2. Related work.

The AltaVista Search Engine implements HITS[10] is a link based algorithm ignoring textual content . In which, ranking of the web page is decided by analyzing their textual contents against a given query. After collection of the web pages, the HITS algorithm concentrates on the structure of the web only, neglecting their textual contents. TagRank Algorithm [11] for ranking the web page based on social annotations. This algorithm calculates the heat of the tags by using time factor of the new data source tag and the annotations behavior of the web users. Relation Based Algorithm [12] proposed this algorithm for the ranking the web page for semantic web search engine. Various search engines are presented for better information extraction by using relations of the semantic web. This algorithm proposes a relation based page rank algorithm for semantic web search engine that depends on information extracted from the queries of the users and annotated resources. Results are very encouraging on the parameter of time complexity and accuracy. Weighted Page Rank Algorithm [14] decides the rank score based on the popularity of the pages by taking into consideration the importance of both the in-links and out-links of the pages. This algorithm provides high value of rank to the more popular pages and does not equally divide the rank of a page among it's out-link pages. Every out-link page is given a rank value based on its popularity. Popularity of a page is decided by observing its number of in links and out links. The length of anchor text seems to be the best attributes in this algorithm. Relative position, which reveals that physical position does not always in synchronism with logical position, is not so result oriented. Distance Rank Algorithm [15] an intelligent ranking algorithm, which is based on reinforcement learning algorithm. In this algorithm, the distance between pages is considered as a punishment factor. In this algorithm the ranking is done on the basis of the shortest logarithmic distance between two pages and ranked according to them. The Advantage of this algorithm is that it can find pages with high quality and more quickly with the use of distance based solution.

2.1 Page Rank Algorithm.

Web pages are represented as a graph G, where each web page is a node and a hyperlink between web pages is a link between two nodes in the graph. It is generally assumed that the relevance of a page is represented by the probability of ending up in that page during surfing on this graph. Within the graph, it is assumed that when a user is surfing, user may do following actions at any time:

1. Jump to a node of the graph
2. Follow a hyperlink from the current page
3. Track a back-link
4. Remain in the same node/page

Furthermore, it is assumed that user actions will depend on the page contents and the links of the web page. Therefore, if the user likes the page, he will likely follow a link contained there, otherwise, he will jump to a new page, perhaps an unrelated one.

The probability that random surfer will get bored and restart from some another random document is damping Factor (say α). We can view a random surfer on the web graph [5] as a Markov chain, with one state for each web page, and each transition probability representing the probability of moving from one web page to another. The teleport operation contributes to these transition probabilities.
We can readily derive the P (transition probability matrix) for our Markov chain from the $N \times N$ matrix A:
1. If a row of A (*Adjacency matrix)* has no 1's, then replace each element by 1/N. For all other rows proceed as follows.
2. Divide each 1 in A by the number of 1's in its row. (eg Thus, if there is a row with three 1's, then each of them is replaced by 1/3).
3. Multiply the resulting matrix by $1 - \alpha$.
4. Add α/N to every entry of the resulting matrix, to obtain P.

We can depict the probability distribution of the surfer's position at any time by a probability vector $\vec{x}$.

At $t = 0$ the surfer may begin at a state whose corresponding entry in $\vec{x}$ is 1 while all others are zero. By definition, the surfer's distribution at $t = 1$ is given by the probability vector $\vec{x}P$;
$at\ t = 2\ (\vec{x}P)P = \vec{x}P^2$ And so on.

We can thus compute the surfer's distribution over the states at any time, given only the initial distribution and the transition probability matrix P. If a Markov chain is allowed to run for many time steps, each state is visited at a (different) frequency that depends on the structure of the Markov chain. In our running analogy, the surfer visits certain web pages (say, popular news home pages) more often than other pages. We now make this intuition precise, establishing conditions under which such the visit frequency converges to fixed, steady-state quantity. Following this, we set the PageRank of each node v to this steady-state visit frequency and show how it can be computed.

The importance of pages is different to users even though their link structure is same because their content is different. So how to relate the content and link structure together? If the page is interested by the user, the visited time will also be longer than the pages don't accord with user's interest. It means the content of page is more probably the user wants to search. If we add the visiting time into the computation of ranking, we may progress the accuracy of ranking score of pages. So how to estimate the visiting time of the pages is the key problem in the present algorithm.
In section 3.1, Proposed Algorithm and detailed description of Algorithm in section 3.2

3. Proposed work.

3.1. Algorithm.

In order to estimate the user's visiting time of the page, we make a reasonable assumption that the user clicks the page orderly from the returned set of pages after submitting the query. It means that after the user type the key word into the search engine, user will get a set of related pages. The user will only click one page at a time, and will not click the pages until he (or she)finishes browsing the opened page. In other words, the user will not open more than two pages at one time, but will browse the pages one after another. If the user browses the pages in this mode, there exists a visiting sequence.
Let P represent the sequence vector.
So $P = \{P_1, P_2 \ldots, P_n\}$, where n is the number of pages related to the key words present in the query, the user visits in the returned set. Based on this sequence, we can examine the logs of the search engine and calculate the visited time of the pages.
Let T_j represent the time point the user click page P_j and T_{j+1} represent the time point the user click page P_{j+1}.

So the time between T_j and T_{j+1} is the user's visiting time of page P_j.
Let t_j represent page's visited time by the user at one time and $t_j = T_{j+1} - T_j$.
The sequence of user's visiting time $T_s = \{t_1, t_2 \ldots t_n\}$.

We add time factor into the random surfing model. Based on random surfing model [fig1.], if the user interested in page's content, the page's topic may be related to the key words and the visiting time may long. Otherwise, the user may leave the page quickly and the visiting time is short. So the process of computation is as follows.
Rank is hyperlink related. Each page has n Time scores, where n is the number of topics. Firstly, compute the algorithm for a web graph and assign a rank to every page offline. This computation is based on the work [13].
Second, compute the similarity between key words and hyperlink. After the user submitting the key words to the search engine, it has to be checked for which hyperlink, the key words belong to.
According to the Bayesian theory [6], the relative probability between query q and hyperlink j is

$$\Pr(H(j)|q) = \frac{\Pr(q,H(j))}{\Pr(q)}$$

$$= \frac{\Pr(H(j)) * \Pr(q|H(j))}{\Pr(q)}$$

$$\approx (\Pr(H(j)) * \Pr(q|H(j))$$

where h(j) represents the hyperlink j of each page, and Pr(H(j)) means the proportion of pages related to hyperlink j in the whole pages set, and Pr(q|H(j)) means the probability of the query q belonging to hyperlink j. The purpose of this computation is to judge how to accumulate time to each hyperlink.

The last step is time addition. For each page, there is a initial hyperlink related to visited time vector $T_v = \{t(1), t(2), \ldots, t(n)\}$, Where t(j) represents to the user's total visiting time of a page related to hyperlink j. To avoiding the zero, the initial t (j) =1. So, after the search engine runs for some time, we can

get the visited time vector of each page. This vector is accumulated from the sequence of user's visiting time T_s.

So the Rank of each page is

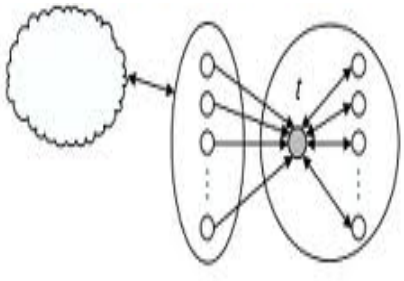
Rank (j) = [Pr ((H (j) |q)] ∗t (j) (5)

We add time factor into We add time factor into the random surfing model. Based on random surfing model [fig1.], if the user interested in page's content, the page's topic may be related to the key words and the visiting time may long. Otherwise, the user may leave the page quickly and the visiting time is short. So the process of computation is as follows.

Rank is hyperlink related. Each page has n Time scores, where n is the number of topics. Firstly, compute the algorithm for a web graph and assign a rank to every page offline. This computation is based on the work [13].

Second, compute the similarity between key words and hyperlink. After the user submitting the key words to the search engine, it has to be checked for which hyperlink, the key words belong to the purpose of this computation is to judge how to accumulate time to each hyperlink.

According to the Bayesian theory [6], the relative probability between query q and hyperlink j. The purpose of this computation is to judge how to accumulate time to each hyperlink.



3.2. Algorithm Description

In order to represent Time in Ranking, we should first stimulate "time-based visiting model". In the model, We have measured the visit time of the page from log files, after applying original and improved methods of web page rank algorithm to know about the degree of importance to the users. This algorithm utilizes the time factor to increase the accuracy of the web page ranking.

Here, in the ranking algorithm meant pages of longer visited time, which will get a higher score, no matter whether the link structure of two pages are same or not. We can see, after adding visited time into the computation of ranking score, we can not only get a more accurate ranking scores of each page, but also avoiding the spamming pages get a high score in the search engine. This is the work contribution in the paper.

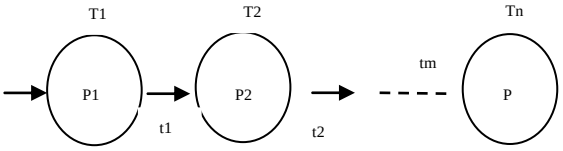


Figure 4. Sequential clicking model

The last step is time addition. For each page, there is a initial hyperlink related to visited time vector T_v = {t(1), t(2), …, t(n)},

Where t(j) represents to the user's total visiting time of a page related to hyperlink j. To avoiding the zero, the initial t (j) =1. So, after the search engine runs for some time, we can get the visited time vector of each page. This vector is accumulated from the sequence of user's visiting time T_s.

So the Rank of each page is

Rank (j) = [Pr ((H (j) |q)] ∗t (j) (5)

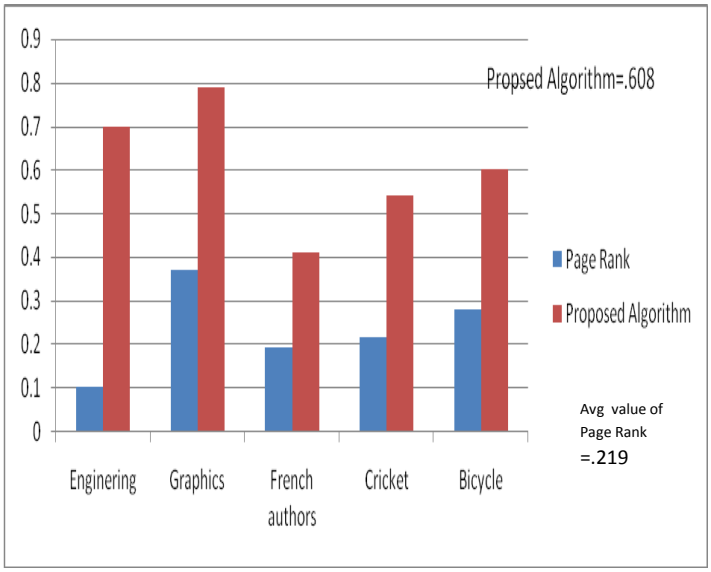


Figure 5. Algorithm result

3.3 The process of log files

In order to get pages' visited time from server's logs, we have to process the logs first. The query word, source ip, visited time are all recorded in server's logs. Because of noises in the logs, we should discriminate the noises first, and arrange the items according to source internet protocol address. Negotiating the use of web proxy, if the source ip in an hour are same, we treat it as the clicking sequence of one user. If the interval time between neighbor items is more than an hour, we will discard the latter item. Under this rule, we can get the page's visited time and computed it by (5).

4. References

[1] T. Berners-Lee and M. Fischetti, Weaving the Web. Harper Audio,1999

[2] Neelam Duhan, A. K. Sharma and Komal Kumar Bhatia, "Page Ranking Algorithms: A Survey", In proceedings of the IEEE International Advanced Computing Conference (IACC), 2009.

[3] L. Ding, T. Finin, A. Joshi, Y. Peng, R. Pan, and P. Reddivari, "Search on the Semantic Web," Computer, vol. 38, no. 10, pp. 62-69, Oct. 2005.[4] A. Pisharody and H.E. Michel, "Search Engine Technique Using Keyword Relations," Proc. Int'l Conf. Artificial Intelligence (ICAI '05), pp. 300-306, 2005.

[5] Christopher D. Manning ,Prabhakar Raghavan, Hinrich Schütze "An Introduction To Information Retrieval book" .

[6] Dimitri, P. Betsekas and John N. Tsitsiklis, Introduction to Probability. Athena Scientific, 2002.

[7] I. Kang and G. Kim, Query type classification for web document retrieval, In Proceedings of ACM SIGIR'03, 2003.

[8] D. E. Rose and D. Levinson, Understanding user goals in Web search, In Proceedings of the 13th International World Wide Web Conference, New York, USA , pp: 13 – 19, 2004.

[9] R. Montenegro,P. Tetali; "Mathematical aspects of mixing times in Markov chains" Foundations and Trends in Theoretical Computer Science Volume 1 , Issue 3 (May 2006) Pages: 237 - 354 ;

[10] Jon Kleinberg, "Authoritative Sources in a Hyperlinked Environment", In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 1998.

[11] Shen Jie,Chen Chen,Zhang Hui,Sun Rong-Shuang,Zhu Yan and HeKun, "TagRank: A New Rank Algorithm for Webpage Based on Social Web" In proceedings of the International Conference on Computer Science and Information Technology,2008.

[12]Fabrizio Lamberti, Andrea Sanna and Claudio Demartini , "A Relation-Based Page Rank Algorithm for. Semantic Web Search Engines", In IEEE Transaction of KDE, Vol. 21, No. 1, Jan 2009.

[13] L. Page, S. Brin, R. Motwani and T. Winograd, The pagerank citation ranking: Bringing order to the web, Technical report, Stanford Digital Library Technologies Project. 1998

[14] Wenpu Xing and Ali Ghorbani, "Weighted PageRank Algorithm", In proceedings of the 2rd Annual Conference on Communication Networks & Services Research, PP. 305-314, 2004.

[15] Ali Mohammad Zareh Bidoki and Nasser Yazdani, "DistanceRank: An Iintelligent Ranking Algorithm for Web Pages", Information Processing and Management, 2007.