Volume 3, Issue 10, October 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Analytical Approach to Ranking Pages Based on Mortal Navigational Demeanor

Dr. Sandeep Gupta
(*Department of Computer Science & Engineering*)
N.I.E.T Gr.Noida
MTU University, Noida, U.P, India

Chhiteesh Rai
M.Tech Research Scholar,(CSE)
V.I.E.T Dadri, Gr.Noida,
MTU University, Noida, U.P, India

Abstract- Mining of data encounters many new challenges with increased amount of information on data repository (Data warehouse, Database, World Wide Web etc.). Data repository documents have been main resource for various purposes; people are really wanted to search the required information in a very efficient manner. The search engines play vital role for retrieving the required information from huge information. In this we assume World Wide Web information as a data repository used for search engine to return quality result by scoring the relevance of web document.. Search engines generally return a large number of pages in response to user queries. To assist the users to navigate in the result list, ranking methods are applied on the search results. Most of the ranking algorithms in the literature are either link or content oriented, which do not consider user usage trends. Here, in this paper, we have used the asp.net framework for calculation of page rank using PRABUBB (Page Ranking Algorithm Based on user Browsing Behavior) for search engines. This framework works on the basic ranking algorithm of Google i.e. PageRank takes number of visits of inbound links of Web

Keywords- Web Mining, Pageindex Incharge, Page Rank, Prabubb, Asp.net.

I. INTRODUCTION

The World Wide Web (Web) is popular and interactive medium to propagate information today. The Web is huge, diverse, dynamic, widely distributed global information service center. As on today WWW is the largest information repository for knowledge reference. With the rapid growth of the Web, users get easily lost in the rich hyperlink structure. Providing relevant information to the users to cater to their needs is the primary goal of website owners. Therefore, finding the content of the Web and retrieving the users' interests and needs from their behavior 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 [1], various ranking meets are applied on the search results.

II. Related Work

PageRank [2, 3] and HITS [4] are popular link analysis algorithms in the literature. The basic idea of PageRank is as follows: the link from a webpage to another can be regarded as an endorsement of the linking page, the more links pointed to a page, the more likely it is important, and this importance information can be propagated across the vertices in the graph. A discrete-time Marko process model which simulates a web surfer's random walk on the graph is defined and page importance is calculated as the stationary probability distribution of the Markov process. HITS are based on the notions of hub and authority to model the two aspects of importance of a webpage. A hub page is one from which many pages are linked to, while an authority page is one to which many pages are linked from. In principle, good hubs tend to link to good authorities and vice versa. Previous study has shown that HITS performs comparably to PageRank [7].Many algorithms have been developed in order to further improve the accuracies and efficiencies of PageRank. Some work focuses on speed-up of the computation [8, 9], while others focus on refinement and enrichment of the model. For example, Topic sensitive PageRank [10] and query-dependent PageRank [11] have been proposed. The basic idea of these algorithms is to introducetopics and assumes that the endorsement from a page that belongs to the same topic is larger. Other variations of PageRank include those modifying the 'personalized vector' [12], changing the 'damping factor' [13], and introducing inter-domain and intra domain link weights [14]. Besides, there is also work on theoretic issues of PageRank [15]

III. Pagerank Algorithm Based On User Browsing Behavior (Prabubb)

A. User Behavior Data

Many web service applications assist users in their accesses to the web; sometimes they record user behaviors under agreements with them. When a user surfs on the web, she usually has some information need. To browse a new page, the user may choose to click on the hyperlink on another page pointing to it, or to input the URL of it in to the web browser.

The user may repeat this until she finds the information or gives up. The user behavior data can be recorded and represented in triples consisting of <URL, TIME, and TYPE> (see Table 1 for examples). Here, URL denotes the URL of the webpage visited by the user, TIME denotes the time of the visit, and TYPE indicates whether the visit is by a URL input (INPUT) or by a hyperlink click on the previous page (CLICK). The records are sorted in chronological order

Table1. Examples of user browsing behavior data

URL	TIME	TYPE
http://xxy.yyx.com/	2013-04-12, 21:33:05	INPUT
http://yyz.zzy.com/1.htm	2013-04-12, 21:34:11	CLICK
http://zzx.xxz.org/index.htm	2013-04-12, 21:34:52	CLICK
-------	--------	----------

From the data we extract transitions of users from page to page and the time spent by users on the pages as follows:

1) Session segmentation:-We define a session as a logical unit of user's browsing. In this paper we use the following two rules to segment sessions. First, if the time of the current record is 30 minutes behind that of the previous record, then we will regard the current record as the start of a new session [16], otherwise, if the type of the record is 'INPUT', then we will regard the current record as the start of a new session. We refer to the two rules as the time rule and the type rule hereafter.

2) URL pair construction:-Within each session, we create URL pairs by putting together the URLs in adjacent records. A URL pair indicates that the user transits from the first page to the second page by clicking a hyperlink.

3) Reset probability estimation: - For each session segmented by the type rule, the first URL is directly input by the user and not based on a hyperlink. Therefore, such a URL is 'safe' and we call it green traffic. When processing user behavior data, we regard such URLs as the destinations of the random reset (when users do not want to surf along hyperlinks).We normalize the frequencies of URLs being the first one in such sessions to get the reset probabilities of the corresponding web pages.

4) Staying time extraction: - For each URL pair, we use the difference between the time of the second page and that of the first page as the observed staying time on the first page. For the last page in a session, we use the following heuristics to decide its observed staying time. If the session is segmented by the time rule, we randomly sample a time from the distribution of observed staying time of pages in all the records and take it as the observed staying time 5. If the session is segmented by the type rule, we use the difference between the time of the last page in the session and that of the first page of the next session (INPUT page) as the staying time.

By aggregating the transition information and the staying time information extracted from the records by an extremely large number of users, we are able to build a user browsing graph (see Figure1). Each vertex in the graph represents a URL in the user behavior data, associated with reset probability and staying time as metadata. Each directed edge represents the transition between two vertices, associated with the number of transitions as its weight. In other words, the user browsing graph is a weighted graph with vertices containing metadata and edges containing weights. We denoted it as G =< V,W, T, σ>, where V = $\{v_i\}$W = $\{w_{ij}\}$,T = $\{T_i\}$, σ = $\{\sigma_i\}$, (I, j = 1,……., N) denote vertices, weights of edges, lengths of staying time, and reset probabilities, respectively denotes the number of web pages in the user browsing graph.

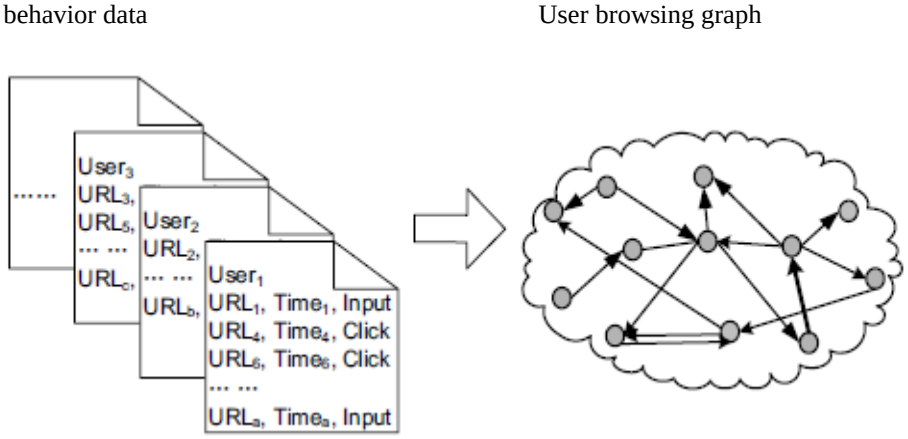

Figure 1. User behavior data & User browsing graph

Updating the PageRank is a big implementation concern since the web is not static. The calculated PageRank now may not be the same after a few days. Extensive research is being done for being able to use the old scores of a page to calculate the new PageRank without having to reconstruct everything. Also, the changing link structure and the addition and deleting of webpages must be taken care of. Moreover to assign larger rank values to more important (popular) pages instead of dividing the rank value of a page evenly among each outlink page to get a value proportional to its popularity (its number of inlinks and outlinks)[5]

B. Parameter

Parameters are function or variable those are used in any algorithm to calculate the output of algorithm. In my proposed algorithm following parameter are used-

1).Page loading &Page review – It defines total time taken by page in loading. If a page takes more time to load its rank score reduced.

2).System configuration for page- System configuration must be in order so that it can support page the page that is going to load.

3).User checking fails- If on a particular page user checking option fails frequently then it should be removed otherwise it causes the degradation of rank.

4).Page design and form button working- Design of page and button should proper otherwise it reduces the rank score of page.

5).Planning and controlling of page hitting- Page hitting operation is the most important factor in deciding the pagerank, so page hitting option must be taken care.

6).Password field is not mandatory- If a page contain password field, then it must be mandatory otherwise it causes bad impact on rank score.

7).Page complexity &encryption- Page complexity &encryption is a great factor to calculate rank score of a page. We assign the rank score to a page on the basis of complexity of page, code written for that particular page and total time taken by page for execution.

8).Internet & database connection- Rank of a page is also depend on its connection with database. If connection of database is not in proper working form it causes reduction in rank score.

9).Server side &client side validation- We also assign rank score base on the time taken in validation. Validation is performed on server as well as client side. So if for a web page validation on both sides is less than its rank score will be high.

IV. Experiment And Result

A. Experimental Evaluation

We have used the asp.net framework for calculation of page rank using PRABUBB in which mainly two modules are used. First one is administrator who is responsible for accepting request for page rank and assigning this task to Page index incharge. Second one is the user of system. In our experiment user can be of two type one is external user, who is requesting rank of any page and other one is internal user. Internal user performs the ranking operation assign by administrator. The decision tree for our project is given below.

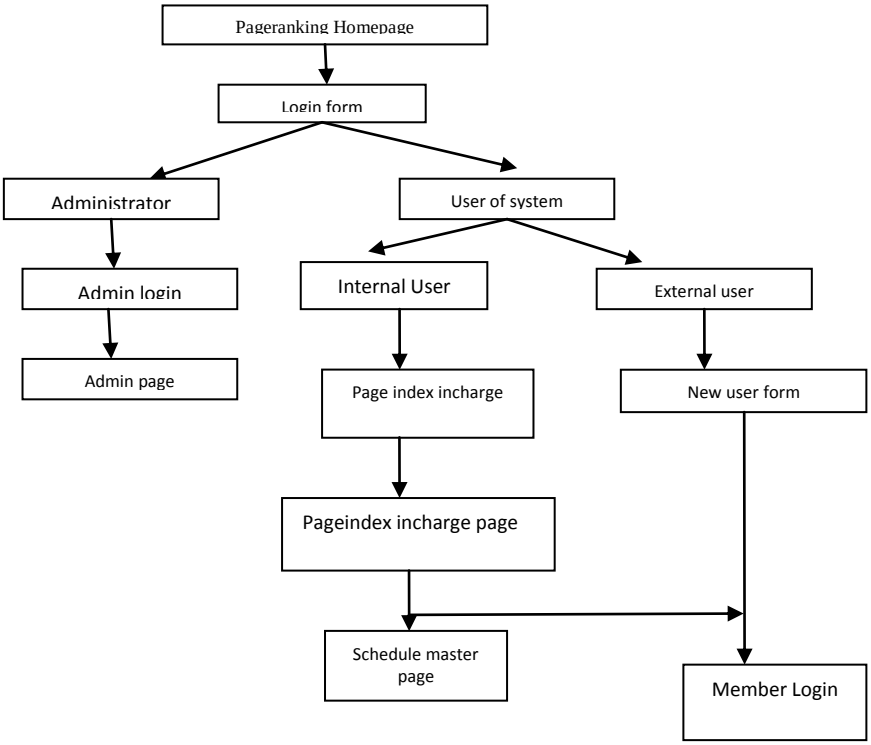

Figure 2. PRBUBB Design Specification

B. Result

For calculating the rank score of a page we use the parameter defined in section III (B) Page index incharger uses that parameter and assign value to each factor. After assigning value to each factor it calculates the total pagerank score of web page. Page index incharge check the requested page carefully and assign them score individually depending on their functionality. By the help of screen shot given below we are trying to explain that how the rank of a page is calculated. For this experiment we have consider certain values as an indicator for the quality search. If value is 10 it means search result is best. , if 9 then better, and if 8 it is good

Figure 3. Snapshot of the Report Generation page

Figure 4. Snapshot showing the search result

From the figure it is clear that how page index incharge calculate the score. In the given chart below colored block represents the score of each parameter and last two blocks represents gained pagerank score with respect to total pagerank score.

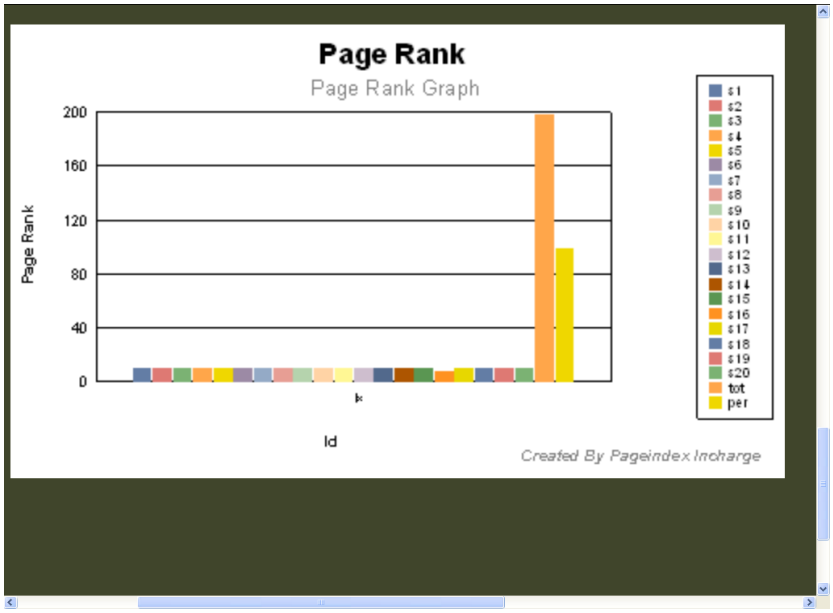

Fig 5.Snapshot of PageRank Graph

V. Conclusion

Web mining is used to extract information from users' past behavior. Web structure mining plays an important role in this approach. Two commonly used algorithms in web structure mining are HITS and PageRank, which are used to rank the relevant pages. Both algorithms treat all links equally when distributing rank scores. Several algorithms have been developed to improve the performance of these methods. This paper uses the asp.net framework for calculation of page rank using PRABUBB.

References

[1] Sandeep Gupta et al. "A Quantitative Approach to Perk up Navigation Efficiency" in proceedings of 1st International Conference On Management Of Technologies & Information security (ICMIS 2010), held at IIIT, Allahabad, India, Jan 21st - 24th, 2010, ISBN No. 978-81-8329-375-4, pp-378-386.

[2] S. Brin and L. Page. The anatomy of a large-scale hyper textual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998

[3] 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.

[4] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. In *SODA '98*, pages 668–677, Philadelphia, PA, USA, 1998. Society for Industrial and Applied Mathematics.

[5] Sandeep Gupta et al, / "An Extended Algorithm of Page Ranking Considering Chronological Dimension of Search "(IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (4) , 2011, 1478-1483.ISSN:0975-9646

[6] Sandeep Gupta et al. "Recuperating Website Link Structure Using Fuzzy Relations between the Content and Web Pages", in International Journal International Journal of Computer Applications, Vol. 1, 2010, URL http://www.ijcaonline.org/archives/number11/245-402.

[7] B. Amen to, L. Terveen, and W. Hill. Does authority mean quality? Predicting expert quality ratings of web documents. In *SIGIR' 00*. ACM, 2000.

[8] T. Haveliwala. E_cient computation of pagerank. Technical Report, Stanford University, 1999.

[9] F. McSherry. A uniform approach to accelerated pagerank computation. In *WWW '05*, pages 575–582, New York, USA, 2005. ACM.

[10] T. H. Haveliwala. Topic-sensitive pagerank. In *WWW ' 02*, Honolulu, Hawaii, May 2002.

[11] M. Richardson and P. Domingos. The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. In Advances in Neural Information Processing *Systems 14*. MIT Press, 2002.

[12] T. Haveliwala, S. Kamvar, and G. Jeh. An analytical comparison of approaches to personalizing pagerank. Technical Report, Stanford University, 2003.

[13] P. Boldi, M. Santini, and S. Vigna. Pagerank as a function of the damping factor. In *WWW ' 05*. ACM, 2005.

[14] A. N. Langville and C. D. Meyer. Deeper inside pagerank. *Internet Mathematics*, 1(3):335–400, 2004.

[15] M. Bianchini, M. Gori, and F. Scarselli. Inside pagerank. *ACM Trans. Interet Technol.*, 5(1):92–128, 2005

[16] .R. W. White, M. Bilenko, and S. Cucerzan. Studying the use of popular destinations to enhance web search interaction. In *SIGIR '07*, pages 159–166, New York, USA, 2007. ACM

