ISSN No. 0976-5697

Volume 4, No. 5, May 2013 (Special Issue)

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

An Efficient Approach for Mining Web Links Based on Priority

K.Swathi, M.Meghana, T.Subba Reddy
M.Tech, Computer Science & Engineering, Vignan's Lara Institute Of Technology & Science
Vadlamudi,Guntur,(A.P)
mandapatimeghana@gmail.com

Abstract— A New Page Rank have been proposed to rank the results of a search system based on a user's topic or query. This paper introduces a concept search based on rating given by the user and the user relevance. New algorithms are presented that utilize web page categories to rank the search results. Web structure mining plays an effective role in this approach. Some page ranking algorithms like PageRank, Weighted PageRank are commonly used for web structure mining. To yield more accurate search results respects to a particular topic, we propose a new concept of taking the user performance rating and user's view of the page by taking his rating and by taking these two ratings we provide the weightage of that link. Web structure mining that will show the relevancy of the pages of a given topic is better determined, as compared to the existing PageRank, Topic sensitive PageRank and Weighted PageRank algorithms. For ordinary keyword search queries, our concept will satisfy the topic of the query.

Keywords— Web structure mining; PageRank Algorithm; Effective Rating Concept; Priority Algorithm

I. INTRODUCTION

TODAY, the World Wide Web is the popular and interactive medium to disseminate information. The Web is huge, diverse and dynamic. The Web contains vast amount of information and provides an access to it at any place at any time. The most of the people use the internet for retrieving information. But most of the time, they gets lots of insignificant and irrelevant document even after navigating several links. For retrieving information from the Web, Web mining techniques are used.

Web Mining Overview

Web mining is an application of the data mining techniques to automatically discover and extract knowledge from the Web. According to Kosala et al [3], Web mining consists of the following tasks:

Resource finding: the task of retrieving intended Web documents.

Information selection and pre-processing: automatically selecting and pre-processing specific information from retrieved Web resources.

Generalization: automatically discovers general patterns at individual Web sites as well as across multiple sites.

Analysis: validation and/or interpretation of the mined patterns.

There are three areas of Web mining according to the usage of the Web data used as input in the data mining process, namely, Web Content Mining (WCM), Web Usage Mining (WUM) and Web Structure Mining(WSM).

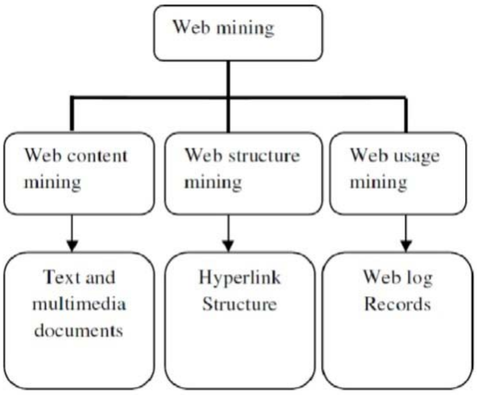

Fig.1 Web Mining Classification

Web content usage mining, Web structure mining, and Web content mining. Web usage mining refers to the discovery of user access patterns from Web usage logs. Web structure mining tries to discover useful knowledge from the structure of hyperlinks which helps to investigate the node and connection structure of web sites. According the type of web structural data, web structure mining can be divided into two kinds 1)extracting the documents from hyperlinks in the web 2) analysis of the tree-like structure of page structure. Based on the topology of the hyperlinks, web structure mining will categorize the web page and generate the information, such as the similarity and mining is concerned with the retrieval of information from WWW into more structured form and indexing the information to retrieve it quickly. Web usage mining is the process of identifying the browsing patterns by analyzing the user's navigational behavior. Web structure mining is to discover the model underlying the link structures of the Web pages, catalog them and generate information such as the similarity and relationship between them, taking advantage of their hyperlink topology. Web classification is shown in Fig 1.

Web Content Mining (WCM)

Web Content Mining is the process of extracting useful information from the contents of web documents. The web documents may consists of text, images, audio, video or structured records like tables and lists. Mining can be applied on the web documents as well the results pages produced from a search engine. There are two types of approach in content mining called agent based approach and database based approach. The agent based approach concentrate on searching relevant information using the characteristics of a particular domain to interpret and organize the collected information. The database approach is used for retrieving the semi-structure data from the web.

Web Usage Mining (WUM)

Web Usage Mining is the process of extracting useful information from the secondary data derived from the interactions of the user while surfing on the Web. It extracts data stored in server access logs, referrer logs, agent logs, client-side cookies, user profile and meta data.

CONFERENCE PAPER
"National Conference on Networks and Soft Computing"
On 26-27 Feb 2013
Organized by
Vignan University, India

Web Structure Mining (WSM)

The goal of the Web Structure Mining is to generate the structural summary about the Web site and Web page. It tries to discover the link structure of the hyperlinks at the inter-document level. Based on the topology of the hyperlinks, Web Structure mining will categorize the Web pages and generate the information like similarity and relationship between different Web sites. This type of mining can be performed at the document level (intra-page) or at the hyperlink level (inter-page). It is important to understand the Web data structure for Information Retrieval.

II. RELATED WORK

A. *PageRank*

Brin and Page developed *PageRank* algorithm during their Ph D at Stanford University based on the citation analysis. *PageRank* algorithm is used by the famous search engine, Google. They applied the citation analysis in Web search by treating the incoming links as citations to the Web pages. However, by simply applying the citation analysis techniques to the diverse set of Web documents did not result in efficient outcomes. Therefore, *PageRank* provides a more advanced way to compute the importance or relevance of a Web page than simply counting the number of pages that are linking to it (called as "back links").

If a back link comes from an "important" page, then that back link is given a higher weighting than those back links comes from non-important pages. In a simple way, link from one page to another page may be considered as a vote. However, not only the number of votes a page receives is considered important, but the "importance" or the "relevance" of the ones that cast these votes as well.

Assume any arbitrary page *A* has pages *T1* to *Tn* pointing to it (incoming link). PageRank can be calculated by the following.

$$PR(A)=(1-d)+d(PR(T1)\ /C(T1) + ...+ PR(Tn\ /C(Tn\ ))\quad (1)$$

The parameter *d* is a damping factor, usually sets it to 0.85 (to stop the other pages having too much influence, this total vote is "damped down" by multiplying it by 0.85). *C(A)* is defined as the number of links going out of page *A*. The *PageRanks* form a probability distribution over the Web pages, so the sum of all Web pages' *PageRank* will be one. *PageRank* can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the Web.

B. *Weighted PageRank*

Wenpu Xing and Ali Ghorbani [1] proposed a *Weighted PageRank* (*WPR*) algorithm which is an extension of the *PageRank* algorithm. This algorithm assigns a larger rank values to the more important pages rather than dividing the rank value of a page evenly among its outgoing linked pages. Each outgoing link gets a value proportional to its importance.

The importance is assigned in terms of weight values to the incoming and outgoing links and are denoted as $^{W\ in}{}_{(m,n)}$ and $^{W\ out}{}_{(m,n)}$ respectively. $^{W\ in}{}_{(m,n)}$ as shown in (2) is the weight of *link*(*m*, *n*) calculated based on the number of incoming links of page *n* and the number of incoming links of all reference pages of page *m*.

$$W_{(m,n)}{}^{in} = \frac{In}{\sum_{p\in R(m)} Ip}\quad \ldots (2)$$

$$WOut_{(m,n)} = \frac{On}{\sum_{P\in R(m)} Op}\qquad \ldots (3)$$

Where and *Ip* are the number of incoming links of page *n* and page *p* respectively. *R*(*m*) denotes the reference page list of

page *m*. $W^{\ out}$ is as shown in (3) is the weight of *link*(*m*, *n*)
(*m*,*n*)

calculated based on the number of outgoing links of page *n* and the number of outgoing links of all reference pages of *m*. Where *On* and *Op* are the number of outgoing links of page *n* and *p* respectively. The formula as proposed by Wenpu et al for the *WPR* is as shown in (4) which is a modification of the *PageRank* formula.

WPR(n) = (1-d)+d	∑	**WPR(m)W^{in}**	**W^{out}**
	m€B(n)	**(m, n)**	**(m,n)**

C. *Topic Sensitive PageRank*

In Topic Sensitive PageRank, several scores are computed: multiple importance scores for each page under several topics that form a composite PageRank score for those pages matching the query. During the offline crawling process, 16 topic-sensitive PageRank vectors are generated, using as a guideline the top-level category from Open Directory Project (ODP). At query time, the similarity of the query is compared to each of these vectors or topics; and subsequently, instead of using a single global ranking vector, the linear combination of the topic-sensitive vectors is weighed using the similarity of the query to the topics. This method yields a very accurate set of results relevant to the context of the particular query.

III. PROPOSED METHODOLOGY

Our topic prioritizing the web links uses the already defined page rank algorithm. which provides a more advanced way to compute the relevance of a web page than simply counting the no. of pages that are linking to it. In addition to this page rank algorithm our concept utilizes the effective rated page rank and priority algorithm.

Background:

Effective rating taken from norm of the two rated values .

Effective Rating(ER)=Norm(AR+Rr(i to n))

Norm: average of two numbers.

Ar: rating from alert box given by user.

Rr: rating got by user performance which will be identified by the system. the system takes the rating in two modes; reading and waiting modes.

Reading mode:

Whenever the user starts reading the system identifies the scroll bar movement. If the scroll bar moves step by step the rating bar which we have given at the ending rises the rating bar .If the user just vents the entire page without reading the rating bar indicates in red color which leaves the rating value null. If the scroll bar just checks the half of the page then the Rating bar indicate some mean color which represents some rating (suppose rating as 5 which indicates the page is used optimally)

Rr=Rr++;

Waiting mode:

If the user kept the page idle the rating bar takes the value depending up on the scroll bar position. If the page is kept idle for long time and vents the page, the rating value takes 0.

The performance identified from system value=Read/Waiting rate.

User Rating:

Whenever the user vents the page it provides an alert box which requests the rating from user by norming these two ratings we can find the effective rating of the page.

$ER(A)=NORM(AR(A)+\sum_{i \text{ to } n}Rr(Ti))$

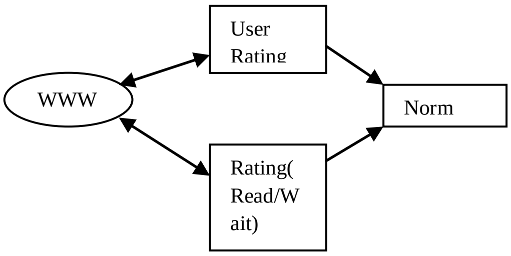

Fig: Norming Effective Rating

PRIORITY ALGORITHM:

input: PR(A)+ER(A)

Ordering: Determine, without looking at P, a total ordering of all possible page links

```
while not empty (P)
   next:= index of page link in P that comes first in the order
Decision: Decides how to prioritize page Links
  Overall Priority Rating of pages
```

(OPRP) = (PR (A) +ER (A))

By using this priority algorithm we get the priority for our required pages.

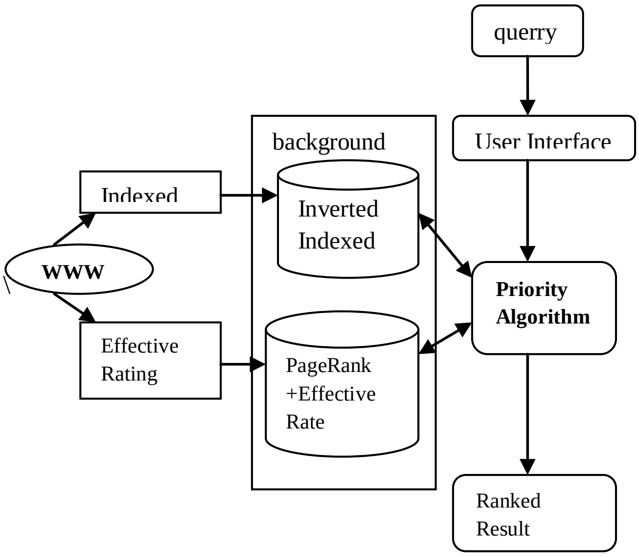

Fig: Proposed System Architecture.

IV. CONCLUSIONS

In this investigation, we proposed a new concept based on Effective Rating Page Rank and Priority Algorithm for web page ranking. Out this approach is based on the PageRank algorithm, and provides a scalable approach for search rankings using Link analysis. For each Web page, compute an importance score per topic. The Effective Rating can be used to produce a final rank for the result pages with respect to the query. This algorithm will improve the order of web pages in the result list so that user may get the relevant pages easily.

V. REFERENCES

[1]. Shesh Narayan Mishra*,Alka Jaiswal,Asha Ambhaikar ,Dept. Comp. Sci & Engg., RCET, April 2012 ISSN: 2277 128X.

[2]. http://pr.efactory.de/e-pagerank-algorithm.shtml 2002/2003 eFactory GmbH & Co. KG Internet-Agentur - written by Markus Sobek.

[3]. N. Duhan, A. K. Sharma and K. K. Bhatia, “Page Ranking Algorithms:A Survey, *Proceedings of the IEEE International Conference on Advance Computing*, 2009.

[4]. A. M. Zareh Bidoki and N. Yazdani, “DistanceRank: An intelligent ranking algorithm for web pages” *Information Processing and Management*, Vol 44, No. 2, pp. 877-892, 2008.

[5]. S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, “Mining the Link Structure of the World Wide Web”, *IEEE Computer Society Press*, Vol 32, Issue 8 pp. 60 – 67, 1999.

[6]. L. Page, S. Brin, R. Motwani, and T. Winograd, “The Pagerank Citation Ranking: Bringing order to the Web”. -*Libraries* SIDL-WP-1999-0120, 1999.

[7]. J. Hou and Y. Zhang, “Effectively Finding Relevant Web Pages from Linkage Information”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 4, 2003.

[8]. J. Dean and M. Henzinger, “Finding Related Pages in the World Wide Web”, *Proc. Eight Int’l World Wide Web Conf.*, pp. 389-401, 1999.

[9]. http://stackoverflow.com/questions/5659272/how-to-get-scroll-event-and-count-the-number-of-scroll-events.

[10]. https://developers.google.com/mobile/articles/webapp_fixed_ui

[11]. http://stackoverflow.com/questions/12725687/hide-navigation-on-scroll-down

[12]. http://www.w3.org/TR/WCAG20-TECHS/SCR33.html