ISSN No. 0976-5697

Volume 3, No. 3, May-June 2012

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

An Integrated Approach of Topic-Sensitive PageRank and Weighted PageRank for Web Mining

Shesh Narayan Mishra*
Department of Computer Science and Engineering
RCET, CSVTU
Bhilai, C.G., India
shesh07.narayan@gmail.com

Alka Jaiswal
Department of Computer Science and Engineering
RCET, CSVTU
Bhilai, C.G., India
alkajaiswal0@gmail.com

Asha Ambhaikar
Department of Computer Science and Engineering
RCET, CSVTU
Bhilai, C.G., India
Asha31.a@rediffmail.com

Abstract: The www sources are increases tremendously. User's surfing the web to fetch the relevant and accurate information of their query. Our proposed approach plays an important and effective role towards this direction. A new PageRank algorithm have been proposed for web structure mining to rank the search result based on the user's topics. Our approach is based on some set of basic category which is top ODP category. A new algorithm is presented to yield accurate search results in respect to user's query. Our approach shows the high relevant pages in top of few links and this links are better determined as compared to existing PageRank algorithm.

Keywords: Web structure mining; PageRank; Weighted PageRank; Topic sensitive PageRank

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 al [1], 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) [2][3].

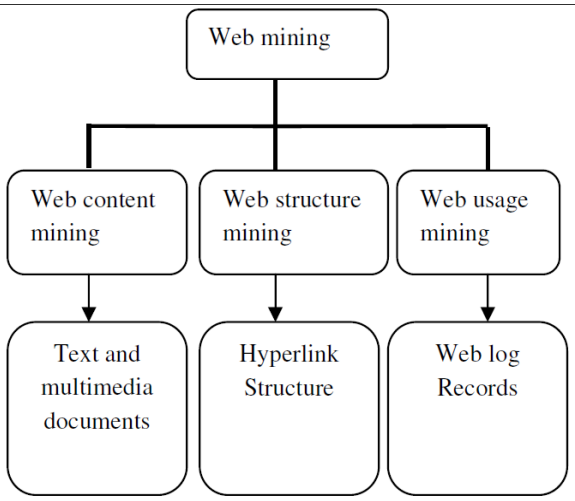

Figure 1. Web Mining Classification

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.

A. *Web Content Mining*

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.

B. *Web Usage Mining*

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.

C. *Web Structure Mining*

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 [4]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 [5][6].

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 [7] 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 equation (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*.

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

$$W^{out}_{(m,n)} = \frac{On}{\sum_{p \in R(m)} Op} \quad (3)$$

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

$$WPR(n) = (1-d) + d \sum_{m \in B(n)} WPR(m) W^{in}_{(m,n)} W^{out}_{(m,n)} \quad (4)$$

C. *Topic Sensitive PageRank*

In Topic Sensitive PageRank [8][9], 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. METHODOLOGY

Our technique Topic sensitive weighted page rank makes use of a subset of the ODP category structure that is associated with individual information needs. This subset is processed locally, aiming at enhancing generic results offered by search engines. Processing entails introducing importance weights to those parts of the ODP structure that correspond to user-specific preferences. The results of local processing are subsequently combined with global knowledge to derive an aggregate ranking of web results. In the following subsection we describe in more detail the theoretical model used and the algorithmic steps deployed

A. *Background*

Let G = (V,E) be a graph[10] denoting the Web where V is a set of vertices representing the set of all Web pages and E contains a direct edge < p, q > if page p links to page q. We number the pages (and their corresponding vertices in V) with labels from 1 to n where n is the number of total page in the Web.

Let A be the web matrix corresponding to theWeb graph G where $A_{ij} = \frac{1}{|O(i)|}$ if page i links to page j and |O(i)| is the number of all outgoing links from page i and $B_{ij} = \frac{1}{|I(i)|}$ if page i links to page j and |I(i)| is the number of all outgoing links from page i.

Let v and u be two vectors each containing n entries, an entry for each page. We refer to v as the rank vector and to u as the customization vector.

Then the Weighted PageRank equation is:

$$v = v(\propto AB + (1-\propto)u) \qquad (5)$$

Where $\propto \in (0,1)$ and is also known as the damping factor as it controls how much weight is given to the Web link matrix, A, versus the weight given to the personalization vector, u. In our experiments we vary to test the performance of WPR under various levels. Search engines typically set to a value of 0.85. Higher values will result in typical iterative methods used to solve for v to consume too many iterations and make the formula too sensitive to minor changes in A and u.

We refer to u as the customization vector and when u is the uniform distribution vector u = [1/n, ..., 1/n] then the resulting solution vector, v is the global WPR vector.

When u is non-uniform, then the resulting rank vector v is the local WPR vector. We consider a topic-biased depending on how we randomly construct the customization vector, u to simulate topic biasing.

B. *Offline Methodology Roadmap*

In our approach, the first step is to generate a biased weighted pagerank vectors using a set of some basis topics. This step is the pre-processing step of the web crawler. This step is performed offline. We select some topics from freely available Open Directory Project as dmoz.

Let Tj be the set of URLs in the ODP category c_j . Then we will computes the Weighted PageRank vector v_j for topic c_j where

$$v_{ji} = \begin{cases} \frac{1}{|Tj|}, & i \in Tj \\ 0, & i \notin Tj \end{cases} \qquad (6)$$

The Weighted PageRank vector for topic c_j is given WPR (α,v_j) where α is bias factor.

We also compute the some class term vectors D_j consisteing of the term in document below each of the top-level categories. D_{it} gives the total number of occurrences of term t in documents listed below class c_j.

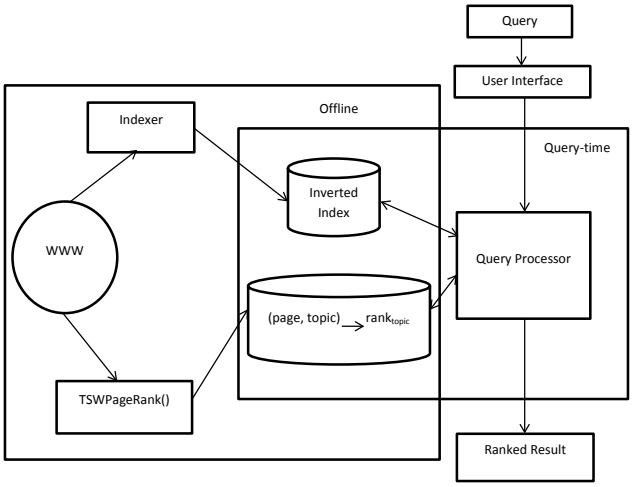

Figure 2. Proposed System Architecture

C. *Compute Importance Score at Query Time*

The second step of our approach will be performed at the time of query. User will provide a query q, let q' be the context of q. In other words, if the query was issued by highlighting the term q in some Web page u, then q' consists of the terms in u. Alternatively , we could use only those terms in u nearby the highlighted term, as often times a single Web page may discuss a variety of topics. For ordinary queries not done in context, let q' = q. Using a unigram language model, with parameters set to their maximum-likelihood estimates, we compute the class probabilities for top-level ODP classes, conditioned on q'. Let q' be the i th term in the query q'. Then given the query q, we compute for each cj the following:

$$P(c_j/q') = \frac{P(c_j).P(q'/c_j)}{P(q')} \propto P(c_j).\prod_i P(q_i'/c_j) \qquad (7)$$

$P((q_i')/c_j)$ is easily computed from the class term-vector Dj . The quantity $P(c_j)$ is not as straight forward. We chose to make it uniform, although we could personalize the query results for different users by varying this distribution. In other words, for some user k, we can use a prior distribution $P_k(c_j)$ that reflects the interests of user k. Using a text index, we retrieve URLs for all documents containing the original query terms q. Finally, we compute the query-sensitive importance score of each of these retrieved URLs as follows.

Let $rank_{jd}$ be the rank of document d given by the rank vector $\overrightarrow{WPR}(\alpha, \vec{v}_j)$ (i.e., the rank vector for topic c_j). For the Web document d, we compute the query-sensitive importance score s_{qd} as follows.

$$s_{qd} = \sum_j P\left(\frac{c_j}{q'}\right).rank_{jd} \quad (8)$$

The results are ranked according to this composite scoress_{qd}.

The above query-sensitive Weighted PageRank computation has the following probabilistic interpretation, in terms of the "random surfer" model . Let w_j be the coefficient used to weight the jth rank vector, with $\sum_j w_j = 1$ (e.g. $w_j = P(c_j/q)$). Then note that the equality holds.

$$\sum_j [w_j \overrightarrow{WPR}(\alpha, \vec{v}_j)] = \overrightarrow{WPR}(\alpha, \sum_j [w_j \vec{v}_j]) \quad (9)$$

Thus we see that the following random walk on theWeb yields the topic-sensitive score s_{qd}. With probability $1-\alpha$, a random surfer on page u follows an outlink of u (where the particular outlink is chosen uniformly at random). With probability $\alpha P(c_j/q')$, the surfer instead jumps to one of the pages in T_j (where the particular page in T_j is chosen uniformly at random). The long term visit probability that the surfer is at page v is exactly given by the composite score s_{qd} defined above. Thus, topics exert inuence over the final score in proportion to their affinity with the query (or query context).

IV. Experimental Setup & Result

In order to integrate topic-sensitive PageRank and Weighted PageRank into a search engine, we must:

1. Decide the topics for which we shall create PageRank vectors.
2. Compute the topic-sensitive Weighted PageRank vector for that topic.
3. Find a way of determining the topic or set of topics that are most relevant for a particular search query.
4. Use the Topic Sensitive Weighted PageRank vectors for that topic or topics in the ordering of the responses to the search query.

If we examine the entire Web, or a large, random sample of the Web, we can get the background frequency of each word. Suppose we then go to a large sample of pages known to be about a certain topic, say the pages classified under sports by the Open Directory. Examine the frequencies of words in the sports sample, and identify the words that appear significantly more frequently in the sports sample than in the background. In making this judgment, we must be careful to avoid some extremely rare word that appears in the sports sample with relatively higher frequency. This word is probably a misspelling that happened to appear only in one or a few of the sports pages. Thus, we probably want to put a floor on the number of times a word appears, before it can be considered characteristic of a topic.

Table 1: Queries Used

Cycling
Death valley
Computer vision
Java
Hockey
Cancer
Seahawk
Gulf war
Web Design
Orthopedics
Hiv

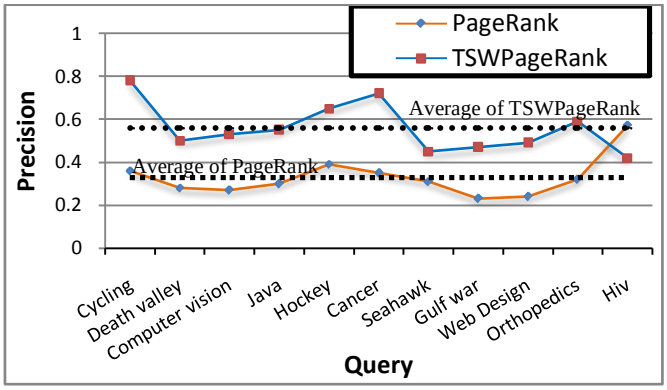

Figure 3. Example of a One-Column figure caption.

In this Figure 3, the PageRank algorithm for Web Information Retrieval is compared with the proposed Topic-Sensitive Weighted PageRank algorithm for Web Information retrieval. Figure 3 shows the result of ten queries. The average of both the algorithm shows the accuracy is improved in the proposed system than the existing system.

V. Conclusion And Future Work

In this investigation, we proposed a new concept based on Topic-Sensitive PageRank and Weighted PageRank 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. At query time, these importance scores are combined based on the topics of the query and associated context to form a composite PageRank score for those pages matching the query. This score can be used in conjunction with other scoring schemes 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.

In future we endeavor to design an architecture which can continuously update and refresh the web information and update the repository periodically.

VI. References

[1] R. Kosala, H. Blockeel, "Web Mining Research: A Survey", SIGKDD Explorations, Newsletter of the ACM Special Interest Group on Knowledge Discovery and Data Mining Vol. 2, No. 1 pp 1-15, 2000.

[2] N. Duhan, A. K. Sharma and K. K. Bhatia, "Page Ranking Algorithms:A Survey", Advanced Computing Conference, 2009, IACC 2009, IEEE International pp. 1530-1537.

[3] M. G. da Gomes Jr. and Z.Gong, "Web Structure Mining: An Introduction", The IEEE International Conference on Information Acquisition, 2005 pp. 6.

[4] S. Brin, and L. Page, The Anatomy of a Large Scale Hypertextual Web Search Engine, Computer Network and ISDN Systems, Vol. 30, Issue 1-7, pp. 107-117, 1998.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd, "The Pagerank Citation Ranking: Bringing order to the Web".

Technical Report, Stanford Digital Libraries SIDL-WP-1999-0120, 1999

[6] M. Bianchini, M.. Gori and F. Scarselli, "Inside PageRank". ACM Transactions on Internet Technology, Vol. 5, Issue 1, 2005 pp. 92-128.

[7] W. Xing and Ali Ghorbani, "Weighted PageRank Algorithm", Proc. ofthe Second Annual Conference on Communication Networks and Services Research (CNSR '04), IEEE, 2004 pp. 305-314.

[8] Taher H. Haveliwal, "Topic-Sensitive PageRank",Proceedings of the Eleventh International World Wide Web Conference, May 2002 pp 517-526.

[9] Taher H. Haveliwala. "Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search". IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No4, July/August 2003, 784-796.

[10] A. Broder, R. Kumar, F Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, J. Wiener, "Graph Structure in the Web", Computer Networks: The International Journal of Computer and telecommunications Networking, Vol. 33, Issue 1-6, pp 309-320, 2000.