Volume 2, Issue 4, April 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Effective Algorithm for Web Mining Based on Topic Sensitive Link Analysis

Shesh Narayan Mishra*,Alka Jaiswal,Asha Ambhaikar
Dept. Comp. Sci & Engg., RCET
shesh07.narayan@gmail.com

Abstract—**A New PageRank have been proposed to rank the results of a search system based on a user's topic or query. This paper introduces a concept towards this direction; search based on ranking of some set of categories that comprise a user search profile. New algorithms are presented that utilize web page categories to search results. Web structure mining plays an effective role in this approach. Some page ranking algorithms PageRank, Weighted PageRank are commonly used for web structure mining. The original PageRank algorithm search-query results independent of any particular search query. To yield more accurate search results respects to a particular topic, we propose a new algorithm Topic sensitive weighted page rank based on 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, Topic Sensitive Weigted PageRank scores will satisfy the topic of the query.**

Keywords— **Web structure mining; Weighted PageRank; Topic sensitive PageRank; TSWPR**

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).

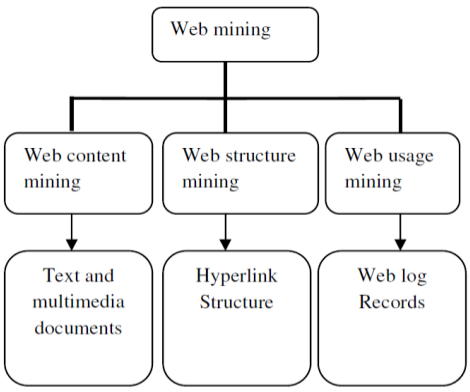

Fig.1 Web Mining Classification

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)

© 2012, IJARCSSE All Rights Reserved

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.

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*.

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

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

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 (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

© 2012, IJARCSSE All Rights Reserved

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, 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 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*

Our proposed techniques is the following. Consider an arbitrary search engine uses a graphs structure G(V,E) of categories, in order to categorize web pages. Graph G consists of nodes $v \in V$ that denote categories and every edge$(v_i, v_j \in E)$denotes that v_j is a subcategory of v_i and is assigned a weight$d(v_i v_j \in [0,1])$. It assumed that every web page is tagged with a specific category.

Overall, the proposed approach introduces the idea of incrementally selecting subgraph G_{sub} of G. This subgraph can be constructed according to set of some basic topic choosen from ODP. In the extreme case $G_{sub} \equiv G$. Every category v of G'will be assigned a relevance-importance weight $\beta(v) > 0$. These weights are used in order to categorize pages returned to the end user, when posing a query. In particular, the position (rank) of a page pin the result-set of an arbitrary user query will be given by a function of the form: $\emptyset(\beta(\gamma(p)), \sigma(p)))$. In the above function, $\gamma(p)$ is the category that a page p belongs to, $\sigma(p))$ is the relevance-importance accordingto the ranking algorithm of the engine, and function $\emptyset()$ indicates how the final ranking will be biased towards machine ranking or category importance defined (for example, $\emptyset(\propto, \beta) = (\alpha + \beta)/2)$. In general, wehave introduced the function $\emptyset()$ that combines search engine ranking and our proposed ranking techniques in order to provide better scalability of our solution's.

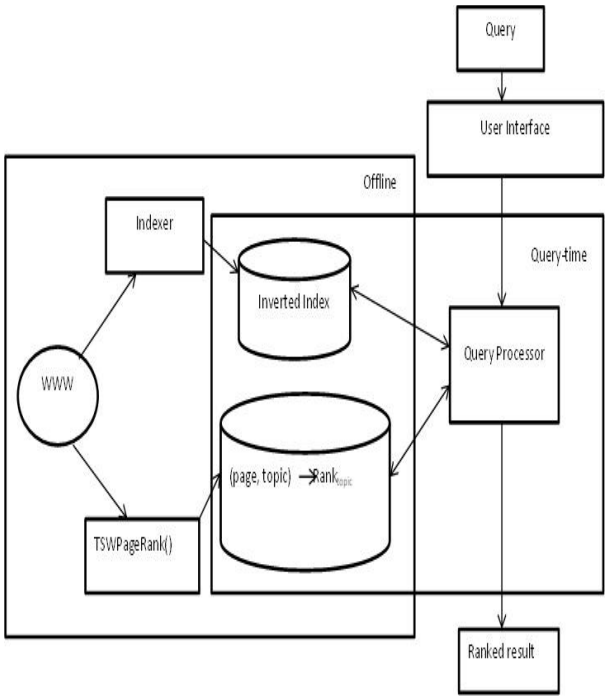

Fig 2: Proposed System Architecture

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 these 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

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

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

We also computes the some class term vectors D_j consisteing of the term in document below each of the top-level categories.D_{jt} simply gives the

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

© 2012, IJARCSSE All Rights Reserved

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 each of the 16 top-level ODP classes, conditioned on q'. Let q' be the i th term in the query (or query context) q'. Then given the query q, we compute for each cj the following:

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

$P((q_i')/c_j)$ 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.

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

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

The above query-sensitive Weighted PageRank computation has the following probabilistic interpretation, in terms of the "random surfer" model [26]. 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

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

holds, as shown in Appendix A. 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. CONCLUSIONS

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.

.

REFERENCES

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

[2] 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.

[3] 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.

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

[5] M. G. da Gomes Jr. and Z.Gong, "Web Structure Mining: An Introduction", *Proceedings of the IEEE International Conference onInformation Acquisition*, 2005.

[6] 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.

[7] X. Wang, T. Tao, J. T. Sun, A. Shakery and C. Zhai, "DirichletRank: Solving the Zero-One Gap Problem of PageRank". *ACM Transaction on Information Systems*, Vol. 26, Issue 2, 2008.

[8] Z. Gyongyi and H. Garcia-Molina, "Web Spam Taxonomy". *Proc. of the First International Workshop on Adversarial Information Retrieval on the Web",* 2005.

[9] M. Bianchini, M.. Gori and F. Scarselli, "Inside PageRank". *ACM Transactions on Internet Technology*, Vol. 5, Issue 1, 2005

[10] C.. H. Q. Ding, X. He, P. Husbands, H. Zha and H. D. Simon, "PageRank: HITS and a Unified Framework for Link Analysis". *Proc. of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2002.

[11] J. Cho and S. Roy, "Impact of Search Engines on Page Popularity". *Proc. of the 13th International Conference on WWW*, pp. 20-29, 2004.

[12] J. Cho, S. Roy and R. E. Adams, "Page Quality: In search of an unbiased web ranking". *Proc. of ACM International Conference on Management of Data"*. Pp. 551-562, 2005.

[13] 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.

[14] 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.

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

© 2012, IJARCSSE All Rights Reserved

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

[16] S. Brin, 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.

[17] 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.

[18] 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.

[19] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tompkins and E. Upfal, "Web as a Graph", *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Database systems*, 2000.

[20] R. Cooley, B. Mobasher and J. Srivastava, "Web Minig: Information and Pattern Discovery on the World Wide Web". *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence*, pp. (ICTAI'97), 1997.

[21] Sung Jin Kim and Sang Ho Lee, "An Improved Computation of the PageRank Algorithm", In proceedings of the European Conference on Information Retrieval (ECIR), 2002.

[22] Ricardo Baeza-Yates and Emilio Davis ,"Web page ranking using link attributes" , In proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, PP.328-329, 2004.

[23] H Jiang et al., "TIMERANK: A Method of Improving Ranking Scores by Visited Time", In proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kunming, 12-15 July 2008.

[24] 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.

[25] Lian-Wang Lee, Jung-Yi Jiang, ChunDer Wu, Shie-Jue Lee, "A Query-Dependent Ranking Approach for Search Engines", Second International Workshop on Computer Science and Engineering, Vol. 1, PP. 259-263, 2009.

[26] NL Bhamidipati et al., "Comparing Scores Intended for Ranking", In IEEE Transactions on Knowledge and Data Engineering, 2009.

[27] Su Cheng,Pan YunTao,Yuan JunPeng,Guo Hong,Yu ZhengLu and Hu ZhiYu "PageRank, "HITS and Impact Factor for Journal Ranking", In proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering – Vol. 06, PP. 285-290, 2009.

© 2012, IJARCSSE All Rights Reserved