Ranking of Web Documents for Domain Specific Database

Ginni Aggarwal
Department of Computer Science & Engineering
M.I.E.T
Meerut

Mukesh Rawat, PhD
Department of Computer Science & Engineering
M.I.E.T
Meerut

ABSTRACT
Now a days, search engines are been most widely used for extracting information from various resources throughout the world. This paper proposed an idea for ranking of web documents offline by mapping the search query terms and the keywords coming in the documents. This paper proposes a new and efficient methodology for indexing of web documents. This technique provide relevant results to the user according to their query. This paper provide better result in retrieving related documents after removing the cue words and frequent used words so, the time will be reduced for finding the appropriate document.

General Terms
Information Retrieval

Keywords
Ranking, query term, relevant.

1. INTRODUCTION
Ranking is an annual performance evaluation method that grades documents on a simple best-to-worst scale to develop a quality work force. Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers). IR can also cover other kinds of data and information problems beyond that specified in the core definition above. The term "unstructured data" refers to data which does not have clear, semantically overt, easy-for-a-computer structure. Web indexing refers to various methods for indexing the contents of a website or of the internet as a whole. Individual websites may use a back-of-the-book index, while search engines usually use keywords to provide a more useful vocabulary for Internet.

Advantage of ranking is that it quickly identifies top performances. With the increasing number of web pages and users on the web, the number of queries Submitted the search engines is also increasing rapidly. Therefore, the search engine needs to be more efficient in its process. The search engines become very popular if they use efficient ranking mechanism. If the search results are not displayed according to the user interest then the search engine will lose its popularity. So the ranking algorithms become very important. Some of the ranking algorithms are Page Rank [PR], Weighted Page Rank (WPR) and Distance Rank.

2. LITERATURE SURVEY
2.1 Page Rank Algorithm
Page Rank 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.

2.2 Weighted Page Rank Algorithm
Weighted Page Rank (WPR) algorithm which is an extension of the Page Rank 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 Win(m, n) and Wout(m, n) respectively. Win(m, n) 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.

$$W^{in}_{(m,n)} = \frac{I_n}{\sum_{p \in R(m)} I_p}$$

$$W^{out}_{(m,n)} = \frac{O_n}{\sum_{p \in R(m)} O_p}$$

Where In 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. Wout(m, n) 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.

2.3 Distance Rank Algorithm
Distance Rank is used to compute the ranks of web pages. The number of average clicks between two pages is defined as distance. The main objective of this algorithm is to minimize distance so that a page with smaller distance to have a higher rank.

Most of the current ranking algorithms have the rich-get richer problem i.e. 45the popular high rank web pages become more and more popular and the young high quality pages are not picked by the ranking algorithms. There are many solutions suggested for this rich-get-richer problem.

The Distance Rank algorithm algorithms is less sensitive to the rich-get-richer problem and finds important pages faster than others. This algorithm is based on the reinforcement learning such that the distance between pages is treated as punishment factor. Normally related pages are linked to each other so the distance based solution can find pages with high qualities more quickly.

2.4 Subsequent Pages

For pages other than the first page, start at the top of the page, and continue in double-column format. The two columns on the last page should be as close to equal length as possible.

3. PROPOSED METHODOLOGY

Description of different models used in proposed methodology explained with the help of the Figure 1 given below:

The system developed is a text based search engine which is capable of extracting the documents. Inputted text processing is initialized at this step which takes in the search text which is analyzed for keywords. Document retrieval is based on the occurrence of terminologies and keywords based on the user search text. Calculate ranking of documents is based on the values of the Document Found.

3.1 Preprocessing

It is done by the following steps-

3.1.1 Removal of Cue Words

Cue words are like is, are, am, was, were, had, it, there etc. The reason behind for removing the cue words from the documents is to save the time during the processing time. When the user enter the keyword in search text box then the system starts processing to search the keyword in the document by ignoring all the cue words, this will help in fast searching.

3.1.2 Removal of Frequent used Words

These are the words which are repeating more than once in the document. This is done because during the searching process, system does not access the same word at the same time. After removing these frequently used words, less time will be consumed while searching.

3.2 Pure Terms

These are the keywords which we get after the preprocessing like A, B, C.

Example- suppose a user enter a keyword like Denial Of Service, then system starts its processing to find these keywords without accessing cue words and frequent used words.

So this A, B, C are the keywords which user wants to search.

3.3 Document

These are the documents which are extracted from database based on extracting keywords and terminologies from the documents and making a comparison. If match found then the document are listed with matched keyword.

Suppose we have documents A1, A2, B1, B2, C1, C2

3.4 Document Found

A1 → A, C

A2 → None

B1 → None

B2 → C

C1 → A, B, C

A B C are the keywords which are found in the documents after searching and preprocessing.

Like A1 document contain A and C keywords, Documents A2 and B1 do not contain any keyword, B2 contain C keyword, C1 contain all the three keywords A B C.

3.5 Ranking Table

This table contains Documents and their values.

Example- value of A1 is 2, value of A2 and B1 is Null, value of B2 is 1and value of C1 is 3.

3.6 Result

All the documents will be shown that contain these keywords. Result is display according to the document priority. The document which have highest priority will display first and vice versa.

Example-document C1 is displaying first in the result table because it has higher rank as compare to other documents, then A1 is displaying because its value is 2. In the end, B2 is displayed because its rank is lowest. A2 and B1 are not displaying because there is no value of them.

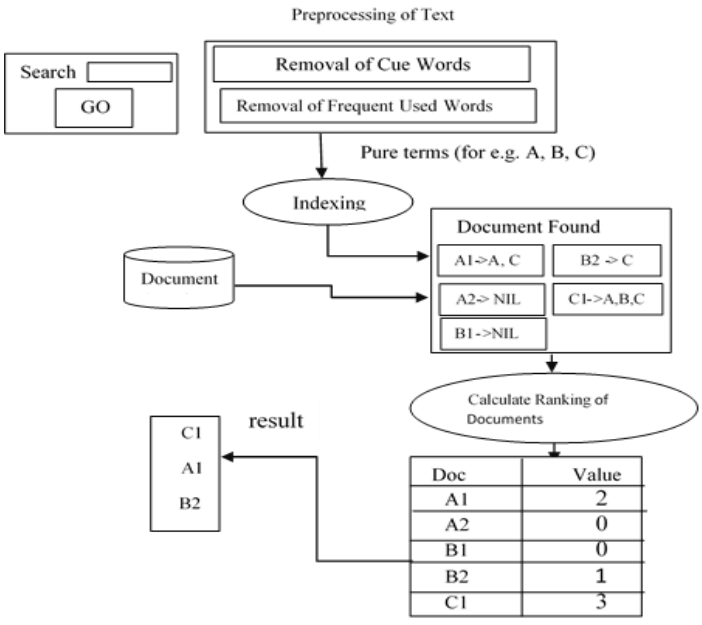


Figure 1: General Architecture of Proposed Methodology

4. ANALYSIS

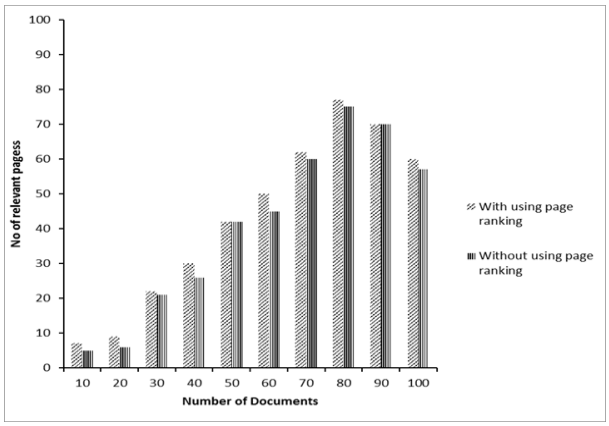
In the below Table 1, there are some query terms. Based on these terms, the no of pages containing maximum no of query terms using Page ranking algorithm and the no of documents from where result will be fetched are calculated. Also it showing the simulation of Page ranking algorithm and the performance of Page ranking.

For example- Query Term IEEE found in 10 documents, 7 pages containing max no of query terms by using by using page ranking and 5 pages without using page ranking algorithm.

Table 1. Table captions should be placed above the table

Query Terms	No of Documents	with using Page Ranking	Without using Page Ranking
		No of pages containing max. no of Query terms	
IEEE	10	7	5
Deadlock	20	9	6
Operating System	30	22	21
Operating System	40	30	26
Computer Network	50	42	42
Ethernet	60	50	45
DBMS	70	62	60
Information Technology	80	77	75
Mutual Exclusion	90	70	70
Wireless LAN	100	60	57

Graph shows that by using Page ranking algorithm, better result will be found as compare to without using Page ranking algorithm.



5. REFERENCES

[1] Jayanthi Manicassamy et al /International Journal on Computer Science and EngineeringVol.1(2),2009,111-115

[2] R. Agrawal, C. Aggarwal, and V. V. V. Prasad. A tree projection algorithm for generation of frequent item sets. Journal of Parallel and Distributed Computing, 61(3):350–371, 2001.

[3] M. Ankerst, M. Breunig, H. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure. In 1999 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'99), pages 49–60, Philadelphia, PA, June 1999.

[4] F. Beil, M. Ester, and X. Xu. Frequent term-based text clustering. In Proc. 8th Int. Conf. on Knowledge Discovery and Data Mining (KDD)'2002, Edmonton, Alberta, Canada, 2002.http://www.cs.sfu.ca/˜ ester/publications.html.

[5] S. Chakrabarti. Data mining for hypertext: A tutorial survey. SIGKDD Explorations: Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining, ACM, 1:1–11, 2000.

[6] P. Domingos and G. Hulten. Mining high-speed data streams. In Knowledge Discovery and Data Mining, pages 71–80, 2000.

[7] R. C. Dubes and A. K. Jain. Algorithms for Clustering Data. Prentice Hall College Div, Englewood Cli®s, NJ, March 1998.

[8] S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data streams. In IEEE Symposium on Foundations of Computer Science, pages 359–366, 2000.

[9] R. Agrawal, C. Aggarwal, and V. V. V. Prasad. Depth-first generation of large item sets for association rules. Technical Report RC21538, IBM Technical Report, October 1999.

[10] M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental clustering and dynamic information retrieval. In Proceedings of the 29th Symposium on Theory of Computing STOC 1997, pages 626–635, 1997.