AN APPROACH
TO
USER RELEVANCY RANKING ON WEB

A Dissertation Submitted in the Partial Fulfillment for the Award of

MASTER OF TECHNOLOGY

IN

SOFTWARE ENGINEERING

Submitted To

Ms. Abhilasha Sharma

Submitted By

Sonal Tuteja

2K11/SWE/15

Department of Software Engineering

Delhi Technological University

New Delhi

2012-2013

DECLARATION

I hereby declare that the thesis entitled ***An Approach to User Relevancy Ranking on Web*** which is being submitted to the *Delhi Technological University*, in partial fulfillment of the requirements for the award of degree of *Master of Technology in Software Engineering* is an authentic work carried out by me.

Sonal Tuteja(2K11/SWE/15)
Department of Software Engineering
Delhi Technological University
Delhi.

CERTIFICATE

DELHI TECHNOLOGICAL UNIVERSITY

(Govt. of National Capital Territory of Delhi)

BAWANA ROAD, DELHI-110042

Date: _____

This is to certify that the thesis entitled **An Approach to User Relevancy Ranking on Web** submitted by *Sonal Tuteja (Roll Number: 2K11/SWE/1*5), in partial fulfillment of the requirements for the award of degree of Master of Technology in Software Engineering, is an authentic work carried out by her under my guidance.

Ms. Abhilasha Sharma

Assistant Professor

Department of Software Engineering

Delhi Technological University

Delhi

ACKNOWLEDGEMENT

I take this opportunity to express my deepest gratitude and appreciation to all those who have helped me directly or indirectly towards the successful completion of this thesis.

Foremost, I would like to express my sincere gratitude to my guide *Ms. Abhilasha Sharma, Assistant Professor, Department of Software Engineering, Delhi Technological University, Delhi* whose benevolent guidance, constant support, encouragement and valuable suggestions throughout the course of my work helped me successfully complete this thesis. Without her continuous support and interest, this thesis would not have been the same as presented here.

Besides my guide, I would like to thank the entire teaching and non-teaching staff in the Department of Software Engineering, DTU for all their help during my course of work.

ABSTRACT

There are billions of web pages available on the World Wide Web (WWW). So there are lots of search results corresponding to a user's query out of which only some are relevant. The relevancy of a web page is calculated by search engines using page ranking algorithms. Most of the page ranking algorithm use web structure mining and web content mining to calculate the relevancy of a web page. In this thesis, we provide an extension to standard Weighted PageRank algorithm by combining web structure mining with web usage mining. The proposed method takes into account the importance of both the number of visits of inlinks and outlinks of the pages and distributes rank scores based on the popularity of the pages. So, the resultant pages are displayed on the basis of user browsing behavior.

Keywords: *World Wide Web, Search Engine, Web mining, Inlinks, Outlinks.*

List of Figure(s)

List of Table(s)

TABLE OF CONTENTS

CHAPTER 1

AN OVERVIEW

AN OVERVIEW

1.1. INTRODUCTION

The World Wide Web (also referred to as the WWW) is the environment that has gained a huge amount of attention from the society. WWW is not the Internet but it is a system of interlinked hypertext documents accessed via the Internet [21]. Since its evolution in the late 1980's, the WWW has grown by almost 2000% and growing at a very fast pace [11]. So a search engine is designed to search information on the World Wide Web and the search results are generally presented in a list format. The viewed information may consist of web pages, photos, graphs or any other type of files or data. Search engines keep their data real-time by running complex ranking algorithms that go around websites and collect necessary data nonstop [20]. A standard search engine consists of the following components:

Crawler

Crawling is the major task performed by a search engine. A search engine should have a highly scalable crawler because the crowd of web pages is increasing in an exponential rate. A web crawler also known as spider, is a program that browses the WWW to retrieve web pages. It has to constantly monitor whether the web page has changed and refresh the downloaded pages. A page is fresh when it is similar to the page present at the web server. It interacts with several

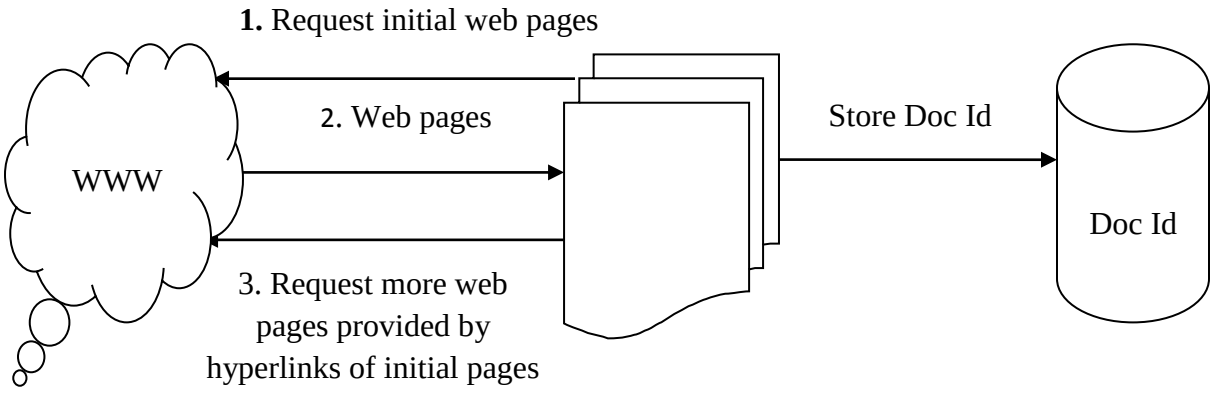

Figure 1.1: Working of a Crawler

web servers and visits the web pages' URLs provided by the URL servers and goes through the hyperlinks given on that page. It adds these URLs in its list and makes its own DNS cache [1]. It also gives unique id to each web page called doc id. Figure 1.1 shows how a crawler works.

Indexer

Indexing is the process of collecting, parsing and storing of the data for the use of searcher module. Indexing associates the keywords to the documents in which keywords are present. Without indexing, search engine would take large amount of time to reply for a query as search engine would have to search all the web pages in the repository at the time of searching. It firstly parses the web pages to extract keywords and generates inverted index. Inverted index consists of keywords and the doc id in which keyword is present. The working of an indexer has been shown in Figure 1.2.

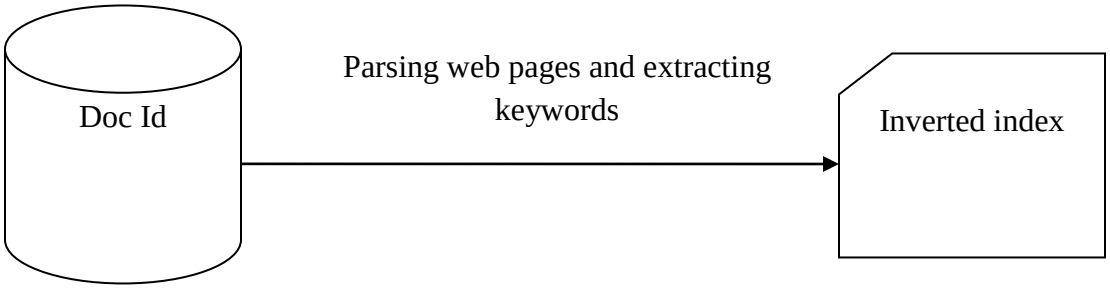

Figure 1.2: Working of an Indexer

Searcher

Searcher gets the query from user and searches the keywords of the query in the inverted index. Then it returns the best matching web pages associated with the query. As the size of web is very large, number of documents retrieved for a particular query is also large. So a search engine has to use some ranking algorithm to prioritize the retrieved web pages. Relevant documents are appeared at the top of the results. Boolean logic operators are used in query to narrow or expand the searches. Logic operator 'OR' is used to expand and 'AND' is used to narrow the results. Figure 1.3 shows the working of searching component.

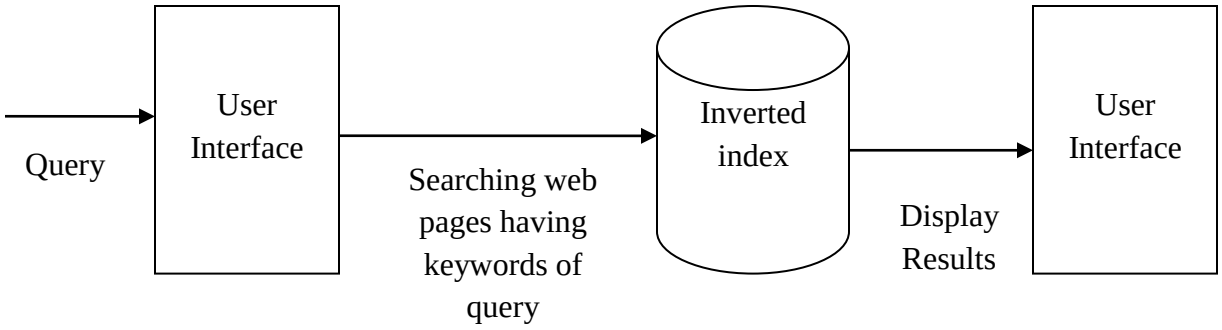

Figure 1.3: Working of a Searcher

Due to enormous amount of data present on the web, the number of web pages returned by a searcher is very large out of which only few are relevant. In a standard search engine, firstly the crawler downloads pages from the WWW. Then the web pages are sent to indexer module that builds the index on the basis of keywords present in the web pages. The query processor module accepts the query from the user and returns the list of web pages which matches the keywords present in the query. But before presenting the resultant web pages to the user, query processor

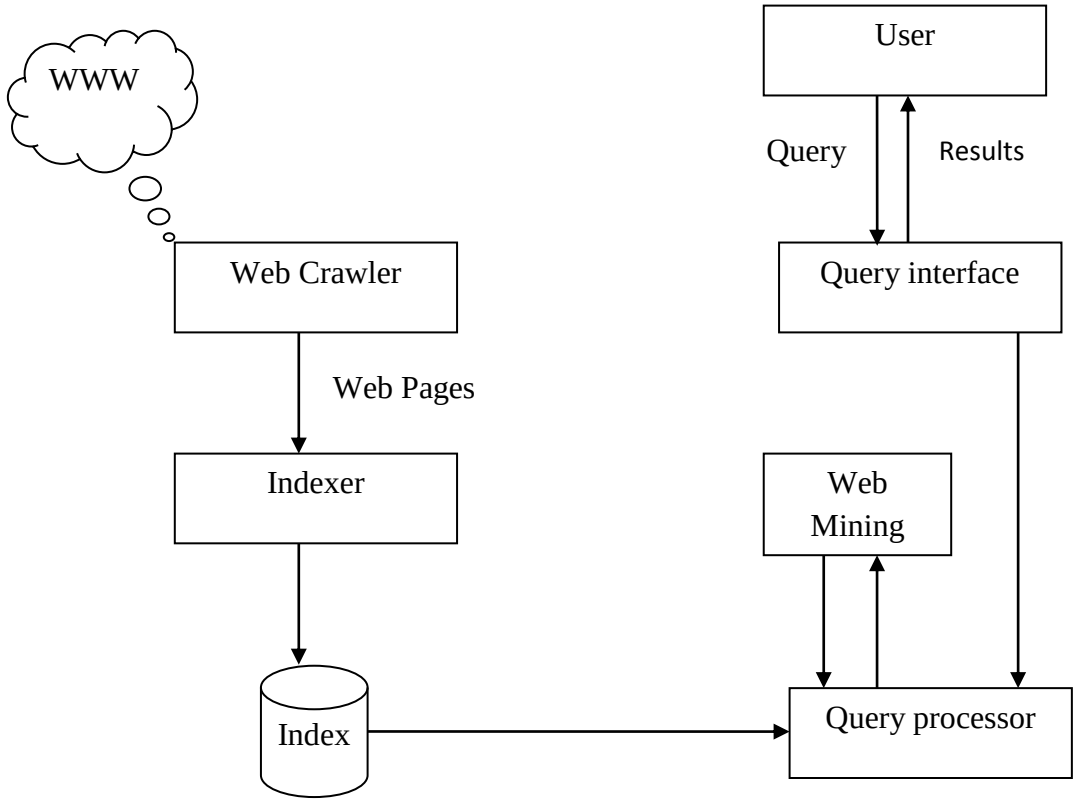

Figure 1.4: Working of a Search Engine [12]

module has to sort the results so that relevant pages are displayed at the top. For sorting the web pages, we need some kind of page ranking algorithms. Web mining plays a vital role in the development of page ranking algorithms. Figure 1.4 shows the working of a standard search engine [12].

The need is to sort the search results leads to the development of first page ranking algorithm called PageRank by Surgey Brin and Larry Page. It uses web graph to calculate the value of page rank of each page. Later on, other variation of page ranking algorithms based on web structure mining and web content mining were also developed like Weighted PageRank, Page Content Ranking, SALSA etc. These ranking algorithms sort the search results returned by a search engine so that most relevant web pages are displayed at the top. These algorithms are based on web structure mining or web content mining or combination of both. But web structure mining only considers link structure of the web and web content mining is not able to cope up with multimedia such as images, mp3 and videos [4]. These algorithms do not take web usage behavior into account. The relevancy of a web page for a user can be determined by how many users click on the link, or recent uses of link or time spent on the link. So the aim is to design an algorithm which takes user relevancy into account.

1.2. MOTIVATION OF THE WORK

Due to enormous amount of data present on the web, information retrieval has been a great issue. It is becoming harder day by day to get the relevant documents in response of a query. This is the reason why we need ranking algorithms to prioritize the search results. Ranking algorithms are driving force behind search engines' working. The first ranking algorithm was introduced by Sergey Brin and Lawrence Page which was based on web structure mining. Later on, other algorithms were developed to rank the web pages such as Weighted PageRank, Page Content Ranking, and HITS etc. Most of the page ranking algorithms are based on web structure mining (WSM) or web content mining (WCM) or combination of both. WSM is extracting information from the structure of the web. A web graph contains nodes representing web pages and links representing hyperlinks between the connected pages. WCM is extracting information from the contents of the web.

The aim of the research work is to first survey the page ranking algorithms and then to propose an algorithm which uses a combination of web structure mining and web usage mining(WUM) to calculate the value of page rank. WUM intends to extract what users are looking on the web. The relevancy of a web page can be estimated by usage trends of web pages. There are different criteria to check the relevancy of web page like frequency of visit of a web page, recent visit of a web page, time spent on a web page etc. The thesis uses number of visit of links (VOL) and combines with WSM to calculate the value of page rank.

1.3. GOALS OF THE THESIS

The overall goal of the thesis is to survey the existing page ranking algorithms and to propose a new ranking algorithm based on web structure mining and web usage mining. The goal of the thesis is:

- To survey the existing page ranking algorithms and discussing their advantages and disadvantages.
- To propose a new ranking algorithm based on web structure mining and web usage mining as web usage mining can help in determining the relevancy of web pages from users' point of view.
- To apply the proposed technique on a web graph to validate the proposed algorithm.

1.4. CONTRIBUTIONS AND GUIDED TOUR OF THE THESIS

The remainder of the thesis is structured as follows:

Chapter 2 discusses the previous work done in the field of page ranking algorithms. This includes the extensive study of various page ranking algorithms that have been proposed in the literature so far. It also highlights some of the most relevant works in the field of work presented in the thesis.

In *Chapter 3,* classification of web mining has been introduced and how it has been used in ranking of web pages. Existing page ranking algorithms have been surveyed with their advantages and disadvantages.

Chapter 4 introduces the proposed algorithm which uses combination of WSM and WCM to calculate the value of page rank. The mathematical expressions have been derived to calculate the value of page rank and the working has been illustrated by an example. It also discusses the benefits of proposed algorithm, its limitations and the improvements that can be made to overcome these limitations.

Chapter 5 analyzes the value of page rank over different values of dampening factor *d* and compares the results with an existing algorithm named Weighted PageRank.

The final section concludes the thesis with a summary of the results, and a discussion on possible future directions along with the references used.

CHAPTER 2

LITERATURE SURVEY

LITERATURE SURVEY

WWW has ample number of hyperlinked documents and these documents contain heterogeneous information including text, image, audio, video, and metadata. So there are lots of search results corresponding to a user's query out of which only some are relevant. It is becoming harder day by day to get the relevant documents in response of a query. Ranking algorithms are the driving force behind search engines' working. The relevancy of a web page is calculated by search engines using page ranking algorithms. Ranking algorithms are required to sort the results so that more relevant documents are displayed at the top.

Brin and Page [1] came up with an idea at Stanford University to use link structure of the web to calculate page rank of web pages. The algorithm was named PageRank after Larry Page (Cofounder of Google Search Engine). PageRank was the first ranking algorithm which was used by Google to prioritize the results produced by keyword based search.

Wenpu Xing and Ali Ghorbani [18] proposed an algorithm called Weighted PageRank algorithm by extending standard PageRank. The working principle behind the algorithm was that an important page has more linkages from other web pages have to it or are linked to by it. Unlike standard PageRank, it did not evenly distribute the page rank of a page among its outgoing linked pages but the page rank of a web page is divided among its outgoing linked pages in proportional to the importance or popularity (its number of inlinks and outlinks). The popularity from the number of inlinks and the popularity from the number of outlinks were used to calculate the values of page rank.

Page Content Ranking (PCR) was developed by Jaroslav Pokorny and Jozef Smizansky [7] which employs web content mining to calculate the value of page rank. The algorithm analyzed the content of web pages by using some heuristics. The importance of a web page was determined on the basis of importance of terms contained in the page and the importance of term was calculated with respect to a query q.

Hyperlink-Induced topic search (HITS) algorithm based on WSM was developed by Kleinberg [3]. It works on the principle that for a given query, there is a set of authority pages that are relevant for a given query and set of hub pages that contain links to relevant pages which

includes links to many authority pages also. The algorithm finds the set of authority pages relevant for a query using sampling.

Gyanendra Kumar et. al. [6] came up with a new idea to incorporate user's broswsing behavior in calculating page rank. Previous algorithms were either based on web structure mining or web content mining but none of them took web usage mining into consideration. A new page ranking algorithm called Page Ranking based on Visits of Links (VOL) was proposed for search engines. It modifies the basic page ranking algorithm by taking into consideration the number of visits of inbound links of web pages. It helps to prioritize the web pages on the basis of user's browsing behavior.

Neelam Tyagi and Simple Sharma [13] incorporated user browsing behavior in Weighted PageRank algorithm to develop a new algorithm called Weighted PageRank based on number of visits of links (VOL). The algorithm assigns more rank to the outgoing links having high VOL .It only considers the popularity from the number of inlinks and ignores the popularity from the number of outlinks which was incorporated in Weighted PageRank algorithm.

CHAPTER 3

PAGE RANKING ALGORITHMS: A SURVEY

PAGE RANKING ALGORITHMS: A SURVEY

3.1. WEB MINING

Data mining can be defined as the process of extracting useful information from large amount of data. The application of data mining techniques to extract relevant information from the web is called as web mining [14] [2]. The process of web mining [2] can be shown in Figure 3.1.

Figure 3.1: Process of Web Mining [2]

The data is retrieved from the web, some data mining and machine learning techniques are applied on the data and useful pattern can be discovered. Web mining can be divided into following three categories [14] as given in Figure 3.2.

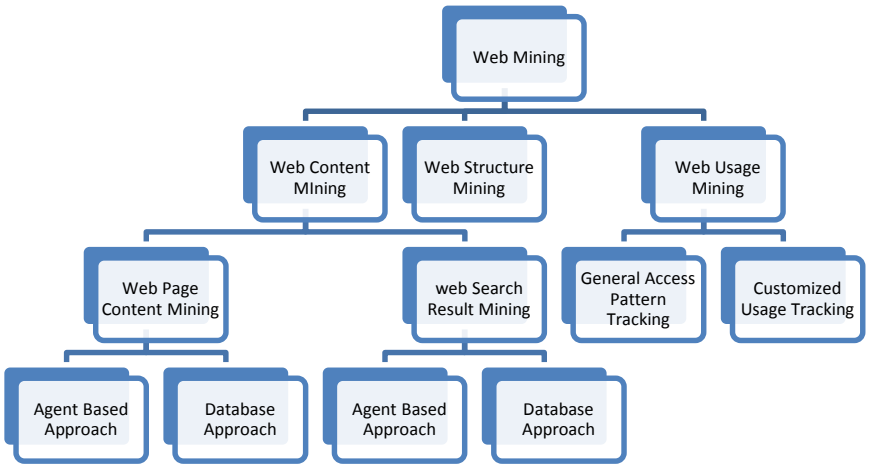

Figure 3.2: Categories of Web Mining [12]

Web Content Mining

Web Content Mining is used to mine the content of web pages. This technique can be applied either on the web pages or on the result pages obtained by the query processor of a search engine

WCM can be differentiated from two different views: Information Retrieval (IR) View and Database (DB) View. IR view works for unstructured and semi-structured data [9]. To represent unstructured data, bag of words is used and HTML structure inside the documents is uses to represent semi-structured data. In DB view, a web site can be transformed to represent a multi-level database and web mining tries to infer the structure of the web site from this database.

Web Structure Mining

WSM [15] can be defined as extracting information from the structure of the web. A web graph contains nodes representing web pages and links representing hyperlinks between the connected pages as shown in Figure 3.3. A hyperlink is a structural unit that is used to move from a location in a web page to a different location, either within the same web (intra-document hyperlink) page or on a different web page (inter-document hyperlink). A hyperlink can be differentiated from two different views:

Inlinks: The links which point to a web page are called as inlinks of that page. e.g.; the links from web pages A, B and C are inlinks of web page P.

Outlinks: The links from a web page to others web page are called outlink of that page. e.g.; the links from web page P to Y and Z are outlinks of page P.

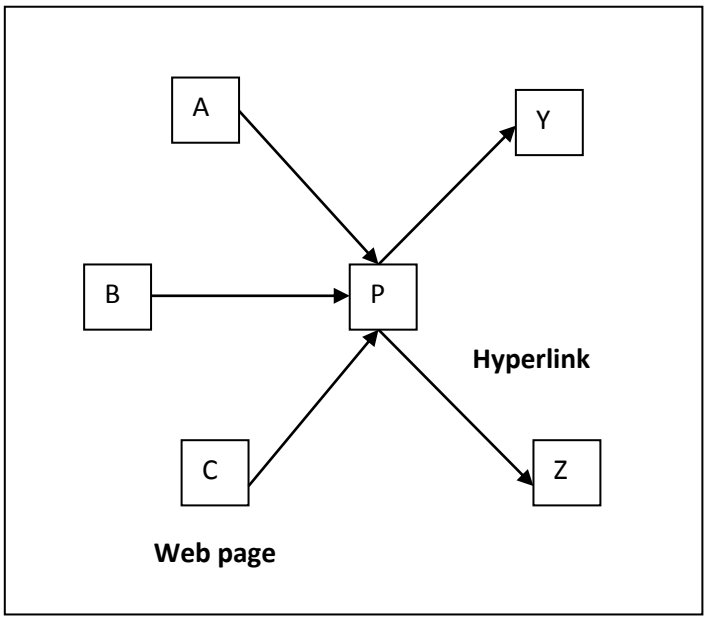

Figure 3.3: Structure of a Web Graph

As the number of inlinks and outlinks of a page are very important parameters in the area of web mining[15], hence importance of web pages is measured by:

- Number of inlinks in PageRank algorithm
- Number of inlinks and number of outlinks in Weighted PageRank algorithm

So WSM is an important area in the field of page ranking algorithm.

Web Usage Mining

Web Usage Mining (WUM) is used to extract information from the server logs which are maintained during interaction with the web. WUM analyze what are the people looking for on the web. Server logs provide information about identity of web users, access time and their browsing behavior [22] which may help in understanding of web based application. The server logs can provide information like

- the frequency of visits per document
- most recent visit per document
- who is visiting which document
- frequency of use of each hyperlink
- most recent use of each hyperlink

which can be used in calculating the value of page rank [8]. It can be further categorized in finding the general access patterns or finding the patterns matching the specified parameters. These web mining techniques are used in page ranking algorithm. In the next section, the page ranking techniques have been explained.

3.2. VARIOUS PAGE RANKING ALGORITHMS

WWW has ample number of hyperlinked documents and these documents contain heterogeneous information including text, image, audio, video, and metadata. So there are lots of search results

corresponding to a user's query out of which only some are relevant. The relevancy of a web page is calculated by search engines using page ranking algorithms. Ranking algorithms are required to sort the results so that more relevant documents are displayed at the top. Various ranking algorithms have been developed such as PageRank, Weighted PageRank, Page Content Ranking, and HITS etc.

3.2.1. PAGERANK ALGORITHM

Brin and Page [1] came up with an idea at Stanford University to use link structure of the web to calculate page rank of web pages. The algorithm was named PageRank after Larry Page (Cofounder of Google Search Engine). The algorithm is used by Google to prioritize the results produced by keyword based search. It works on the principle that if a web page has important

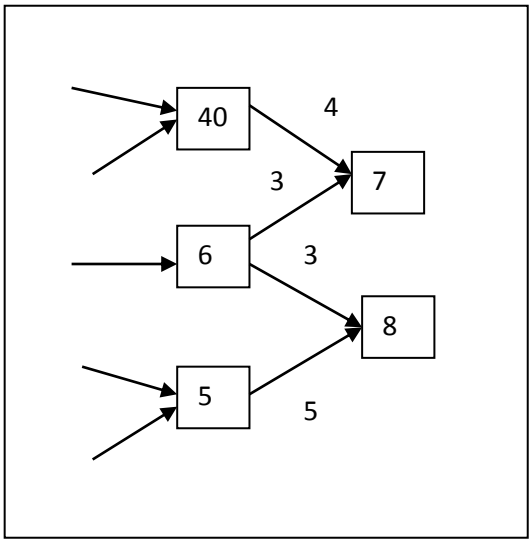

Figure 3.4: Working of PageRank Algorithm

links towards it then the links of this page to other pages are also considered important. PageRank algorithm considers only backlinks into account and propagates the ranking through links. i.e., A web page divides equally its rank to its outgoing links which is illustrated in Figure 3.4. In Figure 3.4, the web page having rank six divides its rank equally to its two outgoing links and the rank of a web page is calculated by summing up the ranks from its incoming links. So the overall rank of a page using PageRank algorithm is calculated by the formula given in equation 3.1:

$$PR(u) = c \sum_{v \varepsilon B(u)} \frac{PR(v)}{N_v} \quad \ldots (3.1)$$

Notations are:

- u represents a web page,
- $PR(u)$ and $PR(v)$ represents the page rank of web pages u and v respectively,
- B(u) is the set of web pages pointing to u,
- N_v represents the total numbers of outlinks of web page v,
- c is a factor used for normalization.

Original PageRank algorithm was modified by taking into consideration that not all users follow direct links on WWW. The modified formula for calculating page rank is given in equation 3.2.

$$PR(u) = (1 - d) + d \sum_{v \varepsilon B(u)} \frac{PR(v)}{N_v} \quad \ldots (3.2)$$

Where *d* is a dampening factor which represents the probability of user using direct links and it can be set between 0 and 1.

Illustration of PageRank algorithm

The working of PageRank algorithm can be illustrated by taking an example shown in Figure 3.5.

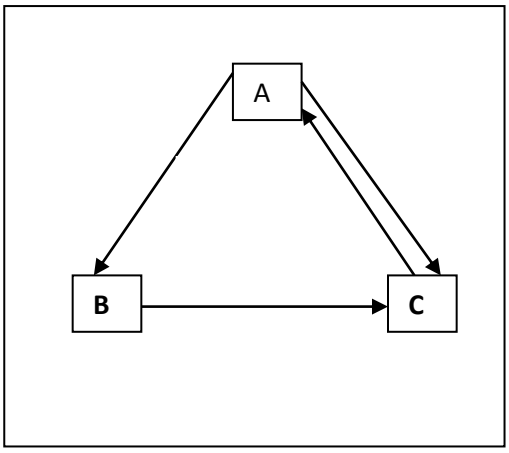

Figure 3.5: A web graph [13]

It has three web pages A, B and C and hyperlinks between them. The page rank of A, B and C can be calculated using equation (3.2) as:

$$PR(A) = (1-d) + d\left(\frac{PR(C)}{1}\right) \quad \ldots (3.2.1)$$

$$PR(B) = (1-d) + d\left(\frac{PR(A)}{2}\right) \quad \ldots (3.2.2)$$

$$PR(C) = (1-d) + d\left(\frac{PR(A)}{2} + PR(B)\right) \quad \ldots (3.2.3)$$

Initially the rank of web page C is considered 1 and value of d is taken as 0.5. Then page ranks are calculated using equation 3.2.1, 3.2.2 and 3.2.3 and the values are A=1, B=0.75000 and C=1.125. These values are used iteratively until the values get stabilized. The intermediate values of A, B and C have been shown in Table 3.1.

A	**B**	**C**
1	0.75000	1.12500
1.06250	0.76563	1.14844
1.07422	0.76855	1.15283
1.07642	0.76910	1.15366

Table 3.1: Values of Page Ranks using PageRank Algorithm

The PageRank algorithm was the first ranking algorithm to use link structure of web to sort the web pages and it helped the users of search engines to find the web pages of their interest. But the method only considered the links of web pages and relevancy of a web page was totally ignored. The presence of query terms in web pages did not affect the rank of web page.

3.2.2. WEIGHTED PAGERANK ALGORITHM

Wenpu Xing and Ali Ghorbani [18] proposed an algorithm called Weighted PageRank algorithm by extending standard PageRank. It works on the principle that if a page is important, more linkages from other web pages have to it or are linked to by it. Unlike standard PageRank, it does not evenly distribute the page rank of a page among its outgoing linked pages. The page rank of a web page is divided among its outgoing linked pages in proportional to the importance or popularity (its number of inlinks and outlinks). The popularity from the number of inlinks and the popularity from the number of outlinks are used to calculate the values of page rank.

$W^{in}(v,u)$, the popularity from the number of inlinks, is calculated based on the number of inlinks of page *u* and the number of inlinks of all reference pages of page *v* as given in equation 3.3.

$$W^{in}(v,u) = \frac{I_u}{\sum_{p \varepsilon R(v)} I_p} \quad \ldots (3.3)$$

Notations are:

- I_u and I_p are the number of inlinks of page *u* and *p* respectively,
- *R(v)* represents the set of web pages pointed by *v*.

$W^{out}(v,u)$, the popularity from the number of outlinks, is calculated based on the number of outlinks of page *u* and the number of outlinks of all reference pages of page *v* as given in equation 3.4.

$$W^{out}(v,u) = \frac{O_u}{\sum_{p \varepsilon R(v)} O_p} \quad \ldots (3.4)$$

Notations are:

- O_u and O_p are the number of outlinks of page *u* and *p* respectively,
- *R(v)* represents the set of web pages pointed by *v*.

The page rank using Weighted PageRank algorithm is calculated by the formula given in equation 3.5.

$$WPR(u) = (1-d) + d \sum_{v \varepsilon \mathrm{B(u)}} WPR(v)\mathrm{W}^{\mathrm{in}}(v,u)\mathrm{W}^{\mathrm{out}}(v,u) \quad \ldots (3.5)$$

Notations are:

- $WPR(u)$ and $WPR(v)$ are page rank of web page u and v respectively,
- $\mathrm{W}^{\mathrm{in}}(v,u)$ represents the popularity from the number of inlinks of web page u,
- $\mathrm{W}^{\mathrm{out}}(v,u)$ represents the popularity from the number of outlinks of web page u,
- B(u) is the set of web pages that point to u,
- *d* is the dampening factor.

Algorithm to calculate WPR

To calculate the value of page rank using WPR, following steps are required [18].

1. *Finding a web site:* A web site with rich hyperlinks is required because the WPR algorithm makes use of the web structure to calculate values of page rank.
2. *Building a web map:* The web map of web site is created in which nodes represent web pages and edges represent hyperlinks between web pages.
3. *Finding the root set:* The root set is created using the IR search engine embedded in web site. This root set contains web pages relevant to a given query.
4. *Finding the base set:* The web pages that directly point to or are pointed by the web pages in root set are put in base set.
5. *Applying algorithms:* The WPR algorithm is applied to the base set.
6. *Evaluating the results:* The algorithm is evaluated by comparing its results with other

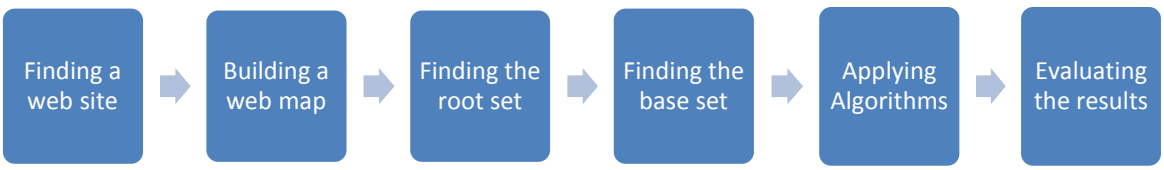

Figure 3.6: Algorithm of Weighted PageRank

algorithms such as Standard PageRank algorithm. The algorithm can be shown as in Figure 3.6.

The working of Weighted PageRank algorithm can be illustrated by taking the example shown in Figure 3.5. It has three web pages A, B and C and hyperlinks between them. The page rank of A, B and C can be calculated using equation 3.5 as:

$$WPR(A) = (1-d) + d\left(WPR(C) * \mathrm{W}^{\mathrm{in}}(C,A) * \mathrm{W}^{\mathrm{out}}(C,A)\right) \quad \ldots (3.5.1)$$

$$WPR(B) = (1-d) + d(\mathrm{WPR(A)} * \mathrm{W}^{\mathrm{in}}(A,B) * \mathrm{W}^{\mathrm{out}}(A,B)) \quad \ldots (3.5.2)$$

$$WPR(C) = (1-d) + d(\mathrm{WPR(A)} * \mathrm{W}^{\mathrm{in}}(A,C) * \mathrm{W}^{\mathrm{out}}(A,C) + \mathrm{WPR(B)} * \mathrm{W}^{\mathrm{in}}(B,C) * \mathrm{W}^{\mathrm{out}}(B,C) \quad \ldots (3.53)$$

The values of $\mathrm{W}^{\mathrm{in}}(v,u)$ and $\mathrm{W}^{\mathrm{out}}(v,u)$ can be calculated using equation 3.3 and 3.4.

$$W^{in}(C,A) = \frac{I_A}{I_A} = \frac{1}{1} = 1$$

$$W^{out}(C,A) = \frac{O_A}{O_A} = \frac{2}{2} = 1$$

$$W^{in}(A,B) = \frac{I_B}{I_B + I_C} = \frac{1}{1+2} = \frac{1}{3}$$

$$W^{out}(A,B) = \frac{O_B}{O_B + O_C} = \frac{1}{1+1} = \frac{1}{2}$$

$$W^{in}(A,C) = \frac{I_C}{I_C + I_B} = \frac{2}{2+1} = \frac{2}{3}$$

$$W^{out}(A,C) = \frac{O_C}{O_C + O_B} = \frac{1}{1+1} = \frac{1}{2}$$

$$W^{in}(B,C) = \frac{I_C}{I_C} = \frac{2}{2} = 1$$

$$W^{out}(B,C) = \frac{O_C}{O_C} = \frac{1}{1} = 1$$

These values are put in equation 3.5.1, 3.5.2 and 3.5.3 and the resultant equations are:

$$WPR(A) = (1-d) + d(WPR(C) * 1 * 1) \quad \ldots (3.5.1)$$

$$WPR(B) = (1-d) + d\left(WPR(A) * \frac{1}{3} * \frac{1}{2}\right) \quad \ldots (3.5.2)$$

$$WPR(C) = (1-d) + d\left(WPR(A) * \frac{1}{3} * \frac{1}{2} + WPR(B) * 1 * 1\right) \quad \ldots (3.5.3)$$

Initially the rank of web page C is considered 1 and value of d is taken as 0.5. Then page ranks are calculated using equation 3.5.1, 3.5.2 and 3.5.3 and the values are A=1, B=0.58333 and C=0.95833. These values are used iteratively until the values get stabilized. The intermediate values of A, B and C have been shown in Table 3.2.

A	B	C
1	0.58333	0.95833
0.97917	0.58160	0.95399
0.97701	0.58142	0.95354
0.97677	0.58142	0.95351

Table 3.2: Values of Page Ranks using Weighted PageRank Algorithm

Unlike PageRank algorithm, Weighted PageRank algorithm used both inlinks as well as outlinks to calculate the values of page rank. The page rank of a web page was not divided equally among its outlinked pages but in proportion to the popularity of outlinked pages. But this method also considered only the links of web pages and relevancy of a web page was totally ignored. The presence of query terms in web pages did not affect the rank of web page.

3.2.3. PAGE CONTENT RANKING

Page Content Ranking (PCR) was developed by Jaroslav Pokorny and Jozef Smizansky [7] which employs web content mining to calculate the value of page rank. The algorithm analyzes the content of web pages by using some heuristics. The importance of a web page is determined

on the basis of importance of terms contained in the page and the importance of term is calculated with respect to a query q.

Suppose for a given query q, a set of R_q pages is returned by a search engine. PCR algorithm classifies the web pages in order of their importance such that more important web pages are displayed at the top. A web page is represented in the similar way as in vector model [17] and frequency of terms in the page is used.

Algorithm of PCR

PCR follows the following steps to calculate page rank:

1. *Term Extraction:* A parser extracts the terms from each web page present in R_q and builds an inverted list [19] which is used in step 4.
2. *Parameters Calculation:* Parameters about the terms such as Term frequency (TF), occurrence position of terms and synonyms are identified.
3. *Term Classification:* Using Parameters Calculation of step 2, the importance of each term is determined. A training set of terms is used for learning of neural network.
4. *Relevance Calculation:* The calculated importance of terms in step 3 is used to determine the page relevance score. The new score of a page is equal to the average importance of terms in P. The algorithm of PCR can be described as shown in Figure 3.7

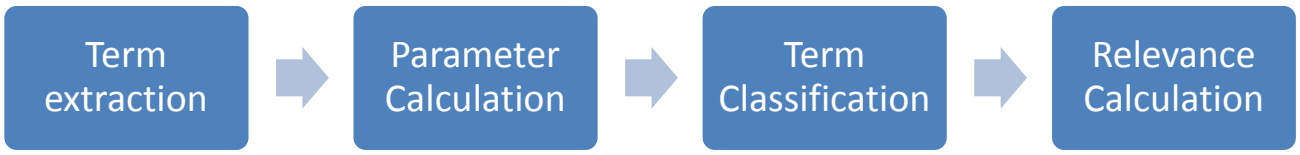

Figure 3.7: Algorithm of PCR

PCR assumes that the importance of a page P is proportional to the importance of all terms in P. The algorithm uses Sum, Min, Max, Average, and Count as aggregation functions.

Symbols used in PCR

These are the symbols used in PCR algorithm to calculate parameters.

D: Set of pages indexed by a search engine.

q: A query fired by a user.

Q: Set of all terms in query q.

$R_q \subseteq D$: Subset of pages indexed by search engine which are considered relevant with respect to q.

$R_{q,n} \subseteq R_q$: Set of n top ranked pages from R_q.

TF(P, t): Number of occurrences of term t in page P. i.e. term frequency.

DF(t): the number of pages which contain the term t. i.e. document frequency.

Pos(P, t): Set of positions of term t in page P.

Term(P, i): The term at i^{th} position in page P.

Parameter calculations in PCR

The importance of a term t is calculated by the formula 5+(2*NEIB) parameters where NEIB denotes the number of neighboring terms included into the calculation. It depends on some attributes like database D, query q and the number n of pages considered. The importance of a term can be described in the terms of parameters given below:

1. *Occurrence frequency:* The total number of occurrences of term t in R_q is called occurrence frequency as given in equation 3.6.

$$freq(t) = \sum_{P \varepsilon R_q} TF(P, t) \quad \ldots (3.6)$$

2. *Incidence of pages*: The occur(t) can be defined as the number of pages containing term t to the total number of pages as given in equation 3.7.

$$occur(t) = \frac{DF(t)}{|R_{q,n}|} \quad \ldots (3.7)$$

3. *Distance of occurrences of t from occurrences of terms in Q***:** If a term t occurs near to the term contained in Q, it is considered significant. QW is the set of all occurrence positions of terms from Q in all pages $P \varepsilon R_{q,n}$ as given in equation 3.8.

$$QW = \cup_{t \varepsilon Q, P \varepsilon R_{q,n}} Pos(P,t) \quad \ldots (3.8)$$

The distance of any term t from the query terms is calculated by the minimum of all distances of t from query term as given in equation 3.9.

$$dist(t) = \min(\{|r-s| : r \, \varepsilon \, Pos(P,t) \; and \; s \, \varepsilon \, QW \quad \ldots (3.9)$$

4. *Frequency in the natural language:* The function $F(t)$ assigns an integer value to all the words to represent its frequency and it can be defined as given in equation 3.10.

$$common(t) = F(t) \quad \ldots (3.10)$$

A term which is more frequent is considered as less important.

5. *Term Importance:* The importance of terms from $R_{q,n}$ can be determined as given in equation 3.11

$$importance(t) = classify(freq(t), dist(t), occur(t), common(t)..) \quad \ldots (3.11)$$

6. *Synonym Classes:* A synonym class database is used and an importance function $SC(S)$ is calculated for each synonym class S as shown in equation 3.12.

$$SC(S) = \sec_{moment(\{importance(t'):t' \varepsilon \, S\})} \quad \ldots (3.12)$$

If a term is synonym of an important term, it also becomes important. The importance is propagated to the term t over all the meaning as given in equation 3.13.

$$synclass(t) = \sec_{moment\left(\left\{SC\left(S_{t'}\right):t' \varepsilon \, SENSE(t)\right\}\right)} \quad \ldots \quad (3.13)$$

Where SENSE(t) contains all the meanings t'of t.

7. *Importance of neighboring term:* If a term has an important term in its neighbor, the term also becomes important. It is described by (2 *NEIB) parameters, that is an aggregation of the importance of terms surrounding the term t. The equation 3.14 given below describes the formula for aggregation of the importance of terms surrounding the term t i.e. RelPosNeib(t, i). The predicate $Inside(P,n)$ is satisfied, if n is an index into the page P.

$$RelPosNeib(t,i) = \bigcup_{P \varepsilon R_{q,n}} \{Term(P, j+1): j \, \varepsilon \, Pos(P,t) \, \varepsilon \, Inside(P, j+1)\} \quad \ldots (3.14)$$

The parameters $Neib(t,i)$ for i = -NEIB, -(NEIB-l), ... -1,1…..NEIB can be calculated as given in equation 3.15.

$$Neib(t,i) = \text{sec}_{moment(RelPosNeib(t,i))} \quad \ldots (3.15)$$

The resultant importance of term t based on all these parameters can be defined as given in equation 3.16.

$$importance(t) = classify(freq(t)dist(t)occur(t)common(t)synclas(t)$$

$$neib(t, NEIB) \ldots . neib(t, NEIB)) \quad \ldots (3.16)$$

Page Classification and Importance Calcualtion

A layered neural network NET is used as a classification tool in PCR and the weights are set from previous experiments. It is assumed that the network has 5+(2*NEIB) neurons in the input and one neuron in the output layer. The input vector v is denoted by NET(v) and excitation of the ith neuron in the output layer of NET after terminating calculation is denoted by NET[9]. The classify() function can be defined as follows as given in equation 3.17:

$$classify(p_1 \ldots . p_{5+(2*NEIB)}) = NET(p_1 \ldots . p_{5+(2*NEIB)}) \quad \ldots (3.17)$$

The importance of a page is calculated by aggregating the importance of all the terms in P as given in equation 3.18

$$Page_{Importance(P)} = \text{sec}_{moment(\{importance(t):t\,\varepsilon\,P\})} \quad \ldots (3.18)$$

The importance of a page calculated by equation 3.18 above imparts ranking to the top n ranked pages according to their content unlike PR and WPR.

3.2.4. HITS ALGORITHM

Hyperlink-Induced topic search (HITS) algorithm based on WSM was developed by Kleinberg [3]. It works on the principle that for a given query, there is a set of authority pages that are relevant for a given query and set of hub pages that contain links to relevant pages which includes links to many authority pages also. The hubs and authorities are shown in Figure 3.8.

The HITS algorithm considers WWW as a graph G(V,E) where V represents set of web pages and E represents hyperlinks between web pages. The search engine cannot retrieve all relevant pages for a given query. So the initial web pages retrieved by a search engine can be used to move further.

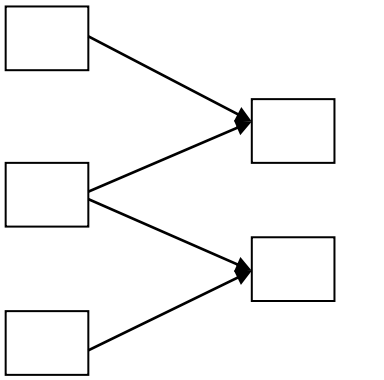

Figure 3.8: Hubs and Authorities in HITS [12]

Working of HITS

The working of HITS can be explained in two steps:

Sampling Step: This step starts with a root set R which are selected from the result list of asearch engine. It determines the base set S using R such that S is small in size, contains relevant pages and strongest authorities. It expands the root set R into base set S using following algorithm

given in [12]. The sampling step of HITS algorithm can be shown in Figure 3.9. Before going to next step, HITS firstly removes all links between web pages of same website because these links are used for navigational purpose not for contributing authority.

Input: Root set R; Output: Base set S

Let S=R

1. For each page pεS, do Steps 3 to 5.
2. Let T be the set of all pages S point to.
3. Let F be the set of all pages that point to S.
4. Let S=S + T + some or all of F.
5. Delete all links with the same domain name.

Figure 3.9: Sampling Step of HITS [12]

Iterative Step (Finding Hubs and Authorities): The hubs and authorities are found using output of sampling step, i.e. base set S. The algorithm for finding hubs and authorities can be given in Figure 3.10. A strong authority is the one which is pointed to by several strong hubs and a strong hub is the one that points to several highly scored authorities. The score of hubs and authorities can be calculated as given in equation 3.19 and 3.20.

$$x_p = \sum_{q \varepsilon B(p)} y_q \quad (3.19)$$

$$y_p = \sum_{q \varepsilon B(p)} x_q \quad (3.20)$$

Where B(p) and R(p) represent the set of referrer and reference pages of page p.

Input: Base Set S, Output: A set of hubs and a set of authorities.

1. Let a page p have a non-negative authority weight x_p and hub weight y_p. Pages with relatively larger weight x_p will be classified to be the authorities, similarly hubs with larger weight y_p.
2. The weights are normalized so the squared sum for each type of weight is 1.
3. For a page p, the value of x_p is updated to be the sum of y_q over all pages q linking to p.
4. The value of y_p is updated to be the sum of x_q over all pages q linked to by p.
5. Continue with step 2 unless a termination condition has been reached.
6. Output the set of pages with largest x_p weights i.e. authorities and those with the largest y_p weights i.e. hubs.

Figure 2.10: Finding Hubs and Authorities

Figure 3.11 shows how hub and authority scores are calculated and these scores are used for ranking of web pages before displaying to the user.

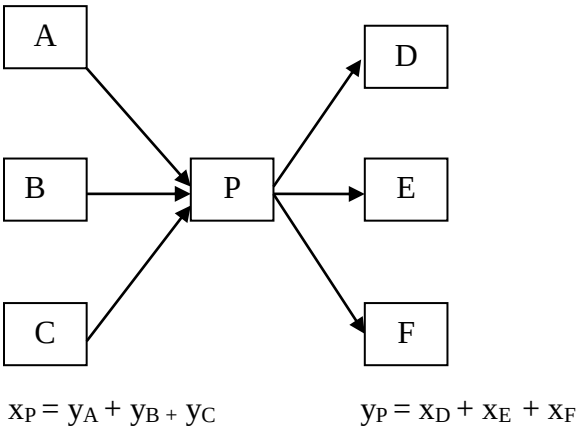

Figure 3.11: Calculation of Hub and Authority Scores [12]

The algorithm uses combination of web structure mining and web content mining to calculate the value of page rank but there are some problems with HITS algorithms. A web site can be a hub as well as an authority. So a distinction between hub and authority is difficult. Some non-relevant results may exist in root set which may lead to unexpected results and the performance

of HITS algorithm is not good in real time. A number of variations like Weighted HITS, Probabilistic HITS, etc. [5, 10, 16] have been proposed in the literature for modifying HITS.

3.2.5. PAGERANK WITH NUMBER OF VISITS OF LINKS

Gyanendra Kumar et. al. [6] came up with a new idea to incorporate user's broswsing behavior in calculating page rank. Previous algorithms were either based on web structure mining or web content mining but none of them took web usage mining into consideration. A new page ranking algorithm called PageRank based on Visits of Links (VOL) was proposed for search engines. It modifies the basic page ranking algorithm by taking into consideration the number of visits of inbound links of web pages. It helps to prioritize the web pages on the basis of user's browsing behavior.

In the original PageRank algorithm, the rank of a page p is evenly distributed among its outgoing links but in this algorithm, rank values are assigned in proportional to the number of visits of links. The more rank value is assigned to the link which is most visited by user. The PageRank based on Visits of Links (VOL) can be calculated by the formula given in equation 3.21.

$$PR(u) = (1-d) + d \sum_{v \epsilon \mathrm{B(u)}} L_u \frac{PR(v)}{TL(v)} \quad \ldots (3.21)$$

Notations are:

- $PR(u)$ and $PR(v)$ represent page rank of web pages u and v respectively,
- *d* is dampening factor,
- *B(u)* is the set of web pages pointing to *u*,
- L_u is number of visits of links pointing from v to u,
- $TL(v)$ is the total number of visits of all links from *v*.

Algorithm for PageRank using Visits of Links

The algorithm to calculate the value of page rank is:

1. *Finding a website:* The website with rich hyperlinks is to be selected because the algorithm depends on the hyper structure of website.

2. *Generating a web graph:* For selected website, web graph a generated in which nodes represent web pages and edges represent hyperlinks between web pages.

3. *Calculating number of visits of hyperlinks:* Client side script is used to monitor the hits of hyperlinks and information is sent to the web server and this information is accessed by crawlers.

4. *Calculate intermediate values:* For each web page u, total number of visits of all links from *u* ($TL(u)$) and the number of visits of links pointing from v to u (L_u), is calculated where $v\varepsilon B(u)$ which is set of web pages pointing to u.

4. *Calculate page rank of each web page:* The intermediate values are substituted in equation 10 to calculate values of page rank.

5. *Repetition of step 5:* The step 5 is used recursively until a stable value of page rank is obtained.

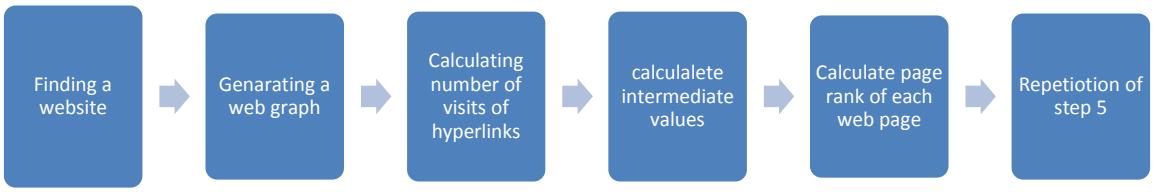

Figure 3.12: Algorithm of PageRank using VOL

Figure 3.12 shown above explains the steps required to calculate page rank using proposed algorithm.

Illustration of PageRank using VOL

The working of PageRank algorithm using visits of links can be illustrated by taking the example shown in Figure 3.13. It has three web pages A, B and C and hyperlinks between them shows

number of visits of links. i.e., how many users have accessed that link. The page rank of A, B and C can be calculated using equation (3.21) as:

$$PR(A) = (1-d) + d\left(\frac{2}{2} * PR(C)\right) \quad \ldots (3.21.1)$$

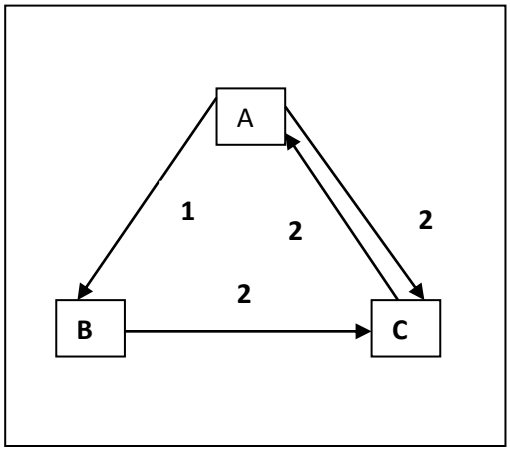

Figure 3.13: A Web Graph with VOL [13]

$$PR(B) = (1-d) + d\left(\frac{1}{3} * PR(A)\right) \quad \ldots (3.21.2)$$

$$PR(C) = (1-d) + d\left(\frac{2}{3} * PR(A) + \frac{2}{2} * PR(B)\right) \quad \ldots (3.21.3)$$

Initially the rank of web page C is considered 1 and value of d is taken as 0.5. Then page ranks are calculated using equation 3.21.1, 3.21.2 and 3.21.3 and the values are A=1, B=0.66667 and C=1.6667. These values are used iteratively until the values get stabilized. The intermediate values of A, B and C have been shown in Table 3.3.

A	B	C
1	0.66667	1.16667
1.08334	0.68056	1.20139
1.10071	0.68345	1.20863
1.10432	0.68405	1.21013

Table 3.3: Values of Page Ranks using PageRank with VOL Algorithm

Unlike PageRank and Weighted PageRank algorithm, PageRank using visits of links makes use of web structure mining and web usage mining to calculate the value of page rank. So the web pages obtained by this algorithm are more relevant to the users as compared to the web pages obtained from PageRank and Weighted PageRank algorithm. In this algorithm too, the presence of query terms in web pages did not affect the rank of web page.

3.2.6. WEIGHTED PAGERANK ALGORITHM USING VISITS OF LINKS

Neelam Tyagi and Simple Sharma [13] incorporated user browsing behavior in Weighted PageRank algorithm to develop a new algorithm called Weighted PageRank based on number of visits of links (VOL). The algorithm assigns more rank to the outgoing links having high VOL .It only considers the popularity from the number of inlinks and ignores the popularity from the number of outlinks which was incorporated in Weighted PageRank algorithm.

In the original Weighted PageRank algorithm, the page rank of a web page is divided among its outgoing linked pages in proportional to the importance or popularity (its number of inlinks and outlinks) but in this algorithm, number of visits of inbound links of web pages are also taken into consideration.

$W^{in}(v,u)$, the popularity from the number of inlinks, is calculated based on the number of inlinks of page u and the number of inlinks of all reference pages of page v as given in equation 3.3.

$$W^{in}(v,u) = \frac{I_u}{\sum_{p \varepsilon R(v)} I_p} \quad \ldots (3.3)$$

Notations are:

- I_u and I_p are the number of inlinks of page *u* and *p* respectively,
- *R(v)* represents the set of web pages pointed by *v*.

The rank of web page using this algorithm can be calculated as given in equation 3.22.

$$WPR_{VOL}(u) = (1-d) + d \sum_{v \varepsilon B(u)} \frac{L_u WPR_{VOL}(v) \mathrm{W}^{\mathrm{in}}(v,u)}{TL(v)} \quad \ldots (3.22)$$

Notations are:

- $WPR_{VOL}(u)$ and $WPR_{VOL}(v)$ represent rank of web pages u and v respectively,
- *d* is the dampening factor,
- *B(u)* is the set of web pages pointing to *u*,
- L_u is number of visits of links pointing from v to u,
- $TL(v)$ is the total number of visits of all links from *v*,
- $\mathrm{W}^{\mathrm{in}}(v,u)$ represents the popularity from the number of inlinks of u.

Algorithm to calculate Weighted PageRank using VOL

The algorithm to calculate the value of page rank is:

1. Finding a website: The website with rich hyperlinks is to be selected because the algorithm depends on the hyper structure of website.

2. Generating a web graph: For selected website, web graph a generated in which nodes represent web pages and edges represent hyperlinks between web pages.

3. Calculating number of visits of hyperlinks: Client side script is used to monitor the hits of hyperlinks and information is sent to the web server and this information is accessed by crawlers.

4. Calculate intermediate values: For each web page u, the value of popularity from the number

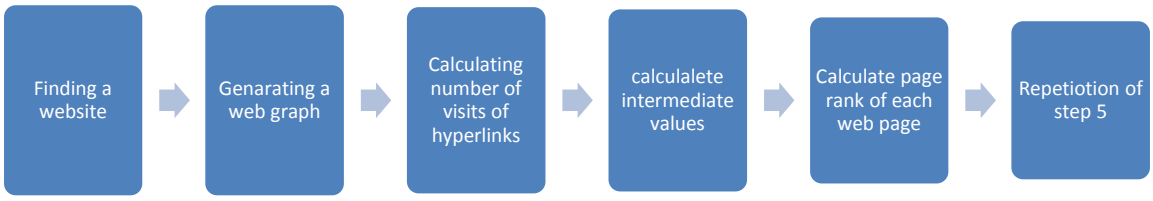

Figure 3.14: Algorithm of WPR_{VOL}

of inlinks ($W^{in}(v,u)$) is calculated by the formula given in equation.

*5. Calculate page rank of each web page***:** The intermediate values are substituted in equation 3.22 to calculate values of page rank.

6. Repetition of step 5: The step 5 is used recursively until a stable value of page rank is obtained.

Figure 3.14 shown below explains the steps required to calculate page rank using $WPR_{VOL}(u)$.

Illustration of Weighed PageRank using VOL

The working of PageRank algorithm using visits of links can be illustrated by taking the example shown in Figure 3.13.

$$WPR_{VOL}(A) = (1-d) + d\left(\frac{2 * WPR_{VOL}(C) * \mathrm{W}^{\mathrm{in}}(C,A)}{2}\right) \quad \ldots (3.22.1)$$

$$WPR_{VOL}(B) = (1-d) + d\left(\frac{1 * WPR_{VOL}(A) * \mathrm{W}^{\mathrm{in}}(A,B)}{3}\right) \quad \ldots (3.22.2)$$

$$WPR_{VOL}(B) = (1-d) + d\left(\frac{2 * WPR_{VOL}(A) * \mathrm{W}^{\mathrm{in}}(A,C)}{3}\right) + d\left(\frac{2 * WPR_{VOL}(B) * \mathrm{W}^{\mathrm{in}}(B,C)}{2}\right) \quad \ldots (3.22.3)$$

The values of $\mathrm{W}^{\mathrm{in}}(v,u)$ can be calculated using equation 3.3.

$$W^{in}(C,A) = \frac{I_A}{I_A} = \frac{1}{1} = 1$$

$$W^{in}(A,B) = \frac{I_B}{I_B + I_C} = \frac{1}{1+2} = \frac{1}{3}$$

$$W^{in}(A,C) = \frac{I_C}{I_C + I_B} = \frac{2}{2+1} = \frac{2}{3}$$

$$W^{in}(B,C) = \frac{I_C}{I_C} = \frac{2}{2} = 1$$

These values are put in equation 3.22.1, 3.22.2 and 3.22.3 and the resultant equations are:

$$WPR_{VOL}(A) = (1-d) + d\left(\frac{2 * WPR_{VOL}(C)}{2} * \frac{1}{1}\right) \quad \ldots (3.22.1)$$

$$WPR_{VOL}(B) = (1-d) + d\left(\frac{1 * WPR_{VOL}(A)}{3} * \frac{1}{3}\right) \quad \ldots (3.22.2)$$

$$WPR_{VOL}(A) = (1-d) + d\left(\frac{2 * WPR_{VOL}(A)}{3} * \frac{2}{3} + \frac{2 * WPR_{VOL}(B)}{2} * \frac{2}{2}\right) \quad \ldots (3.22.3)$$

Initially the rank of web page C is considered 1 and value of d is taken as 0.5. Then page ranks

A	**B**	**C**
1	0.55556	1
1	0.55556	1

Table 3.4: Value of Page Ranks using Weighted PageRank using VOL Algorithm

are calculated using equation 3.22.1, 3.22.2 and 3.22.3 and the values are A=1, B=0.55556 and C=1. These values are used iteratively until the values get stabilized. The intermediate values of A, B and C have been shown in Table 3.4.Weighted PageRank using visits of links makes use of web structure mining and web usage mining to calculate the value of page rank. So the web pages obtained by this algorithm are more relevant to the users as compared to the web pages obtained from PageRank and Weighted PageRank algorithm. In this algorithm too, the presence of query terms in web pages did not affect the rank of web page and it ignores the popularity from the number of outlinks, $W^{out}(v,u)$ which was used in Weighted PageRank algorithm. Table 3.5 gives a brief description of above algorithm using some parameters from [12].

PageRank algorithm and Weighted PageRank algorithms are based on web structure mining only. Page Content Ranking is based on web content mining and HITS is based on web structure mining and web content mining. PageRank algorithm using visits of links and Weighted PageRank algorithm using visits of links are based on combination of web structure mining and web usage mining. PageRank algorithm relies only on the backlinks to calculate the value of page rank. Weighted PageRank algorithm relies on the backlinks and forward links to calculate

the value of page rank. Page Content Ranking algorithm relies on the content to calculate the value of page rank. HITS algorithm relies on backlinks, forward links as well as content to calculate the value of page rank. PageRank using VOL and Weighted PageRank using VOL relies on the backlinks and visits of links to calculate the value of page rank. PageRank and

ALGORITHM	WEB MINING TECHNIQUE USED	INPUT PARAMETERS	IMPORTANCE	RELEVANCE
PageRank	Web structure mining	Backlinks	More	Less
Weighted PageRank	Web structure mining	Backlinks, Forward links	More	Less
Page Content Ranking	Web content mining	Content	Less	More
HITS	Web structure mining, Web content mining	Backlinks, Forward links, Content	Less	More
PageRank with VOL	Web structure mining, Web usage mining	Content	More	More
Weighted PageRank with VOL	Web structure mining, Web usage mining	Backlinks and VOL	More	More

Table 3.5: Comparison of Page Ranking Algorithms

Weighted PageRank algorithms do not consider relevancy of web pages into account. Page Content Ranking and HITS algorithm take content of web pages into consideration. So the relevancy of web pages is improved. PageRank using VOL and Weighted PageRank using VOL

consider relevancy from users' point of view as number of visits of links give the relevancy of a web page.

CHAPTER 4

PROPOSED WORK

PROPOSED WORK

The original Weighted PageRank algorithm distributes the rank of a web page among its outgoing linked pages in proportional to their importance or popularity. The algorithm is purely based on web structure mining and uses web graph to calculate the rank of web pages. $W^{in}(v,u)$, the popularity from the number of inlinks and $W^{out}(v,u)$, the popularity from the number of inlinks do not include usage trends. The rank of web page remains constant whether it has been visited by users or not. It does not give more popularity to the links most visited by the users. i.e.; the relevancy of a web page from user point of view is ignored and the weighted PageRank using VOL makes use of web structure mining and web usage mining to calculate the value of page rank but it neglects the popularity from the number of outlinks i.e., $W^{out}(v,u)$.

In proposed algorithm, $W^{in}_{VOL}(v,u)$, the popularity from the number of visits of inlinks and $W^{out}_{VOL}(v,u)$, the popularity from the number of visits of outlinks are used to calculate the values of page rank.

$W^{in}_{VOL}(v,u)$ is the weight of link(v, u) which is calculated based on the number of visits of inlinks of page *u* and the number of visits of inlinks of all reference pages of page *v* as shown in equation 4.1.

$$W^{in}_{VOL}(v,u) = \frac{I_{u(VOL)}}{\sum_{p \varepsilon R(v)} I_{p(VOL)}} \quad \ldots (4.1)$$

Notations are:

- $I_{u(VOL)}$ and $I_{p(VOL)}$ represent the incoming visits of links of page u and p respectively.
- R(v) represents the set of reference pages of page v.

$W^{out}_{VOL}(v,u)$ is the weight of link(v, u) which is calculated based on the number of visits of outlinks of page *u* and the number of visits of outlinks of all reference pages of page *v* as shown in equation 4.2.

$$W^{out}_{VOL}(v,u) = \frac{O_{u(VOL)}}{\sum_{p \varepsilon R(v)} O_{p(VOL)}} \quad \ldots (4.2)$$

Notations are:

- $O_{u(VOL)}$and $O_{p(VOL)}$ represent the outgoing visits of links of page u and p respectively.
- R(v) represents the set of reference pages of page v.

Then$W^{in}_{VOL}(v,u)$, the popularity from the number of visits of inlinks and $W^{out}_{VOL}(v,u)$, the popularity from the number of visits of outlinks are used to calculate page rank using equation 4.3.

$$EWPR_{VOL}(u) = (1-d) + d \sum_{v \varepsilon B(u)} EWPR_{VOL}\, W^{in}_{VOL}(v,u) W^{out}_{VOL}(v,u) \quad \ldots (4.3)$$

Notations are:

- *d* is a dampening factor.
- B(u) is the set of pages that point to u.
- $EWPR_{VOL}(u)$ and $EWPR_{VOL}(v)$ are the rank scores of page u and v respectively.
- $W^{in}_{VOL}(v,u)$ represents the popularity from the number of visits of inlinks
- $W^{out}_{VOL}(v,u)$ represents the popularity from the number of visits of outlinks.

4.1. ALGORITHM TO CALCULATE $EWPR_{VOL}$

The algorithm depicts the steps required to calculate page rank of web pages using proposed algorithm.

1. *Finding a website*: This step requires finding a website which has rich hyperlinks because the algorithm depends on the hyper structure of website. The website having rich hyperlinks will help in better distribution of page rank.
2. *Generating a web graph:* For selected website, web graph a generated in which nodes represent web pages and edges represent hyperlinks between web pages. The edges will help in evaluating the values of page rank.

3. *Calculating number of visits of hyperlinks:* Client side script is used to monitor the hits of hyperlinks. Whenever a web page is accessed, the script will be loaded on client side from the web server and information is sent to the web server on the form of web page id, hyperlinks of that page and hit counts of hyperlinks and this information is accessed by crawlers.
4. *Calculate page rank of each web page:* The values of $W^{in}_{VOL}(v,u)$, the popularity from the number of visits of inlinks and $W^{in}_{VOL}(v,u)$, the popularity from the number of visits of outlinks are calculated for each node using formulae given in equation 7 and 8 and these values are substituted in equation 9 to calculate values of page rank.
5. *Repetition of step 4:* The step 4 is used recursively until a stable value of page rank is obtained.

Figure 4.1 shown below explains the steps required to calculate page rank using $EWPR_{VOL}$.

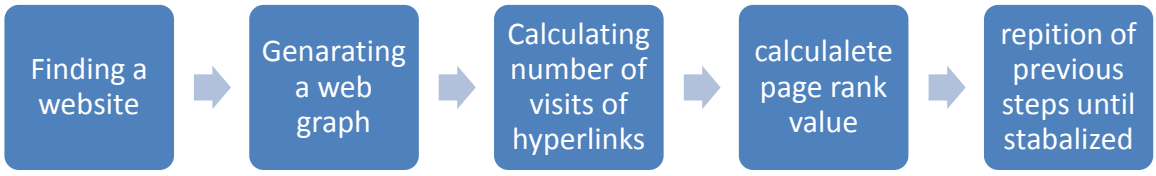

Figure 4.1: Algorithm to calculate $EWPR_{VOL}$

4.2. EXAMPLE TO ILLUSTRATE THE WORKING OF $EWPR_{VOL}$

The working of proposed algorithm has been illustrated via taking a hypothetical web graph having web pages A, B and C and links representing hyperlinks between pages marked with their number of visits shown in Figure 3.13.

The values of page rank for web pages A, B and C are calculated using equation 4.3.

$$EWPR_{VOL}(A) = (1-d) + dEWPR_{VOL}(C)W^{in}_{VOL}(C,A)W^{out}_{VOL}(C,A)$$

$$EWPR_{VOL}(B) = (1-d) + dEWPR_{VOL}(A)\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(A,B)\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(A,B)$$

$$EWPR_{VOL}(C) = (1-d) + d(EWPR_{VOL}(A)\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(A,C)\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(A,C) + EWPR_{VOL}(\mathrm{B})\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(B,C)\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(B,C)$$

Each intermediate value $\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(v,u)$ and $\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(v,u)$is calculated using equation 4.1 and 4.2.

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(C,A) = \frac{I_{A(VOL)}}{I_{A(VOL)}} = \frac{2}{2} = 1$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(C,A) = \frac{O_{A(VOL)}}{O_{A(VOL)}} = \frac{3}{3} = 1$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(A,B) = \frac{I_{B(VOL)}}{I_{B(VOL)} + I_{C(VOL)}} = \frac{1}{1+2} = \frac{1}{3}$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(A,B) = \frac{O_{B(VOL)}}{O_{B(VOL)} + O_{C(VOL)}} = \frac{2}{2+2} = \frac{2}{4}$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(A,C) = \frac{I_{C(VOL)}}{I_{C(VOL)} + I_{B(VOL)}} = \frac{4}{4+1} = \frac{4}{5}$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(A,C) = \frac{O_{C(VOL)}}{O_{C(VOL)} + O_{B(VOL)}} = \frac{2}{2+2} = \frac{2}{4}$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(B,C) = \frac{I_{C(VOL)}}{I_{C(VOL)}} = \frac{4}{4} = 1$$

$$\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(B,C) = \frac{O_{C(VOL)}}{O_{C(VOL)}} = \frac{2}{2} = 1$$

The calculated values are put in above equations to calculate the values of page rank.

I. For the dampening factor *d* = 0.35, we calculate the values of page rank for web page A, B and C.

$$EWPR_{VOL}(A) = 0.65 + 0.35\left(1 * \frac{2}{2} * \frac{3}{3}\right) = 1$$

$$EWPR_{VOL}(B) = 0.65 + 0.35\left(1 * \frac{1}{3} * \frac{2}{4}\right) = 0.70833$$

$$EWPR_{VOL}(C) = 0.50 + 0.50\left(1 * \frac{4}{5} * \frac{2}{4} + .70833 * \frac{4}{4} * \frac{2}{2}\right) = 1.03792$$

These values are used iteratively until the rank values of A, B and C get stabilized as shown in Table 4.1.

A	**B**	**C**
1	0.70833	1.03792
1.01327	0.70911	1.04005
1.01402	0.70915	1.04017
1.01406	0.70915	1.04017

Table 4.1: Values of Page Ranks using $EWPR_{VOL}$ at d=0.35

The final values are:

A=1.01406

B=0.70915

C=1.04017

II. For the dampening factor *d* = 0.50, we calculate the values of page rank for web page A, B and C.

$$EWPR_{VOL}(A) = 0.50 + 0.50\left(1 * \frac{2}{2} * \frac{3}{3}\right) = 1$$

$$EWPR_{VOL}(B) = 0.50 + 0.50\left(1 * \frac{1}{3} * \frac{2}{4}\right) = .58333$$

$$EWPR_{VOL}(C) = 0.50 + 0.50\left(1 * \frac{4}{5} * \frac{2}{4} + .58333 * \frac{4}{4} * \frac{2}{2}\right) = .99167$$

These values are used iteratively until the rank values of A, B and C get stabilized as shown in Table 4.2.

A	B	C
1	0.58333	0.99167
0.99584	0.58299	0.99066
0.99533	0.58294	0.99054
0.99527	0.58294	0.99052

Table 4.2: Values of Page Ranks using $EWPR_{VOL}$ at d=0.50

The final values are:

A=0.99527

B=0.58294

C=0.99052

III. For the dampening factor d = 0.85, we calculate the values of page rank for web page A, B and C.

$$EWPR_{VOL}(A) = 0.15 + 0.85\left(1 * \frac{2}{2} * \frac{3}{3}\right) = 1$$

$$EWPR_{VOL}(B) = 0.15 + 0.85\left(1 * \frac{1}{3} * \frac{2}{4}\right) = .29167$$

$$EWPR_{VOL}(C) = 0.50 + 0.50\left(1 * \frac{4}{5} * \frac{2}{4} + .29167 * \frac{4}{4} * \frac{2}{2}\right) = .73792$$

These values are used iteratively until the rank values of A, B and C get stabilized as shown in Table 4.3. The final values are:

A=.63531

B=0.24001

C=0.57001

A	B	C
1	0.2916	0.73792
0.69005	0.24776	0.59521
0.64258	0.24103	0.57335
0.63531	0.24001	0.57001

Table 4.3: Values of Page Ranks using $EWPR_{VOL}$ at d=0.85

The page rank values of A, B and C at various values of d has been shown in Table 4.4.

D	A	B	C
0.35	1.01406	0.70915	1.04017
0.50	0.99527	0.58294	0.99052
0.85	0.63531	0.24001	0.57001

Table 4.4: Values of Page Ranks using $EWPR_{VOL}$

4.3. BENEFITS OF PROPOSED ALGORITHM

As observed from Table 4.4, if the value of dampening factor *d* increases, the page rank value of web pages also decreases.

The original Weighted PageRank algorithm distributes the rank of a web page among its outgoing linked pages in proportional to their importance or popularity. $W^{in}(v,u)$, the popularity from the number of inlinks and $W^{out}(v,u)$, the popularity from the number of outlinks does not include usage trends. It does not give more popularity to the links most visited by the users. The Weighted PageRank using VOL makes use of web structure mining and web usage mining but it neglects the popularity from the number of outlinks i.e., $W^{out}(v,u)$. In proposed algorithm,$W^{in}_{VOL}(v,u)$, the popularity from the number of visits of inlinks and $W^{out}_{VOL}(v,u)$, the

popularity from the number of visits of outlinks are used to calculate the value of page rank. In this way the algorithm helps in sorting the resultant web pages in accordance of users need and it has following advantages.

1. The page rank using original WPR remains unaffected whether the page has been accessed by the users or not. i.e.; the relevancy of a web page is ignored. But the page rank using proposed method $EWPR_{VOL}$ assigns high rank to web pages having more visits of links.
2. The page rank using original WPR depends only on the link structure of the web and remains same whether the web page has been accessed by the user or not. Although the algorithm WPR_{VOL} makes use of web structure mining and web usage mining to calculate the value of page rank but it ignores the popularity from the number of outlinks$\mathrm{W}^{\mathrm{out}}(\mathrm{v},\mathrm{u})$. On the other side, our proposed method $EWPR_{VOL}$ makes use of $\mathrm{W}_{\mathrm{VOL}}^{\mathrm{in}}(v,u)$, the popularity from the number of visits of inlinks and $\mathrm{W}_{\mathrm{VOL}}^{\mathrm{out}}(A,C)$, the popularity from the number of visits of outlinks to calculate page rank.
3. The proposed method uses number of visits of links to calculate the rank of web pages. So the resultant pages are popular and more relevant to the users need.

4.4. LIMITATIONS AND FURTHER IMPROVEMENTS

The proposed method includes only number of visits of links (VOL) of web pages to calculate page ranks. Suppose there is a junk page whose initial page rank is high then it will be present on the top in the search results. Then users will access it and it will lead to increase in VOL corresponding to that page which will further improve page rank. Now more users will access it and rank will improve continuously. So the pages which are actually relevant for a given query will have less page rank than junk pages. So some other usage behavior factors must be introduced in addition to VOL to calculate the value of page rank. These factors can be:

- *Time spent on web page corresponding to a link***:** In most of the cases, the user spends more time if the web page is relevant to them. The algorithm must assign more weight to the link if more time is spent by the users on the web page corresponding to that link.

Most of the times, the time spent on the junk pages is very less as compared to the time spent on relevant pages. So this factor can help in improving the rank of relevant pages and lowering the rank of junk pages.

- *Most recent use of link:* If a web page is relevant for a user then the links pointing to that web page will be accessed more recently than the link pointing to a less relevant page. So the more recent link must be assigned more weight than the link which has not been used so far. So most recent use of link can also be used to calculate the page rank.
- *Information about the user:* A web page is not equally important for all users. The requirements may vary from user to user. So a web page may be relevant for one user but not for other. Some kind of user's information like age, gender, income, educational background can be used to categorize web pages according to different users' need.

CHAPTER 5

RESULTS AND OBSERVATION

RESULTS AND OBSERVATION

In this section, we compare the page rank of web pages using original WPR algorithm, WPR_{VOL} and the proposed algorithm. The original WPR makes use of web structure mining only to calculate the value of page rank. $W^{in}(v,u)$, the popularity from number of inlinks and $W^{out}(v,u)$, the popularity from the number of outlinks use the hyperlinks of graph and it do not change if the numbers of users accessing a link change. Although the WPR_{VOL} algorithm makes use of both web structure mining and web usage mining to calculate the value of page rank but it does not incorporate the popularity from the number of outlinks. But the proposed algorithm calculates $W_{VOL}^{in}(v,u)$, the popularity from number of visits of inlinks and $W_{VOL}^{out}(v,u)$, the popularity from the number of visits of outlinks by analyzing the user behavior. When the numbers of visits of links change, the rank of web pages also changes. So this technique considers the relevancy from the users' point of view and gives high rank to those web pages which are frequently accessed by users. In this way, the proposed method gives improved results than standard WPR and WPR_{VOL}.

The page rank values of each web page has been calculated using original Weighted PageRank algorithm (WPR), Weighted PageRank algorithm based on visits of links (WPR_{VOL}) and proposed algorithm for a web graph shown in Figure 3.13. The values of page rank using WPR have been calculated by the algorithm given in section 3.2.2. Table 5.1 shows the value of page rank using WPR algorithm at various values of dampening factor *d.*

d	**0.35**	**0.50**	**0.85**
A	1.00535	0.97677	0.58335
B	0.70865	0.58140	0.23335
C	1.01532	0.95351	0.51505

Table 5.1: Values of Page Ranks using WPR

The values of page rank using WPR_{VOL} algorithm have been calculated by the algorithm given in section 3.2.2. Table 5.2 shows the value of page rank using WPR_{VOL} algorithm at various values of dampening factor *d*.

d	**0.35**	**0.50**	**0.85**
A	1.01736	1	0.64037
B	0.68956	0.55556	0.21495
C	1.04960	1	0.57463

Table 5.2: Values of Page Ranks using WPR_{VOL}

The values of page rank using proposed algorithm $EWPR_{VOL}$ have been calculated by the algorithm given in section 4.1. Table 5.3 shows the value of page rank using $EWPR_{VOL}$ algorithm at various values of dampening factor *d*.

d	**0.35**	**0.50**	**0.85**
A	1.01406	0.99527	0.63531
B	0.70915	0.58140	0.23335
C	1.04017	0.99052	0.57001

Table 5.3: Values of Page Ranks using $EWPR_{VOL}$

d		**0.35**	**0.50**	**0.85**
WPR	**A**	1.00535	0.97677	0.58335
	B	0.70865	0.58140	0.23335
	C	1.01532	0.95351	0.51505
WPR(VOL)	**A**	1.01736	1	0.64037
	B	0.68956	0.55556	0.21495
	C	1.04960	1	0.57463
EWPR(VOL)	**A**	1.01406	0.99527	0.63531
	B	0.70915	0.58140	0.23335
	C	1.04017	0.99052	0.57001

Table 5.4: Values of Page Ranks using WPR, WPR_{VOL} and $EWPR_{VOL}$

Table 5.3 above compares the value of page rank using WPR, WPR_{VOL} and $EWPR_{VOL}$ algorithms at various values of dampening factor *d*.

The values of page rank using WPR, WPR_{VOL} and $EWPR_{VOL}$ have been compared using a bar chart. The values retrieved by $EWPR_{VOL}$ are better than original WPR and WPR_{VOL}. The WPR uses only web structure mining to calculate the value of page rank, WPR_{VOL} uses both web structure mining and web usage mining to calculate value of page rank but it uses popularity only from the number of inlinks not from the number of outlinks. The proposed algorithm $EWPR_{VOL}$ method uses number of visits of inlinks and outlinks to calculate values of page rank and gives more rank to important pages. Figure 5.1 compares the page ranks of A, B and C using WPR, WPR_{VOL} and $EWPR_{VOL}$ for d=0.35.

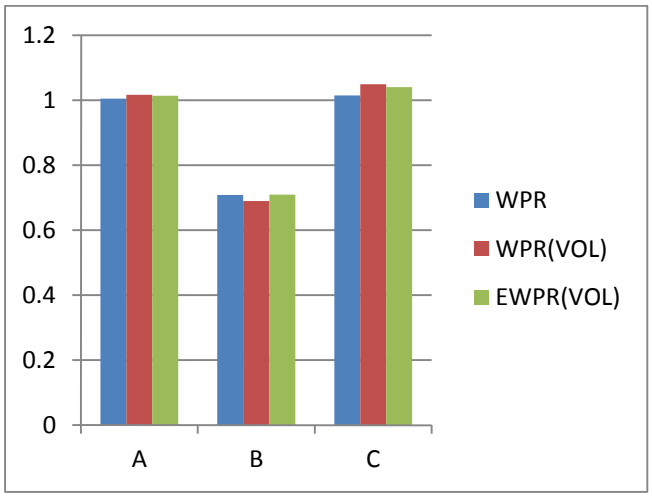

Figure 5.1: Comparison of Page Ranks at d=0.35

Figure 5.2 compares the page ranks of A, B and C using WPR, WPR_{VOL} and $EWPR_{VOL}$ for d=0.50.

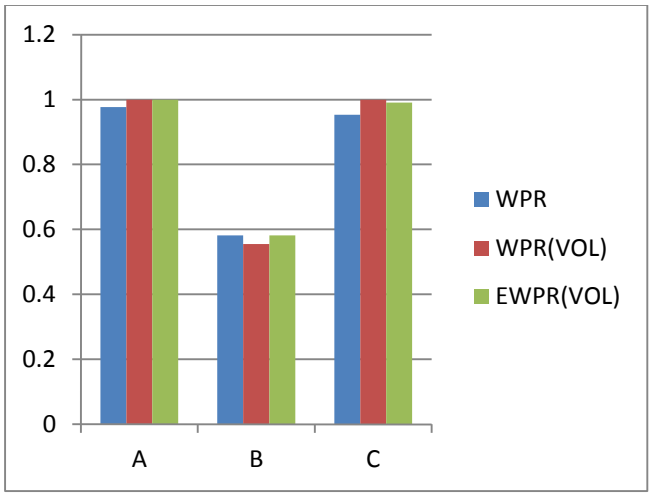

Figure 5.2: Comparison of Page Ranks at d=0.50

Figure 5.3 compares the page ranks of A, B and C using WPR, WPR_{VOL} and $EWPR_{VOL}$ for d=0.85.

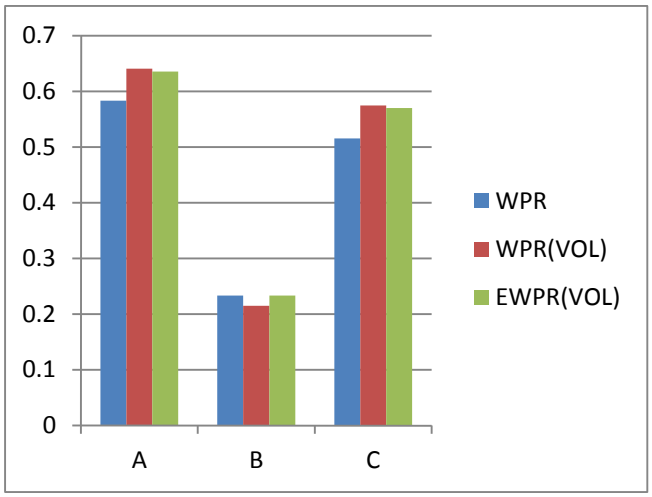

Figure 5.3: Comparison of Page Ranks at d=0.85

CONCLUSION AND FUTURE WORK

Due to enormous amount of information present on the web, the users have to spend a lot of time to get pages relevant to them. So it is necessary for search engine to sort the resultant web pages before presenting to the user. The original WPR algorithm calculates the page rank by using only the web graph and it ignores the relevancy of web pages from user' point of view. $W^{in}(v,u)$, the popularity from the number of inlinks and $W^{out}(v,u)$, the popularity from the number of outlinks does not include usage trends. It does not give more popularity to the links most visited by the users. The weighted PageRank using VOL makes use of web structure mining and web usage mining but it neglects the popularity from the number of outlinks i.e., $W^{out}(v,u)$. The proposed algorithm $EWPR_{VOL}$ makes use of number of visits of links (VOL) to calculate the values of page rank so that more relevant results are retrieved first. In this way, it may help users to get the relevant information quickly. Some of the future works for the proposed algorithm are:

1. The values of page rank have been calculated on a small web graph only. A web graph with large number of websites and hyperlinks should be used to check the accuracy and importance of method.
2. The graph and number of visits of links (VOL) used in the validation of the algorithm are hypothetical. The algorithm needs to be validated with real data so that the actual relevancy of the method could be evaluated.
3. The number of visits of links (VOL) only is not strong enough to determine the value of page rank. Some other measures like most recent use of link, information about the user and time spent on web page corresponding to a link can also be used to calculate the value of page rank. So the future work includes deriving a formula for page rank using these parameters also.

REFERENCES

[1] Brin, Sergey and Page, Lawrence, "The Anatomy of a Large-Scale Hypertextual Web Search Engine. In Seventh International World-Wide Web Conference (WWW 1998), 14-18 April, 1998, Brisbane, Australia.

[2] Companion slides for the text by Dr. M. H. Dunham, "Data Mining: Introductory and Advanced Topics", Prentice Hall, 2002.

[3] C. Ding, X. He, P. Husbands, H. Zha, and H. Simon, "Link Analysis: Hubs and Authorities on the World". Technical report:47847, 2001.

[4] Dell Zhang, Yisheng Dong, "A novel Web usage mining approach for search engines", Computer Networks 39 (2002) 303–310

[5] D. Cohn and H. Chang, "Learning to Probabilistically identify Authoritative Documents". In Proceedings of 17th International Conf. on Machine Learning, pages 167-174. Morgan Kaufmann, San Francisco, CA, 2000.

[6] Gyanendra Kumar, Neelam Duhan, A. K. Sharma, "Page Ranking Based on Number of Visits of Links of Web Page", Department of Computer Engineering, YMCA University of Science & Technology, Faridabad, India.

[7] Jaroslav Pokorny, Jozef Smizansky, "Page Content Rank: An Approach to the Web Content Mining".

[8] Jinguang Liu & Roopa Datla, "Web Usage Mining – Pattern Discovery and its application"

[9] Kosala, Raymond; Hendrik Blockeel ,"Web Mining Research: A Survey". *SIGKDD Explorations* 2 (1) July 2000).

[10] Longzhuang Li, Yi Shang, and Wei Zhang, "Improvement of HITS- based Algorithms on Web Documents", WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA. ACM 1-58113-449-5/02/0005.

[11] Naresh Barsagade, "Web Usage Mining And Pattern Discovery: A Survey Paper", CSE 8331, Dec.8, 2003.

[12] Neelam Duhan, A. K. Sharma, Komal Kumar Bhatia, "Page Ranking Algorithms: A Survey" Advance Computing Conference, 2009. IACC 2009 IEEE International.

[13] Neelam Tyagi, Simple Sharma, "Weighted Page Rank Algorithm Based on Number of Visits of Links of Web Page", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012.

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

[15] Raymond Kosala, Hendrik Blockee, "Web Mining Research: A Survey", ACM Sigkdd Explorations Newsletter, June 2000, Volume 2.

[16] Saeko Nomura, Satoshi Oyama, Tetsuo Hayamizu, "Analysis and Improvement of HITS Algorithm for DetectingWeb Communities".

[17] Salton G. and Buckley, C., "Weighting Approaches in Automatic Text Retrieval". In Information Processing and Management, 1998, Vol. 24, pp. 513-523.

[18] Wenpu Xing and Ali Ghorbani, "Weighted PageRank Algorithm", Faculty of Computer Science, University of New Brunswick,Fredericton, NB, E3B 5A3, Canada.

[19] Zdravko Markov and Daniel T. Larose, "Mining the Web: Uncovering Pattrens in Web Content, Structure, and Usage Data". Copyright 2007 John Wiley & Sons, Inc.

[20] http://en.wikipedia.org/wiki/Web_search_engine

[21] http://en.wikipedia.org/wiki/World_Wide_Web

[22] http://en.wikipedia.org/wiki/Web_mining