International Journal of Applied Engineering and Technology ISSN: 2277-212X (Online)
An Open Access, Online International Journal Available at http://www.cibtech.org/jet.htm
2017 Vol. 7 (1) January-March, pp.1-18/Rane and Pathak
Review Article

WEB STRUCTURE MINING- A STUDY AND FUTURE IMPROVEMENTS

***Devendra Tanaji Rane and Ganesh R. Pathak**
Department of Information Technology, Sinhgad College of Engineering, Pune, India
**Author for Correspondence*

ABSTRACT
Today, internet is full of structured or unstructured information and this information influences people or society directly or indirectly. With the rapid growth of internet technologies, the web is considered as a world's largest repository of knowledge. Web data processing is the method of handling high volume of data which is not so easy. The rapid advent in internet technology has led the users to get easily confused in large hypertext structure. Fetching the relevant information from this huge web of structured data has become the need nowadays. In order to achieve this goal, we employ the concept of web mining. Specifically, we concentrate on a subsidiary of Web Mining: Web Structure Mining which is defined as the process of analysing the structure of hyperlink using graph theory. There are many algorithms for web structure mining such as PageRank Algorithm, HITS, Weighted PageRank Algorithm, Topic Sensitive PageRank Algorithm (TSPR), Weighted Page Content Rank Algorithm (WPCR) etc. In "Web Structure Mining- A study and future improvements" survey paper, we have described the outline of all the algorithms, identify their strengths and limitations and also suggest a few future Improvements.

Keywords: *Web Structure Mining, Web Content Mining, Web Usage Mining, HITS, Efficiency, Hyperlink, Weighted PageRank, TSPR*

INTRODUCTION
Web mining is the application of data mining techniques to extract knowledge from web data, including web documents, hyperlinks between documents, usage logs of web sites, etc. Web mining can be broadly divided into three distinct categories, according to the kinds of data to be mined: Web Content Mining, Web Structure Mining and Web Usage Mining.
Web Content Mining (Ichangimath *et al.,* 2015) is the process of extracting useful information from the contents of web documents. Content data is the collection of facts a web page is designed to contain. It may consist of text, images, audio, video, or structured records such as lists and tables. Application of text mining to web content has been the most widely researched. Issues addressed in text mining include topic discovery and tracking, extracting association patterns, clustering of web documents and classification of web pages. Research activities on this topic have drawn heavily on techniques developed in other disciplines such as Information Retrieval (IR), Natural Language Processing (NLP) and Machine Learning.
The structure of a typical web graph consists of web pages as nodes, and hyperlinks as edges connecting related pages. Web Structure mining (Ichangimath *et al.,* 2015) is the process of discovering structure information from the web. This can be further divided into two kinds based on the kind of structure information used.
Hyperlinks (Ichangimath *et al.,* 2015) (Desikan *et al.,* 2004)*:* A hyperlink is a structural unit that connects a location in a web page to a different location, either within the same web page or on a different web page. A hyperlink that connects to a different part of the same page is called an Intra-document hyperlink, and a hyperlink that connects two different pages is called an inter-document hyperlink. Hence, with the help of Link Analysis, we have many algorithms to rank Web Pages.
Document Structure (Ichangimath *et al.,* 2015) (Desikan *et al.,* 2004)*:* Web page can also be organized in a tree structured format, based on the various HTML and XML tags within the page. Mining efforts here have focused on automatically extracting Document Object Model (DOM) structures out of documents.
Web Structure mining is the main focus of this paper. Detailed algorithms used in this category of mining are mentioned in subsequent chapters. Its application, challenges and improvements are also part of this paper.

Review Article

Web Usage mining (Ichangimath *et al.*, 2015) is the application of data mining techniques to discover interesting usage patterns from web usage data, in order to understand and better serve the needs of web-based applications. It is the process by which we identify the browsing patterns by analyzing the navigational behavior of user. It focuses on techniques that can be used to predict the user behavior while the user interacts with the web. It uses the secondary data (server logs) on the web. This activity involves the automatic discovery of user access patterns from one or more web servers. Through this mining technique we can ascertain what users are looking for on Internet. Some might be looking for only textual data, where as some others might be interested in multimedia data (Desikan *et al*., 2004).

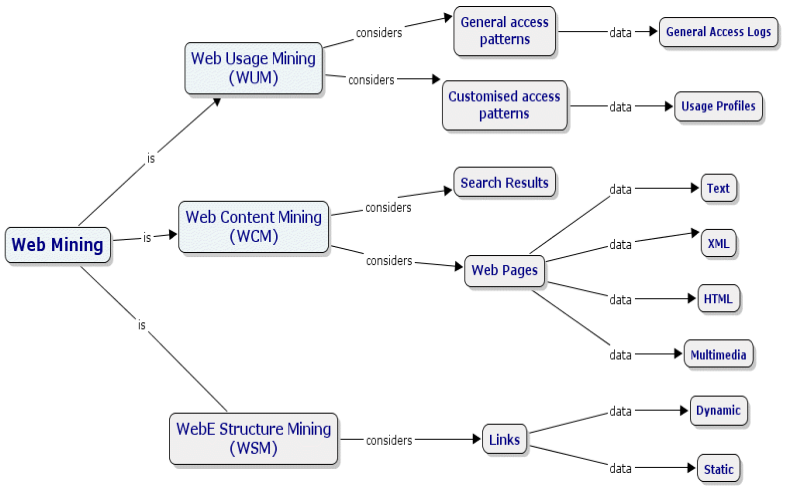

Figure 1: Web Mining Categories

Table 1: Web Mining Categories

	Web Mining			
	Web Content Mining		**Web Structure Mining**	**Web Usage Mining**
	IR View	**DB View**		
View of Data	-Unstructured -Structured	-Semi Structured -WebSite as DB	-Link Structure	-Interactivity
Main Data	-Text Documents -Hypertext Documents	-Hypertext Documents	-Link Structure	-Server Logs -Brower Logs
Representation	-Bag of Words -Phrases, concept of Ontology -Relational	-Edge labelled Graph -Relational	-Graph	-Relational Table -Graph
Method	-Machine Learning -Statistical (including NLP)	-Proprietary Algorithms -Association Rules	-Proprietary Algorithms	-Machine Learning -Statistical -Association Rules
Application Categories	-Categorization -Clustering -Finding extract rules -Finding Patterns in Text	-Finding frequent sub structures -Website schema Discovery	-Categorization -Clustering	-Site Construction -Adaptation and management -Marketing -User Modelling

The objects in the WWW are web pages, and links are in, out and co-citation i.e. two pages that are both linked to the same page. There are some possible tasks of link mining which are applicable in Web structure mining and are described as follows: (Jain and Purohit, 2011).

Link-Based Classification: It is the most recent upgrade of a classic data mining task to linked Domains. The task is to focus on the prediction of the category of a web page, based on words that occur on the page, links between pages, anchor text, html tags and other possible attributes found on the web page.

Link-Based Cluster Analysis: The goal in cluster analysis is to find naturally occurring sub-classes. The data is segmented into groups, where similar objects are grouped together, and dissimilar objects are grouped into different groups. Different than the previous task, link based cluster analysis is unsupervised and can be used to discover hidden patterns from data.

Link Type: There are a wide range of tasks concerning the prediction of the existence of links, such as predicting the type of link between two entities, or predicting the purpose of a link.

Link Strength: Links could be associated with weights.

Link Cardinality: The main task here is to predict the number of links between objects.

There are some uses of web structure mining like it is:

- Used to rank the user's query
- Deciding what page will be added to the collection
- Page categorization
- Finding related pages
- Finding duplicated web sites
- And also to find out similarity between them

Few Commercial Applications of Web Mining (Desikan *et al.*, 2004)*:* Excitement about the web in the past few years has led to the web applications being developed at a much faster rate in the industry than research in web related technologies. Many of these are based on the use of web mining concepts.

Personalized Customer Experience in B2C E-commerce—Amazon.com

In the case of an on-line store, getting in or out requires exactly one click, and thus, the main focus must be on customer experience in the store. This fundamental observation has been the driving force behind Amazon's comprehensive approach to personalized customer experience, based on the mantra "a personalized store for every customer". A host of web mining techniques, such as associations between pages visited and click-path analysis are used to improve the customer's experience during a "store visit". Knowledge gained from web mining is the key intelligence behind Amazon's features such as "instant recommendations", "purchase circles", "wish-lists" etc.

Web Search Engine—Google

Google is one of the most popular and widely used search engines. It provides users access to information from over 2 billion web pages that it has indexed on its server. The quality and quickness of the search facility makes it the most successful search engine. Earlier search engines concentrated on web content alone to return the relevant pages to a query. Google was the first to introduce the importance of the link structure in mining information from the web. PageRank, which measures the importance of a page, is the underlying technology in all Google search products, and uses structural information of the web graph to return high quality results.

News Integrator—Google

It integrates news from the online versions of all newspapers and organizes them categorically to make it easier for users to read "the most relevant news". It seeks to provide latest information by constantly retrieving pages from news site worldwide that are being updated on a regular basis. The key feature of this news page, like any other Google service, is that it integrates information from various web news sources through purely algorithmic means, and thus, does not introduce any human bias or effort.

Personalized Portal for the Web—MyYahoo, Google

Yahoo was the first to introduce the concept of a "personalized portal", i.e. a web site designed to have the look-and-feel and content personalized to the needs of an individual end-user. This has been an extremely popular concept and has led to the creation of other personalized portals such as Yodlee for

private information like bank and brokerage accounts. Mining MyYahoo usage logs provides Yahoo valuable insight into an individual's web usage habits, enabling Yahoo to provide personalized content, which in turn has led to the tremendous popularity of the Yahoo web site.

CiteSeer—Digital Library and Autonomous Citation Indexing (Desikan *et al.,* 2004)

NEC Research Index, also known as CiteSeer26 is one of the most popular online bibliographic indices related to computer science. The key contribution of the CiteSeer repository is its "Autonomous Citation Indexing" (ACI). Citation indexing makes it possible to extract information about related articles. Automating such a process reduces a lot of human effort, and makes it more effective and faster. CiteSeer works by crawling the web and downloading research related papers. Information about citations and the related context is stored for each of these documents. The entire text and information about the document is stored in different formats. Citation statistics for documents are computed that enable the user to look at the most cited or popular documents in the related field. These documents are ordered by the number of citations.

Literature Review

Internet search engines are special sites on the Web that are designed to help people find information stored on other sites. There are differences in the ways various search engines work, but they all perform three basic tasks (Ichangimath *et al.*, 2015):

- They search the Internet or select pieces of the Internet based on important words.
- They keep an index of the words they find, and where they find them.
- They allow users to look for words or combinations of words found in that index.

It is very important to note that the link structure of the web serves to bind all of the pages together. Hence, the emphasis is on Web Structure mining (Ichangimath *et al.*, 2015).

Search engines use automated software (known as crawlers, robots, bots or spiders) to follow links on Websites, harvesting information as they go. When someone submits a query to a search engine, the engine returns a list of sites, ranking them on their relevance to the keywords used in the search. How search engines assess your site and determine the relevance of the words often depends on how the specific search engine works. To understand this further, we need to have a look at the various page ranking algorithms. Without the help of the knowledge of how exactly Search Engines work, we cannot analyze the different algorithms properly.

With respect to Link Analysis in Web Structure Mining, we talk about five important Page ranking algorithms. The various Journal papers referred by talk about these algorithms in detail and give us an insight as to how exactly are we supposed to go about in coming up with a definite analysis for further improvements of these algorithms (Ichangimath *et al.*, 2015).

a) Web Structure Mining

Web structure mining, one of three categories of web mining for data, is a technique used to identify the relationship between Web pages linked by information or direct link connection. This structure data is discoverable by the provision of web structure schema through database techniques for Web pages. This connection allows a search engine to pull data relating to a search query directly to the linking Web page from the Web site the content rests upon.

This completion takes place through use of spiders scanning the Web sites, retrieving the home page, and then linking the information through reference links to bring forth the specific page containing the desired information.

Structure mining uses minimize two main problems of the World Wide Web due to its vast amount of information. The first of these problems is irrelevant search results. Relevance of search information become misconstrued due to the problem that search engines often only allow for low precision (the fraction of retrieved documents that are relevant to the query) criteria. The second of these problems is the inability to index the vast amount if information provided on the Web. This causes a low amount of recall (the fraction of the documents that are relevant to the query that are successfully retrieved) with content mining. This minimization comes in part with the function of discovering the model underlying the Web hyperlink structure provided by Web structure mining.

The main purpose for structure mining is to extract previously unknown relationships between Web pages. This structure data mining provides use for a business to link the information of its own Web site to enable navigation and cluster information into site maps. This allows its users the ability to access the desired information through keyword association and content mining. Hyperlink hierarchy is also determined to path the related information within the sites to the relationship of competitor links and connection through search engines and third party co-links. This enables clustering of connected Web pages to establish the relationship of these pages. On the WWW, the use of structure mining enables the determination of similar structure of Web pages by clustering through the identification of underlying structure. This information can be used to project the similarities of web content. The known similarities then provide ability to maintain or improve the information of a site to enable access of web spiders in a higher ratio. The larger the amount of Web crawlers, the more beneficial to the site because of related content to searches.

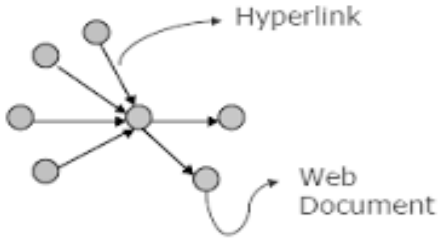

Figure 2: Web Graph Structure

The structure of typical web graph consists of Web pages as nodes, and hyperlinks as edges connecting between two related pages.

In the business world, structure mining can be quite useful in determining the connection between two or more business Web sites. The determined connection brings forth a useful tool for mapping competing companies through third party links such as resellers and customers. This cluster map allows for the content of the business pages placing upon the search engine results through connection of keywords and co-links throughout the relationship of the Web pages. This determined information will provide the proper path through structure mining to improve navigation of these pages through their relationships and link hierarchy of the Web sites.

With improved navigation of Web pages on business Web sites, connecting the requested information to a search engine becomes more effective. This stronger connection allows generating traffic to a business site to provide results that are more productive. The more links provided within the relationship of the web pages enable the navigation to yield the link hierarchy allowing navigation ease. This improved navigation attracts the spiders to the correct locations providing the requested information, proving more beneficial in clicks to a particular site.

Therefore, Web mining and the use of structure mining can provide strategic results for marketing of a Web site for production of sale. The more traffic directed to the Web pages of a particular site increases the level of return visitation to the site and recall by search engines relating to the information or product provided by the company.

This also enables marketing strategies to provide results that are more productive through navigation of the pages linking to the homepage of the site itself. To truly utilize website as a business tool web structure mining is a must (Keole and Pardakhe, 2013).

b) Hyperlinks and Document Structure Analysis

Web Structure mining can be further divided into two kinds based on the kind of structure information used i.e. Hyperlinks and Document Structure as explained in Introduction section of this paper.

Hyperlinks analysis covers the links with the page (Intra-document links) and links that connect different pages (Inter-document links). There has been a significant body of work on hyperlink analysis.

Document structure i.e. content within a webpage can also be organized in tree structured format, based on various HTML and XML tags within the page. Mining efforts here have focused on automatically extracting document object model (DOM) structures out of documents.
Many algorithms like PageRank, HITS developed using aforementioned analysis techniques.
First, we begin with the PageRank algorithm which was developed by Larry Page. This algorithm was the first step in ranking the various web pages which formed the basis for the various other algorithms that were later proposed. PageRank is a metric for ranking hypertext documents based on their quality. Researchers developed this metric for the popular search engine Google 4. The key idea is that a page has a high rank if it is pointed to by many highly ranked pages. So, the rank of a page depends upon the ranks of the pages pointing to it. This process is done iteratively until the rank of all pages is determined.
Secondly, we have *HITS* which was developed which tells us about Hubs and Authorities. The hub and authority scores computed for each web page indicate the extent to which the web page serves as a hub pointing to good authority pages or as an authority on a topic pointed to by good hubs. The scores are computed for a set of pages related to a topic using an iterative procedure called HITS. After HITS, we look at Weighted PageRank Algorithm which is basically an upgrade to the original PageRank algorithm. After this, we study Weighted Page Content Rank (WPCR) algorithm which makes use of both Web Structure Mining as well as Web Content Mining concepts.
Lastly, we have a look at Topic Sensitive PageRank algorithm which is based on the topic sensitivity of user Query. This algorithm is still being developed in theory. In this algorithm, different scores are computed, multiple important scores for each page under several topics that form composite PageRank score for those pages matching the query.
These algorithms are explained in details in chapter 3 ("Web structure mining- Applications, Algorithms, Challenges & Research Directions"). It is important to look at the applications of web structure mining, challenges and ongoing research areas which are covered in next chapter.

Web Structure Mining- Applications, Algorithms, Challenges & Research Directions

There are number of commercial applications on Internet that use web mining techniques. In this section, we will focus only on the important applications related to Web Structural mining. These applications use web structure mining techniques to achieve its task, which reduces lot of human efforts.

a) Applications

Search Engines- Crawling & Indexing (Jain and Purohit, 2011)*:* Web Search Engine is a tool enabling document search, with respect to specified keywords, in the Web and returns a list of documents where the keywords were found. It processes textual and hyper-textual information in diverse ways, but the capability to quickly fetch a large number of Web pages into a local repository and to index them based on keywords is required by many applications. Large scale programs that fetch tens of thousands of web pages per second are called crawlers, spiders, web robots, or bots. Crawling is usually performed to subsequently index the documents fetched. Together, a crawler and an index form key components of a Web Search engines (Online Web Article, http://www.web-datamining.net/structure/).
Components of Web Search Engine
1. User Interface
2. Parser
3. Web Crawler
4. Database
5. Ranking Engine

1) User Interface: It is the part of Web Search Engine interacting with the users and allowing them to query and view query results.
2) Parser: It is the component providing term (keyword) extraction for both sides. The parsers determine the keywords of the user query and all the terms of the Web documents which have been scanning by the crawler.
Term extraction procedure includes the following sub-procedures:
1. Tokenization

2. Normalization
3. Stemming
4. Stop word handling

3) Web Crawler: A web crawler is a relatively simple automated program, or script that methodically scans or "crawls" through Internet pages to create an index of the data it is looking for. Alternative names for a web crawler include web spider, web robot, crawler, and automatic indexer. When a web crawler visits a web page, it reads the visible text, the hyperlinks, and the content of the various tags used in the site, such as keyword rich meta tags. Using the information gathered from the crawler, a search engine will then determine what the site is about and index the information. Lastly, the website is included in the search engine's database and its page ranking process.

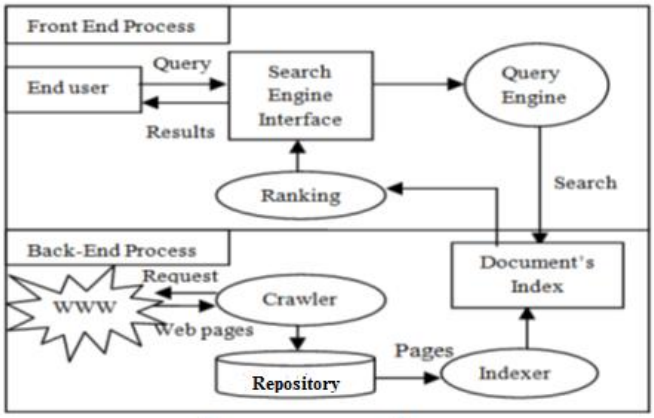

Figure 3: Web Crawler Architecture

4) Database: It is the component that all the text and metadata specifying the web documents scanned by the crawler.

5) Ranking Engine: The component is mainly the ranking algorithm operating on the current data, which is indexed by the crawler, to be able to provide some order of relevance, for the web documents, with respect to the user query.

Topic Directories

After crawlers & search engines, Topic Directories were the next significant feature to gain visibility (Online Web Article, http://www.web-datamining.net/structure/).

It maintains the treelike branching hierarchies of links or keywords related to specific domain or topic. It offers values in two forms. The obvious contribution is the cataloguing of web content, which makes it easier to search. Once the user has located a few sites of interest, others belonging to the same sibling or ancestor categories may also be of interest. The second contribution is in the form of quality control. Because the links in a directory usually go through editorial scrutiny, however cursory, they tend to reflect the more authoritative and popular sections of the Web.

Clustering and Classification

Topic Directories built with human effort (e.g. Yahoo! Or Open Directory). Clustering or unsupervised learning is enhancement to this which automatically constructs directories. It is a classic area of machine learning and pattern recognition. Once taxonomy is created, it is necessary to maintain it with URLs for each topic as the web changes and grows. Human effort to this end may be greatly assisted by supervised learning, or classification. A classifier is first trained with corpus of documents that are labelled with topics. At this stage, the classifier analyses correlation between the labels and other document attributes to form models. Later the classifier is presented with unlabelled instances and is required to estimate their topics reliability.

a) Hyperlink Analysis

In this technique, Hyperlinks are analyzed and hypertext graphs of link structure are created. This helps to increase relevance based web search in the field of IR. The PageRank and HITS algorithms have led to a flurry of research activity in this area. It is also known as topic distillation. These are used in hyperlink-assisted ranking systems in social networking and bibliometry. These algorithms are explained in detail in next chapter.

b) Algorithms

1. *PageRank Algorithm:* This algorithm states that the Page Rank of a page is defined recursively and depends on the number and Page Rank metric of all pages that link to it (incoming links) (Chakrabarti, 2012). If a page has some important incoming links to it than its outgoing links to other pages also become important. A page that is linked to by many pages with high Page Rank receives a high rank itself. A Page Rank Algorithm considers more than 25 billion web pages on the www to assign a rank score (Urvashi and Singh, 2015). A simplified version of Page Rank is defined in Equation 1:

$$PR(u) = C \sum_{V \in B(u)} PR(v) / N v \qquad (1)$$

Here, "u" represents a web page,
B(u) is the set of pages that points to u,
PR(u) and PR(v) are rank scores of pages u and v respectively,
Nv denotes the number of outgoing links of pages v,
C is a factor used for normalization.
In Page Rank, the rank score of a page "p" is evenly divided among its outgoing links. The values assigned to the outgoing links of page p are in turn used to calculate the ranks of the pages to which page p is pointing as shown in following Figure.

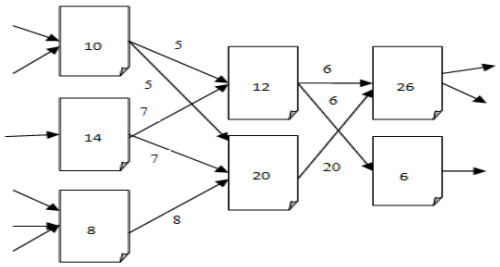

Figure 2: Distribution of Page Ranks

Later algorithm was modified, observing that not all users follow the direct links on WWW. The modified version is given in Equation 2:

$$PR(u) = (1-d) + d \sum_{V \in B(u)} PR(v) / N v \qquad (2)$$

Here, "d" is a damping factor that is usually set to 0.85 and it can be thought of as the probability of users' following the links and (1-d) as the page rank distribution from non-directly linked pages. This is specially done to handle dangling links (without any outgoing links) and disconnected links ("Lecture #3: PageRank Algorithm - The Mathematics of Google Search," Online Web Article, 2009) as shown below:

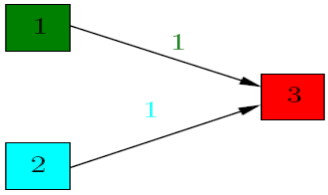

Figure 3: Dangling Links

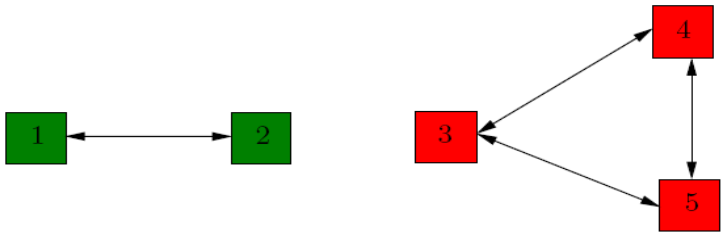

Figure 4: Disconnected Links

Computation of PageRank with Example ("Lecture #3: PageRank Algorithm - The Mathematics of Google Search," Online Web Article, 2009)*:* We compute the page rank algorithm by assuming a small universe of four web pages; A, B, C and D. The links from a page to itself or multiple outbound links from one single page to another single web page are ignored. Page Rank is initialized to the same value for all the web pages present in the web. In the Page Rank, we assume the sum of Page Rank over all pages equal to the total number of pages on the web at that time. We assume a probability distribution between 0 and 1 for all the web pages. The Page Rank transferred from a given page to the other web page of its outbound links in the next iteration is divided equally among all the outbound links of the given web page. Let us suppose, the page B has a link to pages C and A, page C has a link to page A and page D have links to all three pages. We assume the initial value for each web page as 0.25.

To compute the page rank of A:

If the only links in the system were from pages B, C and D to A, each outgoing link would transfer 0.25 to Web page A to compute the Page Rank of A in the next iteration.

PR (A) = PR (B) + PR (C) + PR (D)

With back links the equation will be,

PR (A) = PR (B)/2 + PR(C)/1 +PR (D)/3

Thus, upon the next iteration, page B would transfer half of its existing value or 0.125 to page A, because Page B has 2 back-links; to page A and to page C; and the other half or 0.125 to page C. And page C would transfer all of its existing value, which is 0.25, to the only page it links to the web page A. If D has three outbound links, the web page D would transfer one third of its existing value or 0.083 (0.25/3=0.083) values to web page A. At the completion of this iteration, page A will have a Page Rank of 0. 458.

PR (A) = (0.25/2) + (0.25/1) + (0.25/3) = 0. 458

The Markov Model of the Web: In this method, the founders of the PageRank model, force the transition probability matrix, which is built from the hyperlink structure of the web, to be stochastic and primitive. It represents the hyperlink structure of the web as a directed graph. The nodes of this digraph represent web pages and the directed arcs represent hyperlinks ("Lecture #3: PageRank Algorithm - The Mathematics of Google Search," Online Web Article, 2009). If we consider aforementioned example, our directed graph will be as shown in following figure:

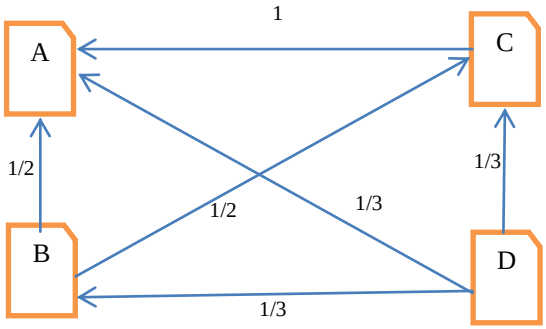

Figure 7: Directed Graph of Page Links

Weights are considered as per outgoing links. In our model, each page should transfer evenly its importance to the pages that it links to. Page A has no outgoing links so it has passed no importance to any page. This is dangling link. Whereas Page B has 2 outgoing links so it will pass on ½ of its importance to outgoing links. Similarly, Page C is passing its complete importance to page A. Page D is passing 1/3 of its importance to outgoing links as shown in figure above.

Let us denote by *A* the transition matrix of the graph,

$$A = \begin{bmatrix} 0 & 1/2 & 1 & 1/3 \\ 0 & 0 & 0 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Suppose that initially the importance is uniformly distributed among the 4 nodes, each getting ¼. Denote by v the initial rank vector, having all entries equal to ¼. Each incoming link increases the importance of a web page, so at step 1, we update the rank of each page by adding to the current value the importance of the incoming links.

This is the same as multiplying the matrix A with v. At step 1, the new importance vector is v1 = Av. We can iterate the process, thus at step 2, the updated importance vector is v2 = A(Av) = A2v. Numeric computations give:

As page A is dangling link here, if we iterate we will get 0 as show below,

$$v = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix}, \quad A.v = \begin{bmatrix} 0.46 \\ 0.08 \\ 0.21 \\ 0 \end{bmatrix}, \quad A.(A.v) = \begin{bmatrix} 0.04 \\ 0 \\ 0.04 \\ 0 \end{bmatrix}$$

$$A(A.(A.v)) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

To avoid this dangling effect damping factor is considered, it is assumed that importance of page A would be same as sum of incoming links and same is redistributed to all other pages equally i.e. (1+(1/2)+(1/3))/3= 0.61. If we form transition matrix again with this assumption we get:

$$A = \begin{bmatrix} 0 & 1/2 & 1 & 1/3 \\ 0.61 & 0 & 0 & 1/3 \\ 0.61 & 1/2 & 0 & 1/3 \\ 0.61 & 0 & 0 & 0 \end{bmatrix}$$

Now if we iterate this repeatedly considering initial page rank matrix as v, we get:

$$v = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix}, \quad A.v = \begin{bmatrix} 0.46 \\ 0.24 \\ 0.36 \\ 0.15 \end{bmatrix}, \quad A.(A.v) = \begin{bmatrix} 0.14 \\ 0.14 \\ 0.26 \\ 0.09 \end{bmatrix}$$

$$A(A.(A.v)) = \begin{bmatrix} 0.08 \\ 0.12 \\ 0.19 \\ 0.09 \end{bmatrix}, A(A(A.(A.v))) = \begin{bmatrix} 0.07 \\ 0.08 \\ 0.14 \\ 0.05 \end{bmatrix}$$

$$A(A(A(A.(A.v)))) = \begin{bmatrix} 0.04 \\ 0.06 \\ 0.10 \\ 0.04 \end{bmatrix}, A(A(A(A(A.(A.v))))) = \begin{bmatrix} 0.03 \\ 0.04 \\ 0.07 \\ 0.02 \end{bmatrix}$$

$$A(A(A(A(A(A.(A.v)))))) = \begin{bmatrix} 0.03 \\ 0.03 \\ 0.07 \\ 0.02 \end{bmatrix}$$

As with iterations, it tends to the equilibrium value so this is final PageRank vector of our web graph.
Problems of PageRank Algorithm are:

- It is a static algorithm, because of its cumulative scheme, popular pages tend to stay popular generally.
- Popularity of a site does not guarantee the desired information to the searcher so relevance factor also needs to be included.
- In Internet, available data is huge and the algorithm is not fast enough.
- It should support personalized search that personal specifications should be met by the search result.

2. *HITS (Hyper-link Induced Topic Search) Algorithm (IBM):* It is executed at query time, not at indexing time, with the associated hit on performance that accompanies query time processing. Thus, the hub (going i.e. page with good sources of links (Ichangimath *et al.*, 2015)) and authority (coming i.e. page with good sources of content (Ichangimath *et al.*, 2015)) scores assigned to a page are query specific ("Lecture #3: PageRank Algorithm - The Mathematics of Google Search," Online Web Article, 2009).
It is not commonly used by search engines. It computes two scores per document, hub and authority, as opposed to a single score of PageRank. It is processed on a small subset of "relevant" documents, not all documents as was the case with PageRank.
According to this algorithm first step is to collect the root set. That root set hits from the search engine. Then the next step is to construct the base set that includes the entire page that points to that root set. The size should be in between 1000-5000.
Third step is to construct the focused graph that includes graph structure of the base set. It deletes the intrinsic link, (the link between the same domains).
Then, it iteratively computes the hub and authority scores. A good hub is a page that points to many good authorities; a Good authority is a page that is pointed to by many good hubs. Hubs are the pages that act as resource lists. Authorities are having important contents.
A fine hub page for a subject points to many authoritative pages on that context and a good authority page is pointed by many fine hub pages on the same subject.
HITS assumes that if the author of page p provides a link to page q, then p confers some authority on page q (Urvashi and Singh, 2015). Kleinberg states that a page may be a good hub and a good authority at the same time.
The HITS algorithm considers the WWW as a directed graph G (V, E) where V is a set of vertices representing pages and E is a set of edges that match upto links (Urvashi and Singh, 2015). Following figure shows the hubs and authorities in web.

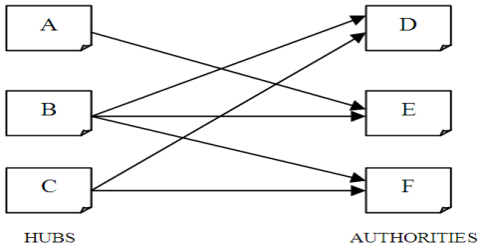

Figure 5: Hubs and Authorities

Implementation of HITS Algorithm (Devi *et al.*, 2014)*:*
The following steps explain the method for implementing HITS Algorithm.
Step 1: In the first step of the HITS algorithm we determine a base set S.

- let set of documents (most relevant pages to the query) returned by a standard search engine be called the root set R.
- Initialize S to R

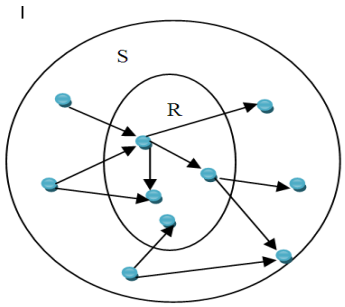

Figure 6: Expansion of Root Set R

Step 2:

- Add to S all pages pointed by any page in R.
- Add to S all pages that point to any page in R.

This is called Sampling step in which a set of relevant pages for the given query are collected.
Step3:

- Iterative step: This step finds hubs and authorities using the output of sampling step. The scores of hubs and authorities are calculated as follows:

$$H\,p = \sum_{q \in I(p)} A\,q$$

$$A\,p = \sum_{q \in B(p)} H\,q$$

Where, Hp and Ap represents the Hub score and authority score of a page.
I(p) and B(p) denotes the set of reference and referrer pages of page p.
The page's authority weight is proportional to the sum of the hub weights of pages that it links to.

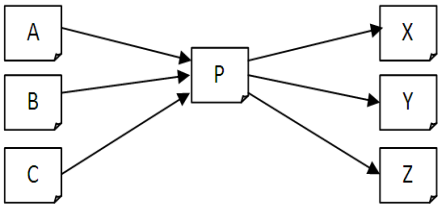

Figure 7: Authority and Hub of a Page P

Authority of page P is given as:
A(p) = H(A) + H(B) + H(C)
Hub of page P is given as:
H(P) = A(X) + A(Y) + A(Z)
Step 4:

- *Normalization:* The final hub-authority scores of nodes are determined after infinite repetitions of the algorithm.

In each iteration diverging values of authority and hub are obtained. So, it is necessary to normalize the values after each iteration.
Normalization is done by dividing each Hub score by the square root of sum of the squares of all the Hub scores, and dividing each Authority score by the square root of sum of the squares of all the Authority scores.
Advantages of HITS (Devi *et al.*, 2014)*:*

- HITS scores due to its ability to rank pages according to the query string, resulting in relevant authority and hub pages.
- The ranking may also be combined with other information retrieval based rankings.
- HITS is sensitive to user query (as compared to PageRank).
- Important pages are obtained on basis of calculated authority and hubs value.
- HITS is a general algorithm for calculating authority and hubs in order to rank the retrieved data.
- HITS induces Web graph by finding set of pages with a search on a given query string.
- Results demonstrate that HITS calculates authority nodes and hubness correctly.

Problems of HITS Algorithm are (Devi *et al.*, 2014)*:* Although HITS provides good search results for a wide range of queries, HITS did not work well in all cases due to the following reasons:

- *More Query Time*: The query time evaluation is expensive. As HITS calculate rank of pages at query time so it takes more time to response to the query.
- *Irrelevant authorities*: The rating or scores of authorities and hubs could rise due to flaws done by the web page designer.
- *Irrelevant Hubs*: A situation may occur when a page that contains links to a large number of separate topics may receive a high hub rank which is not relevant to the given query. Though this page is not the most relevant source for any information, it still has a very high hub rank if it points to highly ranked authorities.
- *Mutually reinforcing relationships between hosts*: HITS emphasizes mutual reinforcement between authority and hub webpages. A good hub is a page that points to many good authorities and a good authority is a page that is pointed to by many good hubs.
- *Topic Drift*: Topic drift occurs when there are irrelevant pages in the root set and they are strongly connected. Since the root set itself contains non-relevant pages, this will reflect on to the pages in the base set. Also, the web graph constructed from the pages in the base set, will not have the most relevant nodes and as a result the algorithm will not be able to find the highest ranked authorities and hubs for a given query.
- *Less Feasibility*: As HITS compute Rank value at query time, it is not feasible for today's search engines, which need to handle tens of millions of queries per day.

As mentioned above though HITS is good algorithm to get relevant results, it can't be used for huge volume of web pages. Variations of PageRank algorithms are listed below to add up relevance factor while computing Page ranking.
3. *Weighted PageRank Algorithm:* A weighted PageRank (WPR) algorithm is an extension of the PageRank algorithm (Devi *et al.*, 2014; Chakrabarti, 2012). 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.

$$Win(m,n) = In/\left(\sum_{p \in R(m)} Ip\right)$$

$$Wout(m,n) = On/\left(\sum_{p \in R(m)} Op\right)$$

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.

The formula as proposed by Wenpu *et al.*, for the WPR is as shown below is a modification of the PageRank formula.

$$WPR(n) = (1-d) + d \sum_{m \in B(n)} WPR(m) \cdot Win(m,n) \cdot Wout(m,n)$$

Problems of WPR:

- Relevancy of the web page is completely ignored in this algorithm.

4. *Weighted Page Content Rank Algorithm (WPCR):* Weighted Page Content Rank Algorithm (WPCR) (Ichangimath *et al.*, 2015) is a proposed page ranking algorithm which is used to give a sorted order to the web pages returned by a search engine in response to a user query. WPCR is a numerical value based on which the web pages are given an order. This algorithm employs web structure mining as well as web content mining techniques. Web structure mining is used to calculate the importance of the page and web content mining is used to find how much relevant a page is. Importance refers to the popularity of the web page, that is, how many pages are pointing to it. Relevancy means matching the web page with the relevant query being entered by the user. It gives different weights to web links based on three important attributes: Relative position of web page, tag where link is present and length of anchor text.

Problems with WPCR:

- Relative position of a web page is not always effective indicating that the logical position of a web page may not always point to the physical page.

5. *Topic Sensitive PageRank Algorithm:* In Topic Sensitive PageRank, several scores are computed (Devi *et al.*, 2014): 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).

6. 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.

For each web document query sensitive importance score. The results are ranked according to this composite score. It 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.

Problems with TSPR:

- Though this has capability to improve relevance it is only available to text based web pages as of now.
- It has performance issues so not yet completely implemented in any commercial search engine.

c) Comparison

Following Table Shows the Comparisons between Different Web Structure Mining Algorithms (Jain and Purohit, 2011; Ichangimath *et al.*, 2015)

Algorithm	PageRank	HITS	Weighted PageRank	WPCR	Topic Sensitive PageRank
Main Technique	Web Structure Mining	Web Structure Mining, Web Content Mining	Web Structure Mining	Web Structure Mining, Web Content Mining	Web Structure Mining, Web Content Mining
Methodology	This algorithm computes the score for pages at the time of indexing of the pages.	It computes the hubs and authority of the relevant pages. It relevant as well as important page as the result	Weight of web page is calculated on the basis of input and outgoing links and on the basis of weight the importance of page is decided.	Weight of web page is calculated basis of importance and relevance of page	Multiple importance scores for each page per topic
Input Parameter	Back links	Content, Back and Forward links	Back links and Forward links.	In and out links & content	In and out links & content
Relevancy	Less (this algo. Rank the pages on the indexing time)	More (this algo. Uses the hyperlinks so it gives good results and also consider the content of the page)	Less as ranking is based on the calculation of weight of the web page at the time of indexing.	More as relevance is considered along with weight.	More as relevance is considered along with importance score.
Quality of Results	Medium	Less than PR	Higher than PR	Higher than PR	Higher than PR
Importance	High. Back links are considered.	Moderate. Hub & authorities scores are utilized.	High. The pages are sorted according to the importance.	High, weight & relevance both are checked	High, weight & relevance both are checked
Limitation	Results come at the time of indexing and not at the query time.	Topic drift and efficiency problem	Relevancy is ignored.	Relative position of a web page is not always effective	Only available to text based web pages
Search Engine	Google	Clever	Research Model	Research model	Research Model

d) Challenges

Performance & Storage Issues (Langville and Meyer, 2002)*:*

All the algorithms proposed above may provide satisfactory performance in some cases but many times the user may not get the relevant information. The problem we all face when we search a topic in the web using a search engine like Google is that we are presented with millions of search results. First of all it not practically feasible to visit all these millions of web pages to find the required information. Sometimes, we may also not get the relevant information. In PageRank size of the Markov matrix (used during PageRank calculation) makes storage issues nontrivial in PageRank. Many times due to huge volumes, matrix does not fit into the main memory even after using compression techniques. This cause Storage and Speed issues in PageRank algorithms.

Unlike PageRank, HITS is query-dependent due to its creation of a neighborhood graph of pages related to the query terms. HITS forms both an authority matrix and a hub matrix from the hyperlink adjacency matrix, rather than one Markov chain. As a result, HITS returns both authority and hub scores for each page, whereas PageRank returns only authority scores. Therefore, HITS requires more space and more time for calculating hubs & authorities.

*Spam (*Langville and Meyer, 2002)*:*

Another area drawing attention is spam identification and prevention. This was cited by Monika Henzinger, former Research Director at Google, as a present "challenge" in an October 2002 paper. Once thought to be impervious to spamming, researchers have been revealing subtle ways of boosting PageRank. The paper by Bianchini *et al.*, based on its suggested ways to alter PageRank, goes on to describe how to identify spamming techniques, such as link farms, which can take the form of a regular graph. This is a first step toward preventing spam. However, as long as the web provides some mercantile potential, search engine optimization companies will exist and the papers they write for spammers will circulate. At least a dozen or so papers with nearly Langville and Meyer (2002): Deeper Inside PageRank 369 the same title exist for spammers, "PageRank Explained and How to Make the Most of It". Clearly, this makes for an ongoing war between search engines and the optimization companies and requires constant tweaking of the underlying algorithms in an attempt to outwit the spammers.

Low Relevance:

All the algorithms explained above may provide satisfactory performance in some cases but many times the user may not get the relevant information. The problem we all face when we search a topic in the web using a search engine like Google is that we are presented with millions of search results. First of all it is not practically feasible to visit all these millions of web pages to find the required information. Sometimes, we may also not get the relevant information. Algorithms which look for content of the document and accordingly give weightage to the page like WPCR have improved relevance to the some extent. Still these algorithms work only on text based content. Lot of improvement is required in Image, media contents based search.

Stale PageRank Vectors (Langville and Meyer, 2002)*:*

PageRank Vectors generated by Search engines do get refreshed or updated periodically. But for few sites like News web pages, Market trading web sites get updates dynamically every 5-10 min. PageRank Vectors updates frequency is less compared to this and this causes stale PageRank. Since the PageRank updating problem is really a Markov chain with a particular form, Markov chain researchers have been studying the updating problem for some time. The dynamic nature of the web creates challenges. It has pushed researchers to develop better solutions to the old problem of updating the stationary vector of a Markov chain.

e) Improvements and Ongoing Research

- The major problem in all these algorithms is that none of them include the "*Intelligent Search Factor*" (Ichangimath *et al.*, 2015). By this, we mean that there is a need for interpreting the inherent meaning of the query and indexing should be based on that. Hence, we can further concentrate on the last algorithm- "Topic Sensitive PageRank Algorithm" wherein we can concentrate on fetching the relevant content correctly and also fetch that relevant web page efficiently thereby maintaining a perfect balance

between the two factors and also keeping in mind, the Intelligent Search factor to improve the user experience.

A large amount of extensive research is going on in analyzing all the algorithms for their efficiency because we are supposed to deal with real time data and every day the amount of data that is being uploaded on the web is increasing exponentially. This will lead to lot of future problems while fetching the relevant content. Providing the user with an enhanced experience without compromising on the performance of the algorithms should be considered a priority.

- *Dynamic PageRank Vectors Updates:*

Viewing the web as a dynamic organism introduces some interesting areas of research. The web's constant growth and frequent updates create an evolving network, as opposed to a static network. Adaptive algorithms have been presented to accommodate for this evolution. Google itself has begun research on "stream of text" information such as news and TV broadcasts. Such dynamic content creates challenges that need tailored solutions. New algorithm like RankBrain which is based on machine learning techniques and keeps updating PageRank vectors dynamically is already in use by Google. RankBrain doesn't need any input. It just needs a dataset, over which it applies its learning process in order to generate and then refine its algorithms. It is enhancements to its earlier algorithms like Panda, Hummingbird. Google claims that this is based on machine deep learning techniques. Rank Brain is capable of learning and recognizing new patterns and then revising SERPS (search engine results page) based on its new knowledge.

Many researches are going on to use genetic algorithms (uses techniques like crossover, inheritance, mutation, selection etc.) in web structure mining to improve indexing and page ranking. Genetic algorithms are capable of deriving classification rules and selecting optimal parameters for detection process. The application of Genetic Algorithm to the network data consist primarily of following steps:

1. Collect the information to be analysed;
2. Apply genetic algorithms that is trained with the classification rules learned from the information collected from the network analysis done;

The designed system then uses the set of rules to classify the documents and vectors are generated accordingly to ranking the pages.

- *Other Research Areas:*

As the web continues its amazing growth, the need for smarter storage schemes and even faster numerical methods will become more evident. Both are exciting areas for computer scientists and numerical analysts interested in information retrieval.

Spam identification and prevention is another area drawing attention of many researchers. New techniques are required to optimize search engines to outwit the spammers.

Conclusion

This paper described several web structure mining algorithms like PageRank algorithm, Weighted PageRank algorithm, Weighted Content PageRank algorithm (WPCR), HITS etc. We analyzed their working and limitations.

Also, certain improvements to increase the efficiency of the algorithms and also to enhance the user experience are mentioned. Hence, this paper can be used as a reference to understand the limitations of the various page ranking algorithms, its calculation methods, challenges and to understand how they can be improved further. Special emphasis is laid on Topic Sensitive PageRank Algorithm and since it can be improved further in order to increase the efficiency with which fetch the relevant content that we need rather than just fetch some content which is not at all relevant. Improvement can be done using machine learning techniques in PageRank.

REFERENCES

Web Datamining (no date). Available: http://www.web-datamining.net/structure/,.

Chakrabarti S (2012). *Mining the Web: Discovering Knowledge from Hypertext Data*, (USA, Boston: Elsevier).

Desikan P, Kumar V and Srivastava J (2004). *Web Mining— Concepts, Applications, and Research Directions* [Online Article]. Available: http://dmr.cs.umn.edu/Papers/P2004_4.pdf.
Devi P, Dixit A and Gupta A (2014). Comparative Study of HITS and Page Rank Link based Ranking Algorithms. *International Journal of Advanced Research in Computer and Communication Engineering, ISSN (Online): 2278-1021; ISSN (Print): 2319-5940* **3**(2).
Ichangimath A, Joshi C and Vathsala MK (2015). Web Structure Mining- A Study on Different Page Ranking Algorithms and their Future Improvements. *International Journal of Computer Systems (ISSN: 2394-1065)* **2**(6).
Jain R and Purohit GN (2011). Page Ranking Algorithms for Web Mining. *International Journal of Computer Applications (0975 – 8887)* **13**(5).
Keole RR and Pardakhe NV (2013). Analysis of Various Web Page Ranking Algorithms in Web Structure Mining. *International Journal of Advanced Research in Computer and Communication Engineering* **2**(12).
Langville AN and Meyer CD (2002). Deeper Inside PageRank. *Internet Mathematics* **1**(3) 335-380.
Lecture #3: Page Rank Algorithm - The Mathematics of Google Search [Online Web Article] (2009). Available: http://www.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture3/lecture3.html.
Urvashi and Singh R (2015). Page Content Rank: An Approach to the Web Content Mining. *International Journal of Engineering Trends and Technology (IJETT)* **22**(2).