Comparative Study of Web Page Ranking Algorithms
Atul Kumar Srivastava¹, Mitali Srivastava², Rakhi Garg³, P. K. Mishra⁴
Department of Computer Science, Faculty of Science¹, ², ⁴, Mahila Maha Vidyalaya³
Banaras Hindu University, Varanasi, (U.P.), INDIA¹, ², ³, ⁴

Abstract: With the exponential growth of information on web, getting relevant information regarding user query through search engines is a tedious job today. Several search engines use link analysis algorithms to rank the web pages according to the need. But these algorithms are still lacking with efficiency, scalability and relevancy issues. This paper put forward survey of various improved ranking algorithms and their pros and cons. Further, we have included comparative study of various ranking algorithms mainly PageRank and HITS based on computation environments like Sequential, Parallel. This will help scientist, researchers, and academicians working in this area to understand the existing algorithms and develop one which is need of today’s environment.

Keywords: Web mining, Web structure mining, PageRank algorithm, HITS algorithm, Improved PageRank algorithms, Comparison of various page rank algorithms.

I. Introduction

World Wide Web (WWW) has become very popular and interactive medium to broadcast information. With the exponential growth of the Web, there is a huge amount of data and information available in web, so that accessing information from web with accuracy and speed is a big challenge for both people and software. Many problems with web exist like personalization of web pages, finding useful information, creating the knowledge based on extracted information and Learning about consumers and users. These problems arise due to multiplicity of data (i.e. structured, semi structured and unstructured) present in a web, containing many redundant information, due to dynamic nature of a web etc. [2, 11]

Web mining is emerged as a broad research area to solve these issues in last few decades. The web mining field is a converging field with Database, Information Retrieval and Artificial Intelligence [1]. Web mining is the application of data mining techniques to discover and extract the useful and previously unknown information from the web data. Web mining can be classified into three main categories according to the type of data to be mined [3, 5]:

![Figure 1: Categorization of web mining](image)

It is difficult to retrieve relevant and useful information from web because of existence of data in large amount and of its heterogeneous nature. As a result of huge and heterogeneous data available on WWW user get thousands or millions of web pages related to their query through search engine. Web users do not have much time and patience to go through all returned pages to find the relevant information that are of their interest and use [2, 3]. Web structure mining uses social network analysis and various link analysis algorithms to help web users to get information of their use on time.

Link analysis or web page ranking algorithms rank web pages based on authority, popularity and prestige of web pages. The role of link analysis algorithm is to select the web pages that are most likely be able to satisfy the user’s need and provide them higher ranks [13]. There are two main Link analyses or ranking algorithms: PageRank (used by Google) and HITS. Page rank algorithms are used to rank web pages according to the relevancy. Link analysis algorithms try to keep the desired result within the top few pages, otherwise, the web search engine could be considered as unserviceable. There are following issues with the basic link analysis algorithms [3, 7, 14]:

a) Uniformly sampling of web pages
b) Discovering duplicate hosts
c) To model in web graph
d) To avoid web spamming

e) To discover the content quality of web page

In this paper, we have reviewed page rank algorithms and focus on their pros and cons. Section II includes web structure mining in details. In section III, we reviewed two popular link analysis algorithms (sometimes referred as page ranking algorithm): PageRank and HITS algorithm, and their improvements over the decades. Section IV shows comparative study of ranking algorithms in environment like Sequential, Parallel and distributed. Finally, Section V concludes the Paper.

II. Web Structure Mining

Web structure mining is a process which discovers the structural summary about the hyperlinks and web pages. Web structure mining could be divided basically into two categories: Document structure analysis and Hyperlink analysis [9]. Document Structure analysis provides structural information of web pages i.e. how contents are organized in HTML and XML tags. On the other hand Hyperlink analysis provides how the web pages are connected with each other. Link information is used to combine with content of web to measure quality of information. It is also useful to find community of web pages according to user’s common interest. Many researchers have summarized the following some basic goals for determining the quality of individual web pages and group of web pages [5, 9]:

a) To count the local links in Web tuples (rows in a web table) in web table (representation of web pages). Local links are those links which connects different web documents which are related to same server. This also specifies the completeness of a particular web site.

b) To count the links in web tuples which are global in nature and the links spans different web sites. This describes visibility of web page and ability to relate same or other related documents across the web.

c) To measure the frequency of identical web tuples that appears in the indexed web pages or among the Web tables.

Link analysis provides an effective tool to calculate the importance of web pages on any particular topics. To search any web page from the search engines involves two main steps: The first step extracts the relevant pages according to user’s query while the second step rank pages based on their quality [8, 10]. There are many different methods proposed to rank the web pages using hyperlinks. Following section include the brief discussion about these methods.

III. Link Analysis Algorithms

Every web search engine uses its own ranking algorithm to put resulted web pages in decreasing order of relevance so that appropriate result would be on first page. Ranking algorithm uses hyperlink analysis of web graph to rank the web pages. There are many different method proposed to identify the relevancy of web pages, some users use random walk on web graph while others uses web graph structure. There are two main ranking algorithms: PageRank (uses by Google) and HITS. Both of them rank relevancy score of individual web page. Some common link analysis algorithms are discussed below:-

A. In-degree Algorithm:

This is the oldest algorithm which ranks the web pages according to the popularity of web pages [13]. Popularity of a particular web page is measured by the number of incoming links; if any page has many incoming links then the page is more popular than less number of incoming links. Mathematically, it can be shown by the following equation [13]:

\[ R(p) = Id(p) \]  \hspace{1cm} (1)

Where \( Id(p) \) denotes the in-degree of that particular page and \( R(p) \) denotes the rank of that page. This algorithm was previously used by AltaVista, HotBot and several other web search engines.

B. PageRank Algorithm

In-degree algorithm ranks the web pages according to the number of incoming links. Brin & Page [4], who is the founders of Google search engine, observed that the number of incoming links is not the only criteria to determine rank. Further, they extended this algorithm and observed that all incoming should not be given equal importance. A web page should assigned higher rank if it is referenced by many high ranked web pages. By using this concept they proposed the simplified algorithm called as PageRank algorithm which computes rank of web pages iteratively. Mathematical equation for this algorithm is stated as [4, 15]:

\[ PR(u) = c \sum_{v \in B(u)} \frac{PR(v)}{Od(v)} \]  \hspace{1cm} (2)

Where \( u \) and \( v \) are the web pages, \( PR(u) and PR(v) \) denotes PageRank values of web page \( u \) and \( v \), \( B(u) \) is the set of web pages point to page \( u \), \( Od(v) \) indicates the pages that link to page \( u \), and \( c \) is normalization constant (\( c<1 \)). Table 1 summarizes few advantage and disadvantage of PageRank algorithm [4, 6, 27, 30]:
The HITS Algorithm

C. HITS Algorithm

According to Brin and Page, PageRank algorithm is one level wait propagation scheme. Later Kleinberg proposed a different query dependent ranking algorithm which is based on two level weight propagation schemes [12]. In HITS algorithm rank of every web page is determined by two score rather than one in PageRank: First score is authoritative score $a_i$ which measures the quality of page by containing relevant or valuable information regarding query, and Second score is hub score $h_i$ which measures the quality of page by containing link of authority pages. Clearly, a good authority page is a page which pointed by many good hub pages and a good hub page is a page which points to many good authority pages. We can bipartite web graph according to hub and authority pages by making two parts of each page, where one set contains hub pages and other contains authority pages.

Computation of HITS: Computation of hits algorithm mainly done in two steps: In first steps collect the web pages on which hits algorithm applied and second steps perform hits computation in those web pages-

Step 1: HITS algorithm is applied to a subset of web graph. The subset of web pages is the collection of relevant pages depending on the query. The algorithm starts with a root set $R_q$ of the top $t$ pages from the result list of query $q$, by some content-depend rank algorithms (from a text based search engines ex- Alta-vista or HotBot).

From the root set $R_q$ a neighbour set $N_q$ is obtained by [12, 13]:

$$B_G(i) = \text{Set of at most } d \text{ nodes which point to } i \in R_q$$

$$F_G(i) = \text{Set of all nodes which pointed by } i \in R_q$$

$$N_q = R_q + B_G(i) + F_G(i)$$

Where $N_q$ denotes the neighbored set on which authority and hub score would be computed.

Step-2 Computing Authorities and Hubs: To calculate the hub and authority score for page Kleinberg defined that authority score of a page is the sum of all hub score of pages that point to it and hub score of a page is the sum of all authority score of pages that is pointed by it. There is mutual reinforcement relationship between hubs and authorities, by make use of this relationship updates the hub and authority score of each page iteratively [12].

$$a_p = \sum_{q:(p,q) \in E} h_q, \quad h_p = \sum_{q:(p,q) \in E} a_q$$

Where $h$, $a$ represents n-dimensional vector of hub and authority score of n pages.

Issues with HITS algorithm: There are several issues with HITS algorithm [31, 33, 34]

a) The main advantage of HITS algorithm is its query dependent nature i.e. it computes the rank of web pages based on user’s query. But due to query time evaluation this algorithm is not feasible with respect to time because search engines contain billion to trillion of query per day.

b) This algorithm is not capable to fight with spam web pages as it computes the rank of web pages based on hub and authority score.

c) This algorithm also suffers from topic drift problem. When it collect the web pages for root set it does not uses its own algorithm, so maybe it collects the web pages which are not relevant to query topic.
D. Summary of Improved PageRank algorithms

Several researchers have proposed improved PageRank algorithms to overcome issues as stated in table 1 with basic PageRank algorithms. Categorizations of improved PageRank algorithm based on their issues are summarized in table-2:

<table>
<thead>
<tr>
<th>Algorithm designed by Authors</th>
<th>Handling issue</th>
<th>Ideology for the method</th>
<th>Mathematical formulation of the method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sergey Brin and Lawrence Page [4]</td>
<td>Rank Sink Problem</td>
<td>Uses Random surfer model to get out of the loop by using damping factor</td>
<td>[ PR(u) = \frac{1 - d}{n} + d \sum_{v \in B(u)} \frac{PR(v)}{Dd(v)} ]</td>
</tr>
<tr>
<td>Sergey Brin and Lawrence Page [4]</td>
<td>Dangling Link</td>
<td>Remove the entire dangling link until all PageRank are calculated and after calculation simply add them.</td>
<td>Use stochastic matrix to find the dangling links</td>
</tr>
<tr>
<td>Sung Jin Kim and Sang Ho Lee [21]</td>
<td>Dangling Link</td>
<td>Define a new matrix A in which all dangling column have value ( \frac{1}{n} ) and then compute the PageRank ( (A^*) ) value by using this matrix</td>
<td>[ A^* = (1 - d)A + d[I/n]_{xx} ] Where ( A^* ) denotes PageRank matrix</td>
</tr>
<tr>
<td>ShuguangJu, Zheng Wang and Xia Lv [28]</td>
<td>Recency Search</td>
<td>Proposed a new method which combines the last modification time of web pages with the in-link and out-link weight of concerned web pages.</td>
<td>[ PR_i(p) = a \frac{1}{\text{in} + (1 - a) \sum_{q \in B(u)} PR(q) W^{-1} W^\text{decay} W^\text{in} W^{-1} W^\text{out}} W^\text{in}(u) W^\text{out}(u) ] Where ( W^\text{in} ) is the weight of incoming link and ( W^\text{out} ) is weight of outgoing link</td>
</tr>
<tr>
<td>Wengpu Xing and Ali Ghorbani [16]</td>
<td>Recency Search</td>
<td>Proposed a new method which considers the in-degree and out-degree of web pages and distributes the rank based on the importance of pages.</td>
<td>[ PR(u) = \frac{(1-d)}{m} + d \sum_{v \in B(u)} PR(v) W^\text{in}(v) W^\text{out}(u) ] Where ( W^\text{in} ) is the weight of incoming link and ( W^\text{out} ) is weight of outgoing link</td>
</tr>
<tr>
<td>Jing Wan and Si-XueBai [32]</td>
<td>Recency Search</td>
<td>Combines the time activity model (based on web site, user interest, content of web pages and web site developer) with the traditional PageRank algorithm.</td>
<td>[ PR_{\text{rec} + \text{time}} = TR \times PR ] Where TR denotes the time-activity importance vector of page. If TR value is more that means the web page importance is more at that time.</td>
</tr>
<tr>
<td>Zhou Cailan and Chen Kai [27]</td>
<td>Topic Drift Problem</td>
<td>Proposed a method which maintains user search log based on random query and click log file, which contains click time and information about URL’s. This method add an attribute with PageRank algorithm i.e. Click weight.</td>
<td>[ PR(i) = PR(i) \frac{S(i)}{F(i)} ] Where ( S(i) ) denotes click weight and ( T(i) ) denotes weight of click time</td>
</tr>
<tr>
<td>Xiaoyun Chen, BaojunGao and Ping Wen [22]</td>
<td>Topic Drift Problem</td>
<td>Proposed a method which distributed the rank of web page to its outgoing page is based on page similarity, and latent semantic model is used to determine the similarity between web pages.</td>
<td>[ LPR(v) = d \sum_{i \in B(u)} LPR(i) \times a + (1-d) ] Where ( a = \frac{1}{\sum_{i \in B(u)} \text{similarity(web page i, web page v)}} ) denotes similarity between web pages.</td>
</tr>
<tr>
<td>Chia-Chen You and Jih-Shuh Hsu [29]</td>
<td>Link Spam Problem</td>
<td>In the proposed method the PageRank score of web pages does not distributed equally by the number of out-degree, but only relevant web pages can share the score according to relevance degree.</td>
<td>[ PR(u) = (1-d) + d \sum_{\text{relevant pages}} \frac{R_i(p)}{\sum_{\text{out links of page i}}} ] Where ( R_i(p) ) is the relevancy score between web page j to i.</td>
</tr>
<tr>
<td>AndriMizal and Masada Furukawa [37]</td>
<td>Topic Drift problem</td>
<td>Proposed a link-viewer tool to observe the topic drift problem. Solve these issues in two steps: in first step find out the most relevant web pages for the root set by projection method. In next step extract the web pages which do not have many out links from the root set.</td>
<td>Uses Projection and Base-set Downsizing method to filter out the irrelevant pages from root set.</td>
</tr>
<tr>
<td>BunditManasak asemnak and ArmonRungsa wang [18]</td>
<td>Computation time and Resource</td>
<td>This method speedup the PageRank computation by partitioning the URL index files by source URL into n files of same size called ( L_i ); ( 0 \leq L_i \leq n ) and each files is executed on separate processor.</td>
<td>[ VR[v] = \frac{(1-d)}{N} + (d \times R[v]) ] Where ( R ) is rank vector which computed parallel.</td>
</tr>
<tr>
<td>Yizhou Lu, Benyu Zhang, Wens Xi et al. [35]</td>
<td>Efficient Computation</td>
<td>Proposed method is based on two attributes of web graph i.e. “Power Law Distribution” and “Hierarchical Structure”. First it finds the low scorer pages then plot the global PageRank by combining these low scorer pages.</td>
<td>Power law distribution is used to for finding the importance of web pages in web graph.</td>
</tr>
<tr>
<td>Atish, Ansur and Ei [36]</td>
<td>Scalability</td>
<td>Proposed a scalable distributed model which used Monte-Carlo method to compute PageRank of web graph.</td>
<td>Implement random surfer model based distributed algorithm to compute PageRank of web graph.</td>
</tr>
</tbody>
</table>
Table III, IV consecutively describes the comparative study of sequential algorithms and parallel algorithms. These comparisons are based on parameter like method type, additional attribute, relevancy score, input, computation time, merit and issues.

### Table 3 - Comparison of Sequential Ranking Algorithms

<table>
<thead>
<tr>
<th>Parameter Methods</th>
<th>Method</th>
<th>Attribute used</th>
<th>Input</th>
<th>Relevancy</th>
<th>Complexity</th>
<th>Merits</th>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>HITS</td>
<td>Query dependent method which computes the importance of web page based on two scores: Hub and Authority</td>
<td>Hyperlink and content structure</td>
<td>Incoming link + Outgoing link + Content of web pages</td>
<td>&lt;PR</td>
<td>&lt;O(log N)</td>
<td>More relevance than PR based on content</td>
<td>Time complexity is more than PR to give output result, Topic Drift, link Spamming Problem etc.</td>
</tr>
<tr>
<td>PageRank (PR)</td>
<td>Query independent method which sorted the result based on the relevancy score of web pages</td>
<td>Random surfer model to avoid rank sink issue</td>
<td>Incoming link</td>
<td>&gt;PR but &gt;HITS</td>
<td>O(log N)</td>
<td>Less time to give output of search results than HITS.</td>
<td>Topic Drift, Recency Search and Dangling node Problem.</td>
</tr>
<tr>
<td>Weighted PageRank (WPR)</td>
<td>Modified basic PR by computing the relevancy of web pages based on the weight of in-links and out-links</td>
<td>Weight of incoming and outgoing links</td>
<td>Incoming link + Outgoing link</td>
<td>&lt;PR</td>
<td>&lt;O(log N)</td>
<td>More quality pages than PR</td>
<td>Lacking with Recency Search Problem</td>
</tr>
<tr>
<td>Latent Semantic Model(LSM)-PageRank</td>
<td>Modify PageRank by distributing the rank of web pages to outgoing link based on the having similar contents in pages</td>
<td>LSM is used to determine similarity between pages</td>
<td>Incoming links + similarity score between pages</td>
<td>&lt;PR</td>
<td>&gt;PR</td>
<td>More relevant pages than PR by resolving Topic Drift problem</td>
<td>More complex than PR due to finding the similarity score between pages.</td>
</tr>
<tr>
<td>Feedback of user click – PageRank [27]</td>
<td>Modification to PageRank by combining PR method with the user feedback of search results.</td>
<td>HTTP log file for the feedback of the search result</td>
<td>PR + Click weight + Weight of click time</td>
<td>&lt;PR and WPR</td>
<td>&gt;PR</td>
<td>Resolve topic drift and Recency Search Problem</td>
<td>More computation time due to use of Web log Mining.</td>
</tr>
<tr>
<td>Time-Activity based PageRank [22]</td>
<td>Modify basic PR by computing the rank of web pages based on time activity vector</td>
<td>Time-Activity important level model</td>
<td>PR + Time activity important degree</td>
<td>&lt;PR</td>
<td>&gt;PR</td>
<td>Solve Recency Search Problem</td>
<td>More complex than PR.</td>
</tr>
<tr>
<td>Page Relevancy based PageRank [29]</td>
<td>Combined basic PR with the user feedback of result by calculating relevancy score between web pages.</td>
<td>Relevancy score to distribute rank of pages to other pages</td>
<td>Incoming links + relevancy score of web pages</td>
<td>&lt;PR</td>
<td>&gt;PR</td>
<td>Solve link spamming problem and convergence rate &gt; PR</td>
<td>More complex to calculate rank value than PR.</td>
</tr>
<tr>
<td>Power-Rank Algorithm [17]</td>
<td>Computes the rank score in two step: In first step it filters out low score web pages</td>
<td>Power law distribution</td>
<td>Incoming links</td>
<td>=PR</td>
<td>&lt;PR</td>
<td>Less computation time and filter out the</td>
<td>Possibility to filter out some new relevant pages</td>
</tr>
</tbody>
</table>

### IV. Comparative Study

These table comparisons are based on parameter like method type, additional attribute, relevancy score, input, computation time, merit and issues.
and in next step it computes the global importance score of pages.

<table>
<thead>
<tr>
<th>Method</th>
<th>Attribute used</th>
<th>Input</th>
<th>Relevancy</th>
<th>Time complexity</th>
<th>Merits</th>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>WTPR [28]</td>
<td>Associate basic PR with time attribute i.e. last modification time of web pages and hyperlinks</td>
<td>Log-file gives the modification time of page &amp; link</td>
<td>In-links + Time-stamp of node &amp; links</td>
<td>&lt; PR &gt; PR</td>
<td>Solve Recency search issue and suitable for dynamic changes of web</td>
<td>More complex due to dynamic nature and takes more time than PR</td>
</tr>
</tbody>
</table>

**Table 4 - Comparison of Parallel and distributed ranking algorithms**

<table>
<thead>
<tr>
<th>Parameter Algorithms</th>
<th>Method</th>
<th>Attribute used</th>
<th>Input</th>
<th>Relevancy</th>
<th>Time complexity</th>
<th>Merits</th>
<th>Issues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yuan Wang, David J.DeWitt [24]</td>
<td>Computes the rank score of pages on individual server then merge the overall rank of pages to get the result.</td>
<td>Distributed system to enable scalability</td>
<td>Intra server links, inter server links</td>
<td>Relevancy is same as PR</td>
<td>Less than Sequential PR when dealing with large data.</td>
<td>Highly Scalable</td>
<td>Merging of rank value of web pages returned by individual server is too complex</td>
</tr>
<tr>
<td>BunditManaskasem and ArnonRungsa万股 [18]</td>
<td>Divide the web graph in several parts then apply PageRank method on low cost Parallel system.</td>
<td>MIPICH v 1.2.5 for Parallel computation, and PC Cluster for storing Partitioned web graph</td>
<td>Equally partitioned web graph into each PC cluster</td>
<td>More quality pages are returned than sequential PR</td>
<td>Time complexity will decrease when no. of processors increases till the threshold time</td>
<td>Low cost parallel system to compute large web graph.</td>
<td>Takes more computation time when number of cluster increases</td>
</tr>
<tr>
<td>A. Rungsawang and B. Manaskasemak [19]</td>
<td>Computes the PageRank value on Graphical Processing Unit (GPU) system.</td>
<td>GPU used for storage and to compute rank value on Meka cluster, and Open-MPI</td>
<td>Smaller size of web graph that fit in GPU cluster</td>
<td>Approximate same as Sequential PR</td>
<td>Computation time is less than Sequential PR</td>
<td>Fast computatio with no constraint on the size of web graph</td>
<td>Takes more time on the copy of data between GPUs to CPU devices than other part of the task.</td>
</tr>
<tr>
<td>Sarma, Molla, GopalPandurangan, and EliUpfal [36]</td>
<td>Implement the fast random server based PageRank algorithm in distributed environment.</td>
<td>Monte Carlo method with distributed environment</td>
<td>Same as PR</td>
<td>More quality pages than Sequential PR and HITS method</td>
<td>$O(\sqrt{\log n/e})$ for undirected graph, $O\left(\frac{\log n}{\epsilon}\right)$ for directed graph</td>
<td>Takes to less computatio n time than other distributed algorithm</td>
<td>Used in large-scale, distributed network &amp; resource constrained system where computation time is important.</td>
</tr>
</tbody>
</table>

**V. Conclusion**

Web structure mining is one of the categorization of web mining which mines the hyperlink structure of web graph. Now days search engines are producing trillions of web pages based on user’s search query. Ranking algorithms plays a vital role to find the quality of web pages. So efficiency, relevancy and scalability of ranking algorithms are important issues. Various ranking algorithms like HITS, PageRank and their modifications were proposed. This paper has included study of PageRank algorithm and their modification based on disadvantages of basic PageRank algorithm. It also explores modified algorithms in various environments used by researchers like parallel distributed and their pros and cons. Sequential modified PageRank algorithms almost resolved issues (Rank Sink, Dangling node, topic drift etc.) with basic PageRank algorithm but they are still lacking with some other prospects like scalability, relevancy etc. In Future Parallel, Distributed, Grid environment could be used to solve Topic Drift, Recency search and scalability issues by applying on various modified PageRank algorithms.

**References**


