A Simulation Based Comparative Analysis for Web Pages and Link Queries Using Web Ranking Algorithms

Laxmi Choudhary a* and Rekha Jain a

* Department of Computer Science, Banasthali Vidyapith, Rajasthan, India.

Authors’ contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/CJAST/2023/v42i204152

ABSTRACT

In the realm of web information retrieval, the effectiveness of ranking algorithms plays a pivotal role in providing accurate and relevant search results. This simulation-based comparative analysis aims to explore the performance of two prominent ranking algorithms, namely PageRank and Weighted Page Ranking, in the context of web pages and link queries. By leveraging a comprehensive dataset comprising web pages and links, we conduct a meticulous simulation study to evaluate the effectiveness of these algorithms. Through iterative calculations and convergence analysis, we determine the rankings assigned to web pages based on their importance and connectivity within the web graph. The comparison is carried out using multiple evaluation metrics, including precision, recall, and mean average precision, to assess the algorithm’s performance in retrieving relevant web pages and handling link queries. The simulations provide valuable results of both PageRank and Weighted Page Ranking algorithms, shedding light on their applicability in various information...
retrieval scenarios. The performance of PageRank and Weighted PageRank algorithms can vary depending on the specific dataset, weighting factors and evaluation metrics used. The better algorithm in terms of results may depend on the particular goals and requirements of applications.

Keywords: Damping factor; PageRank; Web Graph; Web Mining; Weighted PageRank; WWW: World Wide Web.

1. INTRODUCTION

World Wide Web has very big warehouse of resources of data where the users can search the specified information. Now a day, with the increasing use of internet, the large amount of data stored online. This increase in availability of knowledge has resulted in problems to access the specified and relevant information for the users [7-9]. Therefore, many users use different search engines to collect the information from the web pages by all the users [2]. Most of the web content is not structured so collecting and analyzing such data is very tedious. A user uses “keywords” for required information in a search engine [11,12]. The search engine then provides set of results that are relevant to the entered keywords. But sometimes the search engines are not able to search relevant information. The search engines download, index and store hundreds of millions of web pages [4,5]. They provide results tens of millions of queries every day to the users. So Web mining and ranking mechanism becomes very important for effective information retrieval [6]. The simple architecture of a search engine is shown below with 3 components. They are Crawler, Indexer and Ranking mechanism. The crawler is also called as a robot or spider that traverses the web and downloads the web pages. The downloaded pages are sent to an indexing module that parses the web pages and builds the index based on the keywords in those pages [13-15].

![Fig. 1. Simple Architecture of a Search Engine](image)

This paper focuses on ranking algorithms to provide effective outcomes for need information in search engine. Basically PageRank algorithm and Weighted PageRank algorithms are implemented to provide better rank for information [3].

2. WEB RANKING ALGORITHMS

When a user types a query using keywords in web search engine, the query processor component match the query keywords with the index and returns the URLs of the pages to the user. But before providing the results to the user, a ranking mechanism is done by the search engines to show the most relevant pages at the top and less relevant ones at the bottom [10]. It makes the search results navigation easier and faster for the user. PageRank algorithms are based on the Web Structure Mining. Now these days it is very successful because of its PageRank algorithm and web mining techniques to order them according to the user interest. Two popular page ranking algorithms or approaches are discussed below [22].

2.1 PageRank Algorithm

PageRank algorithm is developed by Brin and Page during their Ph. D at Stanford University. PageRank algorithm is used by the famous search engine that is Google. This algorithm is the most commonly used algorithm for ranking the various pages. Working of the PageRank algorithm depends upon link structure of the web pages [23]. The PageRank algorithm is based on the concepts that if a page contains important links towards it then the links of this page towards the other page are also to be considered as important pages [16-19]. The PageRank considers the back link in deciding the rank score. If the addition of the all the ranks of the back links is large then the page then it is provided a large rank [20,21]. Therefore, PageRank provides a more advanced way to compute the importance or relevance of a web page than simply counting the number of pages that are linking to it.

If a back-link comes from an important page, then that back-link is given a higher weighting than those back-links comes from non-important
pages. In a simple way, link from one page to another page may be considered as a vote. However, not only the number of votes a page receives is considered important, but the importance or the relevance of the ones that cast these votes as well. We assume page A has pages T₁...Tₙ which point to it (i.e., are citations or incoming links). The variable d is a damping factor, which can be set between 0 and 1. We usually set the value of d to 0.85.

Also C (A) is defined as the number of links going out of page A. The Page Rank of a page A is given by the following (1):

\[ PR(A) = (1-d) + d \frac{PR(T_1)}{C(T_1)} + \ldots + d \frac{PR(T_n)}{C(T_n)} \] (1)

The damping factor usually sets it to 0.85, is used to stop the other pages having too much influence, this total vote is damped down by multiplying it by 0.85. One important thing is noted that the page ranks form a probability distribution over web pages, so the sum of all web pages’ page ranks will be one and the damping factor is the probability at each page the random surfer will get bored and request another random page. Another simplified version of PageRank is given by:

\[ PR(N) = \sum_{m \in B_n} \frac{PR(M)}{L(M)} \] (2)

Where the page rank value for a web page u is dependent on the page rank values for each web page v out of the set Bₙ (This set contains all pages linking to web page N), divided by the number L (M) of links from page M. An example of back link is shown in Fig. 3 below. N is the back link of M & Q and M & Q are the back links of O.

![Fig. 3. Hyperlink Structure of Four Pages](image)

Let us assume the initial PageRank as 1 and do the calculation. The damping factor d is set to 0.85.

\[ PR(A) = (1-d) + d \frac{PR(B)}{C(B)} + \frac{PR(C)}{C(C)} + \frac{PR(D)}{C(D)} \]
\[ = (1-0.85) + 0.85(1/3+1/3+1/1) \]
\[ = 1.5666667 \] (3)

\[ PR(B) = (1-d) + d \frac{PR(A)}{C(A)} + \frac{PR(C)}{C(C)} \]
\[ = (1-0.85) + 0.85(1.5666667/2+1/3) \]
\[ = 1.0991667 \] (4)

\[ PR(C) = (1-d) + d \frac{PR(A)}{C(A)} + \frac{PR(B)}{C(B)} \]
\[ = (1-0.85) + 0.85(1.5666667/2+1.0991667/3) \]
\[ = 1.127264 \] (5)

\[ PR(D) = (1-d) + d \frac{PR(B)}{C(B)} + \frac{PR(C)}{C(C)} \]
\[ = (1-0.85) + 0.85(1.5666667/2+1.0991667/3) \]
\[ = 0.7808221 \] (6)

For the second iteration by taking the above PageRank values from (3), (4), (5) and (6). The second iteration PageRank values are as following:

\[ PR(A) = 0.15 + 0.85((1.0991667/3)+(1.127264/3)+(0.7808221/1)) \]
\[ = 1.4445208 \] (7)

\[ PR(B) = 0.15 + 0.85((1.4445208/2)+(1.127264/3)) \]
\[ = 1.0833128 \] (8)

\[ PR(C) = 0.15 + 0.85((1.4445208/2)+(1.0833128/3)) \]
\[ = 1.07086 \] (9)

\[ PR(D) = 0.15 + 0.85((1.0833128/3)+(1.07086/3)) \]
\[ = 0.760349 \] (10)
The importance is assigned in terms of weight values to the incoming and outgoing links and are denoted as \( W_{in} \) and \( W_{out} \) respectively. The weight of link \((m,n)\) as shown in equation (11) is the weight of link \((m,n)\) calculated based on the number of incoming links of page \( n \) and the number of incoming links of all reference pages of page \( m \).

\[
W_{in} = \frac{I_p}{\sum I_p}
\]

\[
W_{out} = \frac{O_p}{\sum O_p}
\]

Where \( I_p \) and \( O_p \) are the number of incoming links of page \( p \) and page \( p \) respectively. \( R(m) \) denotes the reference page list of page \( m \).\( W_{out} \) is as shown in (12) 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 \( O_p \) are the number of outgoing links of page \( n \) and \( p \) respectively. The formula as proposed for the WPR is as shown in (13) which is a modification of the PageRank formula.

\[
WPR(n) = (1-d) + \sum_{m \in B(n)} W_{in}(m,n) W_{out}(m,n)
\]

2.2 Weighted Page Rank Algorithm

Weighted PageRank Algorithm is proposed by Wenpu Xing and Ali Ghorbani which is modification of the original PageRank algorithm. WPR decides the rank score based on the popularity of the pages by taking into consideration the importance of both the in-links and out-links of the pages. This algorithm provides high value of rank to the more popular pages and does not equally divide the rank of a page among it’s out-link pages. Every out-link page is given a rank value based on its popularity. Popularity of a page is decided by observing its number of in links and out links.

During the computation of 34th iteration, the average of all the web pages is 1. Some of those values are shown in Table 1. The table with the graph is shown in the simulation results section.

### Table 1. Iterative calculation for pagerank [1]

<table>
<thead>
<tr>
<th>Iteration</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1.5666667</td>
<td>1.0991667</td>
<td>1.127264</td>
<td>0.7808221</td>
</tr>
<tr>
<td>3</td>
<td>1.4445208</td>
<td>1.0833128</td>
<td>1.07086</td>
<td>0.760349</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>17</td>
<td>1.3141432</td>
<td>0.9886763</td>
<td>0.9886358</td>
<td>0.7102384</td>
</tr>
<tr>
<td>18</td>
<td>1.313941</td>
<td>0.9885384</td>
<td>0.98851085</td>
<td>0.71016395</td>
</tr>
<tr>
<td>19</td>
<td>1.3138034</td>
<td>0.98844457</td>
<td>0.98842573</td>
<td>0.7101132</td>
</tr>
</tbody>
</table>

WPR calculation calculated for the same hyperlink structure as shown in Fig. 5. The WPR equation for Page A, B, C and D are as follows:

\[
WPR(A) = (1-d) + d \sum WPR(B) W_{in}^{(B,A)} W_{out}^{(B,A)} + WPR(C) W_{in}^{(C,A)} W_{out}^{(C,A)} + WPR(D) W_{in}^{(D,A)} W_{out}^{(D,A)}
\]

So for getting the value of WPR(A), before it we will calculate the value of incoming links and outgoing links weight as bellow:

\[
W_{in}^{(B,A)} = \frac{I_B}{I_A + I_C} = \frac{3}{3+2} = \frac{3}{5}
\]

\[
W_{out}^{(B,A)} = \frac{O_B}{O_A + O_C + O_D} = \frac{2}{2+3+1} = \frac{2}{6}
\]
W^{in}_{(C,A)} = I_A/(I_A+I_B) = 3/(3+2) = 3/5 (17)

W^{out}_{(C,A)} = O_A/(O_A+O_B+O_D) = 2 / (2+3+1) = 2/6 = 1/3 (18)

W^{in}_{(D,A)} = I_A/(I_B+I_C) = 3/(2+2) = 3/4 (19)

W^{out}_{(D,A)} = O_A/O_A = 2 / 2 = 1 (20)

W^{in}_{(A,B)} = I_B/(I_B+I_C+I_D) = 2/(2+2+2) = 2/6 = 1/3 (25)

W^{out}_{(A,B)} = O_B/(O_B+O_C) = 3/(3+3) = 3/6 = 1/2 (26)

W^{in}_{(C,B)} = I_B/(I_A+I_B) = 2/(3+2) = 2/5 (27)

W^{out}_{(C,B)} = O_B/(O_A+O_B+O_D) = 2/(2+3+1) = 2/6 = 1/3 (28)

Now these inlinks and outlinks weight, equations number (15, 16, 17, 18, 19, 20) are put in the equation (14) to calculate the weighted rank of the nodes A, B, C, and D as following:

\[ WPR(B) = (1 - d) + d \sum WPR(A) W^{in}_{(A,B)} W^{out}_{(A,B)} + WPR(C) W^{in}_{(C,B)} W^{out}_{(C,B)} \] (21)

\[ WPR(C) = (1 - d) + d \sum WPR(A) W^{in}_{(A,C)} W^{out}_{(A,C)} + WPR(B) W^{in}_{(B,C)} W^{out}_{(B,C)} \] (22)

\[ WPR(D) = (1 - d) + d \sum WPR(B) W^{in}_{(B,D)} W^{out}_{(B,D)} + WPR(C) W^{in}_{(C,D)} W^{out}_{(C,D)} \] (23)

For \( WPR(A) \) calculation the value of d is set to 0.85 (standard value) and the initial values of \( WPR(B) \), \( WPR(C) \) and \( WPR(D) \) is considered 1, so calculation for 1st iteration as follows:

\[ WPR(A) = (1 - 0.85) + 0.85 (1.127 * 1/ 3 * 1/2 + 1*2/5 *1/2) = 0.4989 \] (29)

\[ WPR(B) = (1 - 0.85) + 0.85 ((1.127 * 1/ 3 * 1/2) + (0.499 * 2 / 5 *1/2)) = 0.392 \] (30)

\[ WPR(D) = (1 - 0.85) + 0.85 ((0.499 * 1/ 2 * 1) + (0.392 * 2 / 5 *1/3)) = 0.406 \] (31)

The values of \( WPR(A) \), \( WPR(B) \), \( WPR(C) \) and \( WPR(D) \) are shown in equations (24), (29), (30) and (31) respectively. In this, \( WPR(A) > WPR(B) > WPR(D) > WPR(C) \). This results shows that the Weighted PageRank order is different from PageRank.

For the same above example the iterative computation of weighted page rank is computed. The some Weighted PageRank as shown in Table 2. The table values with the chart are shown in the simulation results section.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.1275</td>
<td>0.47972</td>
<td>0.3912</td>
<td>0.19935</td>
</tr>
<tr>
<td>2</td>
<td>0.425162</td>
<td>0.27674</td>
<td>0.25727</td>
<td>0.18026</td>
</tr>
<tr>
<td>3</td>
<td>0.355701</td>
<td>0.244128</td>
<td>0.24189</td>
<td>0.177541</td>
</tr>
<tr>
<td>4</td>
<td>0.34580</td>
<td>0.247110</td>
<td>0.239808</td>
<td>0.17719</td>
</tr>
<tr>
<td>5</td>
<td>0.34454</td>
<td>0.23957</td>
<td>0.23953</td>
<td>0.17714</td>
</tr>
<tr>
<td>6</td>
<td>0.34438</td>
<td>0.23950</td>
<td>0.23950</td>
<td>0.17714</td>
</tr>
<tr>
<td>7</td>
<td>0.34436</td>
<td>0.23950</td>
<td>0.23949</td>
<td>0.17714</td>
</tr>
</tbody>
</table>
So we can easily differentiate the WPR from the PageRank, categorized the resultant pages of a query into four categories based on their relevancy to the given query.

4. DISCUSSION

The program is developed for the PageRank and Weighted PageRank algorithm using advance java language and apache tomcat server tested on an Intel Core (2 duo) with 4GB RAM machine. The input is shown in Fig. 4, the user can enter any type and any size of directed graph which contains the number of nodes that behaves as a web pages, the number of incoming and outgoing links of the nodes. After press on ok1 button, matrix of entered directed graph appears beside graph on window. Now user wants the rank scores of web pages then click on submit button to calculate PageRank and Weighted PageRank comes as an output with iterative method. The output of PageRank is shown in Fig. 4 and PageRank values is also shown in Table 3. In this simply PageRank and Weighted PageRank is calculated then their values retrieved and designed the chart of that values for web pages and compared those ranks to get higher rank web page.

![Image](image_url)

**Table 3. Simulative iterations of pagerank for a web graph [1]**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1.4445208</td>
<td>1.0833128</td>
<td>1.07086</td>
<td>0.760349</td>
</tr>
<tr>
<td>5</td>
<td>1.3766</td>
<td>1.0313</td>
<td>1.0272</td>
<td>0.7332</td>
</tr>
<tr>
<td>7</td>
<td>1.34284</td>
<td>1.00825</td>
<td>1.00638</td>
<td>0.720813</td>
</tr>
<tr>
<td>9</td>
<td>1.3271</td>
<td>0.9975</td>
<td>0.9966</td>
<td>0.71502</td>
</tr>
<tr>
<td>11</td>
<td>1.319839</td>
<td>0.99256</td>
<td>0.99215</td>
<td>0.712336</td>
</tr>
<tr>
<td>13</td>
<td>1.316449</td>
<td>0.990249</td>
<td>0.990061</td>
<td>0.711088</td>
</tr>
<tr>
<td>15</td>
<td>1.314874</td>
<td>0.989175</td>
<td>0.98908</td>
<td>0.710507</td>
</tr>
<tr>
<td>17</td>
<td>1.31414</td>
<td>0.988676</td>
<td>0.988635</td>
<td>0.710238</td>
</tr>
<tr>
<td>19</td>
<td>1.31380</td>
<td>0.988444</td>
<td>0.988425</td>
<td>0.710113</td>
</tr>
</tbody>
</table>

![Image](image_url)

**Fig. 4. A Web Graph with Matrix for PageRank and Weighted PageRank**

**Fig. 5. Page rank convergence chart [1]**
Table 4. Simulative iterations of weighted pagerank for a web graph [1]

<table>
<thead>
<tr>
<th>Iteration</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1.1275</td>
<td>0.47972</td>
<td>0.3912</td>
<td>0.19935</td>
</tr>
<tr>
<td>3</td>
<td>0.425162</td>
<td>0.27674</td>
<td>0.25727</td>
<td>0.18026</td>
</tr>
<tr>
<td>4</td>
<td>0.355701</td>
<td>0.244128</td>
<td>0.24189</td>
<td>0.177541</td>
</tr>
<tr>
<td>5</td>
<td>0.34580</td>
<td>0.247110</td>
<td>0.239808</td>
<td>0.17719</td>
</tr>
<tr>
<td>6</td>
<td>0.34454</td>
<td>0.23957</td>
<td>0.23953</td>
<td>0.17714</td>
</tr>
<tr>
<td>7</td>
<td>0.34438</td>
<td>0.23950</td>
<td>0.23950</td>
<td>0.17714</td>
</tr>
<tr>
<td>8</td>
<td>0.34436</td>
<td>0.23950</td>
<td>0.23949</td>
<td>0.17714</td>
</tr>
</tbody>
</table>

Fig. 6. Weighted page rank convergence chart [1]

5. CONCLUSION


A typical search engine should use web page ranking techniques based on the specific needs of the users because the ranking algorithms provide a definite rank to resultant web pages. After going through this exhaustive analysis of algorithms for ranking of web pages against the various parameters such as methodology, input parameters, relevancy of results and importance of the results, it is concluded that existing algorithms have limitations in terms of time response, accuracy of results, importance of the results and relevancy of results. This paper also concludes the introduction of Web mining and the three areas of Web mining used for Information Retrieval. The main purpose is to inspect the important page ranking based algorithms used for information retrieval and compare those algorithms. An efficient web page ranking algorithm should meet out these challenges efficiently with compatibility with global standards of web technology. The work applies the PageRank program in the Web, calculates PageRank values by Rage Rank algorithm and weighted page rank values using Weighted PageRank algorithm. Finally, simulation results are shown for the PageRank and Weighted PageRank algorithm and compares to web page’s value in chart that shows which is better depends on the specific requirements and characteristics of applications. Here are some considerations to help you make an informed decision: link structure emphasis, additional factors such as content relevance, user preferences, or link quality, that significantly impact the importance of web pages, weighted PageRank may provide more accurate and personalized rankings.

DISCLAIMER

This manuscript is an extended version of the previously published article [Published by the same author]:

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES


Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle5.com/review-history/103029

© 2023 Choudhary and Jain; This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.