American Journal of Information Science and Technology
2025, Vol. 9, No. 1, pp. 15-23
https://doi.org/10.11648/j.ajist.20250901.12

SciencePG
Science Publishing Group

Research Article

Comparative Analysis of Page Ranking Algorithms for Efficient Information Retrieval

Zahir Edrees[1, 2, *], Henda Juma[2]

[1]Department of Software Engineering, International Science and Technology University, Warsaw, Poland
[2]Department of Computer Engineering, Faculty of Engineering- Karabuk University, Karabuk, Turkey

Abstract

Search engines have become crucial tools today, providing users with access to vast amounts of information. At the core of search engine functionality lies the ranking algorithm, which is responsible for determining the relevance and order of web pages returned in response to user queries. Ranking algorithms play a critical role in ensuring that users receive the most relevant and useful results, particularly in the face of exponentially growing web content. This paper provides an in-depth analysis of PageRank algorithms, focusing on their significance in information retrieval systems. The study begins with an overview of the foundational PageRank algorithm developed by Google, detailing its reliance on hyperlink structures to rank web pages. The limitations of the original algorithm, such as its inability to consider page content relevance and dynamic updates, are explored. In response to these limitations, the paper examines advanced ranking methods, including the Weighted PageRank (WPR), Hyperlink-Induced Topic Search (HITS), and the Stochastic System Analysis Approach (SALSA). Each of these algorithms is analyzed in terms of efficiency, response time, scalability, and effectiveness. Additionally, the paper investigates recent enhancements in ranking methods that address the evolving needs of modern search engines, such as personalized search and semantic relevance. Experimental comparisons are conducted to evaluate the performance of these algorithms on large-scale datasets. Key metrics, including time response, computational efficiency, and relevance accuracy, are used to compare and rank the algorithms. The findings provide valuable insights into the strengths and weaknesses of different PageRank methods, contributing to the development of more efficient and effective information retrieval systems.

Keywords

Information Retrieval, Web Page Ranking, Effectiveness, Efficiency, Search Engines

1. Introduction

Information retrieval (IR) is the process to find material (documents) of an unstructured nature [1]. one of benefits of IR system is that it does not just get documents [11]. It gives the researcher the Uniform Resource Locator of these documents (URL). IR systems must define or handle some problems: Firstly giving the users related information according to his searching (effectiveness of IR system) [17], and secondly, minimize the response time to get users requirements (efficiency of IR system), based on these two criteria's the user will decide which search engine can use it. The main difference between information retrieval and data retrieval is that we use artificial query in data retrieval but in information

[*]Corresponding author: zahir-mohamed2010@hotmail.com (Zahir Edrees)

Received: 26 December 2024; **Accepted:** 13 January 2025; **Published:** 11 February 2025

retrieval, we use natural language, also the query may be incomplete in information retrieval but must be complete in data retrieval, we will explain the important components of the web IR system [1]. It is shown in Figure 1:

1. Browse documents by using Uniform Resource Locator
2. (URL). – Crawling process.
3. Building the index of the documents – Indexing process
4. User search about information - Querying process.
5. Retrieves for documents that are related to the user requirements- Ranking process.
6. Users give IR system Feedback about satisfaction or not.

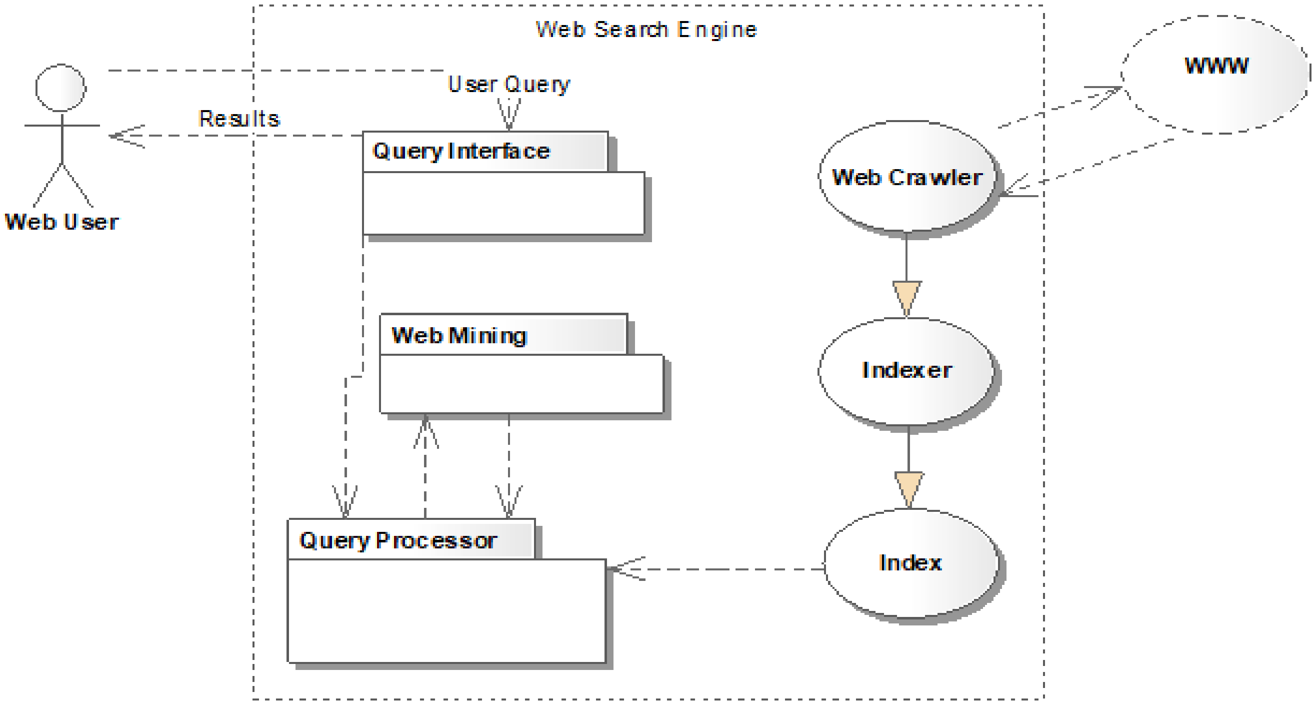


Figure 1. *Important processes of web information retrieval.*

This paper is organized as the following sections; Section 1 is about the introduction. Section 2 discusses the related work which will discusses some of the available PageRank algorithms for Information Retrieval; section 3 compares between these algorithms according to different criteria. Section 4 concludes this paper.

2. Literature Review

The literature review provides an in-depth exploration of the current state of research in Information Retrieval (IR). It covers ranking algorithms relevant to enhancing IR effectiveness and efficiency for web searching.

2.1. Technology

Page ranking is a technique used to rank web pages based on their degree of importance [2, 13, 14]. Various algorithms employ different criteria for page ranking. Some algorithms rely on the content of the pages, while others utilize the link structure. The first classification is content-based page ranking, which depends on the textual and contextual information of the pages [3, 12]. The second classification is connectivity-based page ranking, which evaluates the structure and relationships of links between pages (link-based ranking), as illustrated in Figure 2 [4, 15].

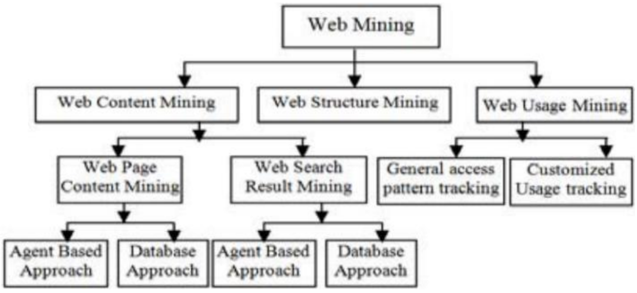


Figure 2. *Classification of Web Mining.*

2.2. Page Ranking Algorithms

In this section will discuss some issues related to information retrieval. Starting with description for some available algorithms and models, which is one of major challenge in this area. Then go through the experiments carried out in information retrieval.

2.2.1. PageRank Algorithm

PageRank is one of commonly algorithms used in ranking [3, 11], the Google used it in ranking process, it link based algorithm. Formula of Page Ranking algorithm calculation [16]:

$$PR(A)=(1-d) + d(C(T1)\ PR(T1) + \cdots + C(Tn)PR(Tn)) \quad (1)$$

Here, n is the number of pages accounted, d is dampening factor equlas 0.85 and it is used to represent pages that have no inlinks to give it some Page Rank value. C (T1), C (T2)... C (Tn) are the number of outlinks of pages, T1, T2… Tn are links to the page A.

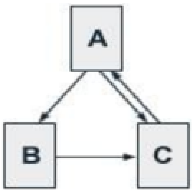


Figure 3. Example of Hyperlinked Structure.

Figure 3, is example to shows that how page rank algorithm works, if we have websites that includes A, B and C, page A links to B, C; and B links to the C and C links to the A. to calculate the page rank through these steps.

$$PR\ (A) = 0.15 + 0.85\ PR\ (C)$$

$$PR\ (B) = 0.15 + 0.85\ (PR\ (A)/2)$$

$$PR\ (C) = 0.15 + 0.85\ (PR\ (A)/2+PR\ (B))$$

Table 1. Iteration to calculate Page Rank of each page.

Iteration	PR (A)	PR (B)	PR (C)
0	1	1	1
1	1.00	0.58	1.06
2	1.05	0.60	1.10
3	1.09	0.61	1.13
4	1.11	0.62	1.15
5	1.13	0.63	1.17
6	1.14	0.63	1.17
7	1.14	0.63	1.17

After the iteration of calculation, we will get the below result of page ranking PR (A) = 1.14, PR (B) = 0.63, PR (C) = 1.17, we can see that PR (C) > PR (A) > PR (B). Page rank uses link structure, this a reason for the result that produced is not relevant to the user’s query this problem is called theme drift.

2.2.2. Weighted PageRank Algorithm

The (WPR) Algorithm [7] is an improved of the Page Rank, to rank web pages it using the weight of inlinks and the weight of outlines gives better results as compared to Page RankAlgorithm to calculate the weight of inlinks Eq. 2 and outlinks of the web pages Eq. 3 by using the following formula [2].

$$w^{in}_{(u,v)} = \frac{I_u}{\Sigma_{p \epsilon R(v)} I_p} \quad (2)$$

Where I_u is the sum of links that coming to page u, I_p is the sum links that coming to page p.

$$w^{out}_{(u,v)} = \frac{O_u}{\Sigma_{p \epsilon R(v)} O_p} \quad (3)$$

Here is O_u represents the sum of links that outgoing of page u, and O_p is the sum of links that outgoing of page p according to these changes the formula of page rank Eq. 1 will be changed as Eq. 4.

$$WPR(u)^{=(1\text{-}d)\ +d} \sum WPR(v)\ w^{in}_{(u,v)} w^{out}_{(u,v)} \quad (4)$$

2.2.3. HITS Algorithm

The Hyperlink-Induced Topic Search (HITS) is a link recognition algorithm developed by Jon Kleinberg that scores Web pages. It decides the value of the content of the website, its authority and its hub value which calculates the value of the links to other pages. HITS is a search algorithm that separates the web into links and related pages by its full processing [10]. As with PageRank, HITS is an iterative web-based document connection algorithm. It has several variations. It is based upon demand, i.e. search words are decided by the outcomes of (Hubs and Authority) analyzes. As a corollary, the associated hit on the performance accompanying query time processing is carried out during query time and not at indexing time. As was the case for PageRank, not all records. Steps involved in HITS algorithm: The first move is to locate a variety of web sites [8]. Thus for example, the search engine thinks this may be an interesting page for the web pages which contain the question string on the web page document [9]. They are possible sources, theoretically relevant sites despite the user's request. So let us assume that these are pages that represent at least one of the pages on the base, in this case the nodes E, F, G, D so H. All of this node collection is considered the foundation group, irrespective of whether it was in the root and how it meant about something in the heart.

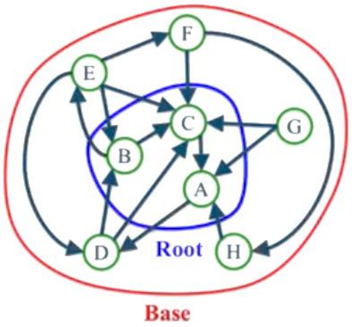


***Figure 4**. The Base.*

And thus this is the network we can use to locate the relevant web sites. Now on this network we must run the HITS algorithm. Just like PageRank, the HITS algorithm operates by calculating k iterations just maintaining the scoring track for each node. Then the System Upgrade Law is the other norm. This is kind of symmetric. The hub mark of each node would then be the sum of the authority mark of any node to which it refers. Then we will have to normalize, as these scores tend to develop and to rise, then we must normalize the hub and the authority value with each iteration. And then, for example, you take the authority score of each node, say j, and divide the authority score of j by the number of all authorities around the network. The authority score for j is the quantity of authority. And then eight times we will replicate this cycle. it may seem at this stage a little vague, so let's take an illustration and see how it functions exactly. So we can use the network we have already learned about and measure two iterations of the HITS network algorithm. As in PageRank, we would have to log the old scores in order to determine the new scores.

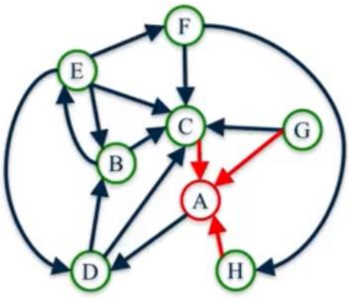
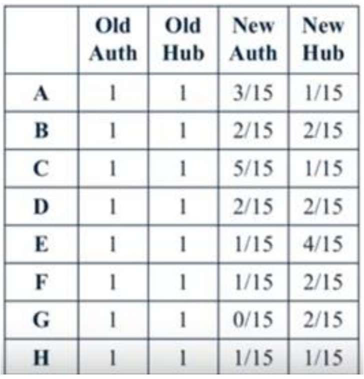


***Figure 5**. Scores of node A.*

We look at node A and we analyze which nodes refer to node A in order to find out what A's current score is and what C, G, and H lead to A turned out to be. And because C, G, and H have all 1 hub value, so the new A value would be 3. Now H has a server, and it has a new rank of authority 1. All right, now we're going to switch to the latest center ratings. It would be quite close, except now we can look at the auth tier, rather than the level of growing node. And so, for examples, A has an auth grade 1, heading towards D. So we will now glance at the old score of power. And again, every node has an old authority score of 1 because this is our first move. And D does, and A would have a fresh hub ranking of 1. what we have to do is determine the degree of credibility for growing node, then the platform for new system ratings. We should standardize next. Then we have to include the scores of control and apply the scores for the center in order to normalize. Both should sum up to 15 in this situation. This is not accurate that they would add up to the same value with each length, but they will do the first. They sum up to fifteen in this situation. So all the ratings will be separated by 15. And then we normalize the ratings if we do so. And now our old scores will be the latest authority and turn. So we're prepared to head into the next iteration that could be much more challenging because not all nodes have the same authority so hub ranking, and we have to be cautious with our nodes.

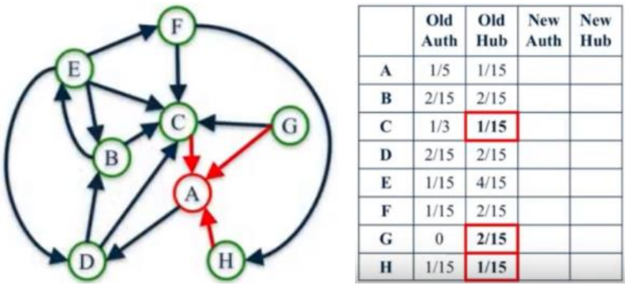


	Old Auth	Old Hub	New Auth	New Hub
A	1	1	3/15	1/15
B	1	1	2/15	2/15
C	1	1	5/15	1/15
D	1	1	2/15	2/15
E	1	1	1/15	4/15
F	1	1	1/15	2/15
G	1	1	0/15	2/15
H	1	1	1/15	1/15

***Figure 6**. The scores of controls.*

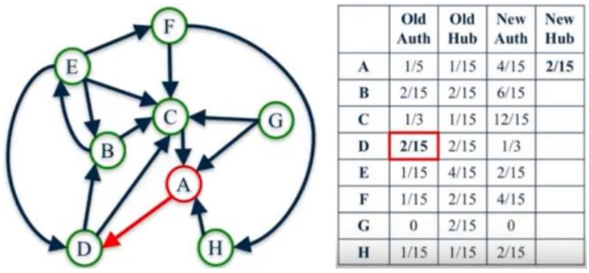
Let's begin with Node A, then. We want to work out the latest A score and so need to decide which nodes mean A. Therefore, all leads to A, C, G, and H. And now the node values of C, G and H, 1/15, 2/15 and 1/15 have to be looked at.

Then we bring them up to 4/15, and this would be a fresh ranking from the authority. So then C has five domains, E, F, G, B, D, which are some old domains I show. Therefore and now again, looking at A, we don't look at the number, we don't look at who points at that, but who points at that. And now the prior authority ratings of such nodes must be taken into account. So, in this situation, a point to D and D have an old ranking of 2/15, and that would be the current hub ranking for A. points to D.



	Old Auth	Old Hub	New Auth	New Hub
A	1/5	1/15		
B	2/15	2/15		
C	1/3	**1/15**		
D	2/15	2/15		
E	1/15	4/15		
F	1/15	2/15		
G	0	**2/15**		
H	1/15	**1/15**		

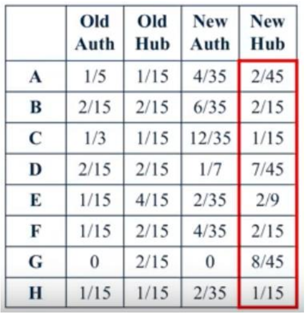
***Figure 7**. The scores of A.*



	Old Auth	Old Hub	New Auth	New Hub
A	1/5	1/15	4/15	2/15
B	2/15	2/15	6/15	
C	1/3	1/15	12/15	
D	2/15	2/15	1/3	
E	1/15	4/15	2/15	
F	1/15	2/15	4/15	
G	0	2/15	0	
H	1/15	1/15	2/15	

Figure 8. The scores of controls.

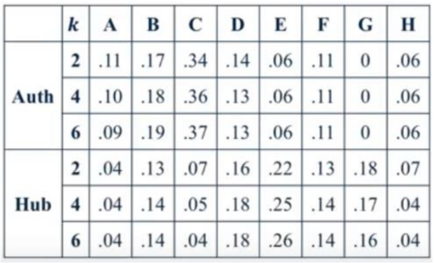
With the other nodes we will start doing this and discover different center results. We will standardize next. We would then include all the scores of the authority. They are up to 35/15 in this situation. And any new authority ranking has to be split by 35/15. If we do so, the standardized values are modified. And for the nodes we do the same. Then we apply the centre labels for all nodes, which in this case corresponds to 3. So any new Hub score will be divided by 3, and these are the normalized scores that are modified.



	Old Auth	Old Hub	New Auth	New Hub
A	1/5	1/15	4/35	2/45
B	2/15	2/15	6/35	2/15
C	1/3	1/15	12/35	1/15
D	2/15	2/15	1/7	7/45
E	1/15	4/15	2/35	2/9
F	1/15	2/15	4/35	2/15
G	0	2/15	0	8/45
H	1/15	1/15	2/35	1/15

Figure 9. The scores of controls.

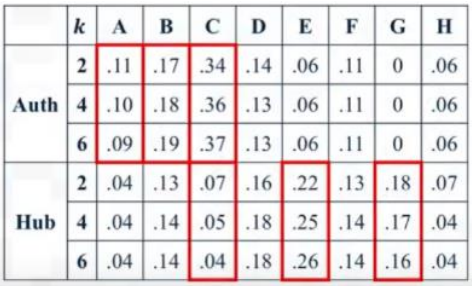
From two variations of the HITS algorithm these are our final latest authority and center results. Unlike PageRank, we ask what happens to the results as we start to constantly iterate the algorithm. Would it become a common value, here are the scores we determined only for k = 2, 2 iterations; these are the scores of authority. And what's likely as we head to four iterations? that's what you'd expect. So that's what you'd see, 6 variations. And for the center performs the same stuff. Those are the attributes we have already found for 2 iterations, so we will decide what they are with 4 so 6 iterations.



	k	A	B	C	D	E	F	G	H
Auth	2	.11	.17	.34	.14	.06	.11	0	.06
	4	.10	.18	.36	.13	.06	.11	0	.06
	6	.09	.19	.37	.13	.06	.11	0	.06
Hub	2	.04	.13	.07	.16	.22	.13	.18	.07
	4	.04	.14	.05	.18	.25	.14	.17	.04
	6	.04	.14	.04	.18	.26	.14	.16	.04

Figure 10. Latest authority and centre results.

And what you find here is that these ratings don't shift for certain domains, but adjust for others. Here I highlight the nodes that modified the authority or center score after 6 iterations. For starters, Node B here begins with a 15-score authority. It goes to 18 following four iterations. It's going to move to 19 following six iterations. So, is this B score going to expand more at any stage or is it going to saturate? And then I show you here what happens to node B at the center and authority value if we proceed with this version.



	k	A	B	C	D	E	F	G	H
Auth	2	.11	.17	.34	.14	.06	.11	0	.06
	4	.10	.18	.36	.13	.06	.11	0	.06
	6	.09	.19	.37	.13	.06	.11	0	.06
Hub	2	.04	.13	.07	.16	.22	.13	.18	.07
	4	.04	.14	.05	.18	.25	.14	.17	.04
	6	.04	.14	.04	.18	.26	.14	.16	.04

Figure 11. The nodes that modified the authority.

There is a set of variations of this plot on the x-axis, then the y-axis has the authority and center values for node B. The authority and hub scores are gradually converging into a common rating for most of the networks, ask grows larger. So that is a special attribute, in this situation, which you consider ask grows bigger and bigger.

The nodes with the lowest ranking are B and C, and the nodes with the highest rating are D and E. So when you bear in mind the network's initial configuration B, C, and A, those root nodes were the ones most significant, and it turns out that B and C have the highest ratings. And the main hubs, the nodes that lead to nodes that are especially insignificant or unique, are D and E. So if all point A and D, B and C, and other nodes are found.

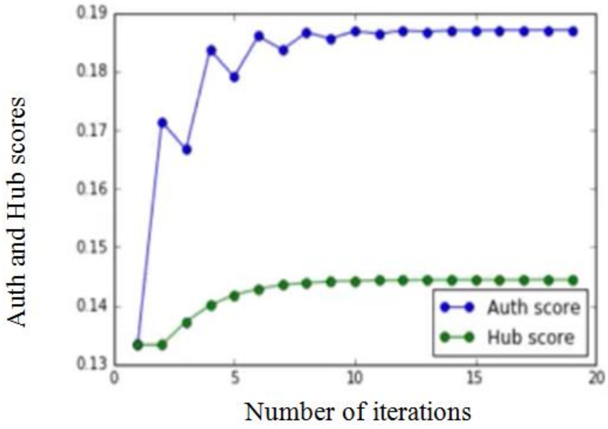
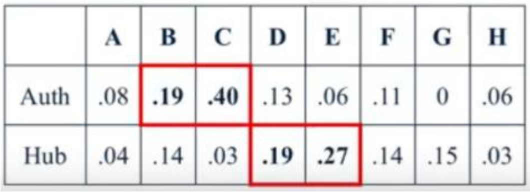


Figure 12. The number of iterations.



	A	B	C	D	E	F	G	H
Auth	.08	**.19**	**.40**	.13	.06	.11	0	.06
Hub	.04	.14	.03	**.19**	**.27**	.14	.15	.03

Figure 13. *The lowest ranking nodes.*

The lower graded nodes are B and C, while the highest graded nodes are D and E. When you find the original B, C, and A configurations of the network, these root nodes are the most critical, with B and C being the largest. And the principal hubs, the nodes which lead to relatively insignificant and special nodes, are D and E. Then if you consider both points A, D, B, C, and other nodes. For brief, the HITS algorithm starts by constructing the root set of web sites to a basis set for the network layout. For short, it expands to a base level. Then HITS assigns to each node in the network an authority score and a center score. And nodes with incoming edges are assumed to be weak authorities, and instead nodes with outcoming edges are called strong hubs for weak authorities.

2.2.4. Stochastic Approach for Link-Structure Analysis (SALSA)

A web page rating algorithm developed by R is Stochastic System Analysis Approach (SALSA). S. and Lempel. Moran can give high scores to web pages depending on the amount of hyperlinks between them. SALSA is rooted in the following two other link-oriented rankings: HITS and PageRank: Like HITS, the algorithm provides two values for each web page: one center and one authority. SALSA encourages the following outcomes. An office is a page that is far more important than other pages to a single subject and a portal is a website with a number of ties to government [5], SALSA functions like Classics, on a concentrated subsection that depends on the topic. This centered sub gram is obtained first by seeking a collection of pages specific to a particular subject, then by inserting web pages closely connected to it and using pages directly linked from the top-n [6].

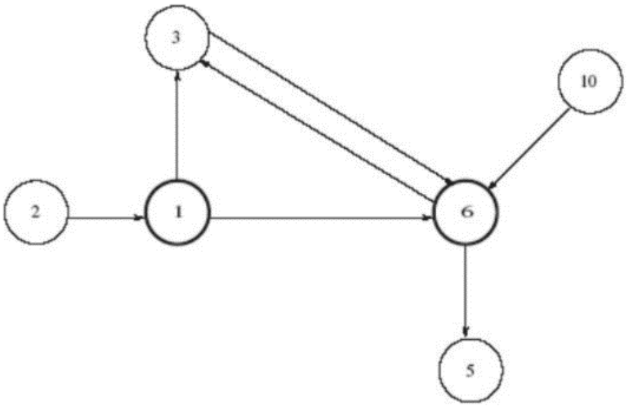


Figure 14. *The SALSA community maps.*

SALSA's similarities to HITS and PageRank:

1. SALSA uses authority and hub ranking to equate SALS to HITS and PageRank.
2. Using the authority and center sites and connections, SALSA builds community maps.

SALSA's differences between HITS and PageRank:

1. The SALSA method generates bi-part map of the authority pages and the middle pages of the community table. The SALSA method creates a two section diagram.
2. A package comprises port pages
3. One set includes portal pages
4. All sets of Neighborhood Graph G will be used on each page:

Neighborhood Bipartite Network G Network N: Markov Chains:

1. Three graph G matrices. Four matrices.
2. Markov Matrix H center series.
3. Markov Matrix A chain control.

$$V_h = \{1, 2, 3, 6, 10\},$$
$$V_a = \{1, 3, 5, 6\}.$$

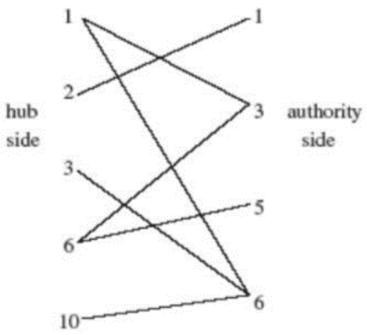


Figure 15. *The SALSA bi-part map.*

SALSA should fit into Matrix H and A will come from the HITS- and PageRank-type adjacency-matrix L, HITS-type L-matrix, PageRank-type weight-type L-matrix, SALSA-like column weighting and row-weighted. H and A could be determined by allowing Lr to be L with each non-zero row, and allowing Lc to be L with each non-zero column split by its column number.

$$\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}, \quad \mathbf{L}_r = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array},$$

$$\mathbf{L}_c = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \\ 6 \\ 10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 0 & 0 & \frac{1}{2} & 0 & \frac{1}{3} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \end{pmatrix} \end{array}.$$

H, the center matrix of SALSA consists of non-null columns and rows of L_rL_cT. A comprises of non-zero rows and columns of $L_c TL_r$, the SALSA matrix for authority

$$\mathbf{L}_r\mathbf{L}_c^T = \begin{array}{c} \\ 1\\2\\3\\5\\6\\10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} \frac{5}{12} & 0 & \frac{2}{12} & 0 & \frac{3}{12} & \frac{2}{12} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & 1 & 0 & \frac{3}{4} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{pmatrix} \end{array}, \quad \mathbf{L}_c^T\mathbf{L}_r = \begin{array}{c} \\ 1\\2\\3\\5\\6\\10 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 5 & 6 & 10 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{6} & 0 & \frac{5}{6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}.$$

$$\mathbf{H} = \begin{array}{c} \\ 1\\2\\3\\6\\10 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 6 & 10 \end{array} \\ \begin{pmatrix} \frac{5}{12} & 0 & \frac{2}{12} & \frac{3}{12} & \frac{2}{12} \\ 0 & 1 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{4} & 0 & 0 & \frac{3}{4} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{pmatrix} \end{array} \cdot \mathbf{A} = \begin{array}{c} \\ 1\\3\\5\\6 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 3 & 5 & 6 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{5}{6} \end{pmatrix} \end{array}.$$

Eigenvectors:

Av = λv

vTA = λ vT

Numerically: Power Method

The Power Method:

Xk+1 = AXk

Xk+1T = XkTA

Converges to the dominant eigenvector (λ = 1).

In order to converge into a single autovector given some starting point, matrices H and A must be irreducible for the power process. The two H and A are irreducible if our neighborhood graph G is associated. When G is not related, it would be difficult to converge to the particular dominant prospector by using the power method H and A. The explanation demonstrates that the diagram is not linked because page 2 of the hub collection is only related to page 1 and vice versa. H and A are reducible and therefore have other elements that cannot be minimized.

Connected Components:

H contains two connected components, C = {2} and D = {1, 3, 6, 10}

A contains two connected components, E = {1} and F = {3, 5, 6}

Cutting and Pasting. Part I: we can now perform the power method on each component for H and A:

$$\pi_h^T(C) = \begin{array}{c} 2 \\ (1) \end{array}, \quad \pi_h^T(D) = \begin{array}{cccc} 1 & 3 & 6 & 10 \\ (\frac{1}{3} & \frac{1}{6} & \frac{1}{3} & \frac{1}{6}) \end{array}.$$

$$\pi_a^T(E) = \begin{array}{c} 1 \\ (1) \end{array}, \quad \pi_a^T(F) = \begin{array}{ccc} 3 & 5 & 6 \\ (\frac{1}{3} & \frac{1}{6} & \frac{1}{2}) \end{array}.$$

Cutting and Pasting. Part II: we can now paste the two components together for each matrix. We must multiply each entry in the vector by its appropriate weight:

H:

$$\pi_h^T = \begin{array}{ccccc} 1 & 2 & 3 & 6 & 10 \\ (\frac{4}{5}\cdot\frac{1}{3} & \frac{1}{5}\cdot 1 & \frac{4}{5}\cdot\frac{1}{6} & \frac{4}{5}\cdot\frac{1}{3} & \frac{4}{5}\cdot\frac{1}{6}) \end{array}$$
$$= \begin{array}{ccccc} 1 & 2 & 3 & 6 & 10 \\ (.2667 & .2 & .1333 & .2667 & .1333) \end{array}.$$

A:

$$\pi_h^T = \begin{array}{cccc} 1 & 3 & 5 & 6 \\ (\frac{1}{4}\cdot 1 & \frac{3}{4}\cdot\frac{1}{3} & \frac{3}{4}\cdot\frac{1}{6} & \frac{3}{4}\cdot\frac{1}{2}) \end{array}$$
$$= \begin{array}{cccc} 1 & 3 & 5 & 6 \\ (.25 & .25 & .125 & .375) \end{array}.$$

The Closely Knit Group (TKC) results of SALSA have a lower susceptibility than that of HITS. A TKC is an internet topology system made up of a limited number of very interconnected sites. The existence of TKCs in a clustered subgraph impacts the identification by HITS of appropriate authorities [7].

3. Comparative Analysis and Summary of Web Page Ranking Algorithms

Tables 2 and 3 below provide a comprehensive overview and comparison of various web page ranking algorithms, focusing on their advantages, disadvantages, methodologies, and performance metrics. Table 2 summarizes the strengths and limitations of four widely studied algorithms: PageRank, Weighted PageRank, HITS, and SALSA. Each algorithm's advantages and disadvantages highlight their unique contributions and shortcomings in addressing web page ranking challenges.

1. *PageRank* offers fast query time and feasibility but falls short in producing results closely aligned with user requirements.
2. *Weighted PageRank* improves the quality of returned pages compared to the original PageRank but still struggles with relevance to user needs.
3. *HITS* excels in identifying authority and hub pages and supports user-sensitive ranking but suffers from high query time costs and susceptibility to designer-induced errors.
4. *SALSA* effectively mitigates the TKC effect and performs well on general queries but sometimes fails in scenarios requiring mutual reinforcement approaches.

This summary provides a quick reference for understanding how each algorithm performs in different contexts, aiding researchers and developers in selecting the most suitable ranking algorithm for specific applications. Table 3 delves deeper into the technical methodologies, input parameters, quality of results, computational complexity, strengths, and weaknesses of the same algorithms.

1. *PageRank* relies on backlinks and computes scores during indexing, offering medium-quality results but lacking adaptability to natural language queries.
2. *Weighted PageRank* incorporates both forward and backward links, producing higher-quality results than

PageRank but still grapples with theme drift.

3. *HITS* combines web structure and content mining, utilizing hub and authority scores, but faces challenges like topic drift and high computational costs.
4. *SALSA*, which builds on HITS, analyzes sub-web correlations, scoring general queries effectively but exhibiting less reliable relative positioning.

These tables underscore the evolving nature of web page ranking algorithms, emphasizing the trade-offs between efficiency, quality, and adaptability. Together, they serve as a foundational reference for understanding and comparing algorithmic approaches to ranking web content in information retrieval systems.

***Table 2**. Summary of Various Web Page Ranking Algorithms.*

Algorithm	Advantages	Disadvantages
PageRank	1. Query time is less. 2. Feasibility As compared to another algorithms the PageRank algorithm is more feasible.	Relevancy to user's requirements is Less
Weighted PageRank	The Quality of the pages that is returned is more high than PageRank algorithm	This algorithm results also are less in relevancy to user's requirements.
HITS	1. HITS returns in the appropriate authority and hub pages thanks to its ability to identify pages according to the question chain. 2. The rating may also be paired with other rankings dependent on knowledge set. 3. HITS is user-sensitive (in contrast with PageRank).	1. Query Time Cost: Calculation of query time is costly. It is a big downside since HITS is an algorithm based on a question. 2. Important: the ranking or the number of authorities and hubs may increase due to the web page designer's failures.
Stochastic Approach Forlink Structure Analysis (SALSA)	1. SALSA is less vulnerable to the TKC effect, and produces good results in many cases where the mutual reinforcement approach fails to do so. 2. SALSA is particularly effective for very general queries.	Sometimes the effect is beyond mutual reinforcement Approach, and it prevents it from finding relevant trusted sites (or from finding authorities at all

***Table 3**. Comparison of various web page ranking algorithms.*

Algorithm	PageRank	Weighted PageRank	HITS	SALSA
Technique	The Structure of web page	Based on Structure of web page	Web Structure Mining and Web Content Mining	analysis of the correlation structure of sub-web graphics.
Working methodology	scores of pages are computed at indexing time	The web page waiting depends on in links and out links	Compute the Hubs and Authority	SALSA can be seen as an improvement of HITS. (finding hubs and authorities)
Input Parameter	The back links	Forward links and the back links	Contents, Back link and forward links	Contents, Links
Quality of results	Medium	More than Page rank algorithm	Less than page rank	SALSA computationally lighter than the Mutual Reinforcement approach
TIME	O (Log n)	< O (Log n)	<O (Log n) (higher than WPR)	O(N + E)
Strength	Back links (in links) are very considered	The pages are sorted according to the weight of in links and out links	Moderate. Hub & authorities scores are utilized.	SALSA is particularly effective at scoring fairly general queries
Weakness	Results generated at indexing time not query time, also inability to	Theme drift	Topic Drift and Efficiency	Relative position was not so effective

Algorithm	PageRank	Weighted PageRank	HITS	SALSA
	handle results by using natural language without keywords			
Technique	Web Structure Mining	Web Structure Mining	Web Structure Mining and Web Content Mining	analysis of the correlation structure of sub-web graphics.

4. Conclusion

In this paper, we discussed some algorithms that are used in web page ranking, and then we looked at related work. This paper describes a broad evaluation of the performance of ranking algorithms relative to other link-based features. It builds on a previous comparison between HITS, PageRank, SALSA and Weight Page Rank. While our previous study found that HITS and PageRank were below baseline performance A feature based on the correlation between domains in degrees, which casts doubt Benefit on link-based advanced features Web search results ranking.

Author Contributions

Zahir Edrees: Conceptualization, Formal Analysis, Methodology, Resources, Validation, Visualization, Writing – original, Writing -review & editing.

Henda Juma: Resources, Supervision, Validation, Visualization, Writing - review & editing.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Manning, C. D., Raghavan, P., and Schutze, H., ‘Introduction to Information Retrieval’, Introduction to Information Retrieval, (2008).

[2] ME, Mrs Mercy Paul Selvan, A. Chandra Sekar ME, and A. P. D., ‘Ranking Techniques for Social Networking Sites based on Popularity’, (2012).

[3] Ridings, C., & Shishigin, M., ‘Pagerank uncovered. Technical Paper for the Search Engine Optimization Online Community’, (2002).

[4] Grover, N. and Wason, R., ‘Comparative Analysis of Page Rank And HITS Algorithms’, International Journal of Engineering Research & Technology (IJERT), (2012).

[5] Gupta, P., Goel, A., Lin, J., Sharma, A., Wang, D., and Zadeh, R., ‘WTF: The Who to Follow service at Twitter’, (2013).

[6] Patel, P., ‘Research of Page ranking algorithm on Search engine using Damping factor’, International Journal Of Advance Engineering And Research Development, (2014).

[7] Lempel, R. and Moran, S., ‘Stochastic approach for link-structure analysis (SALSA) and the TKC effect’, Computer Networks, (2000).

[8] T. Mandl, “Artificial Intelligence for Information Retrieval,” Encyclopedia of Artificial Intelligence, Jan. 2011, https://doi.org/10.4018/9781599048499.CH023

[9] V. Rijsbergen, “Information Retrieval - Chapter 7,” Inf Retr Boston, pp. 112–140, 1979, Accessed: Nov. 12, 2024. [Online]. Available: http://www.dcs.gla.ac.uk/Keith/Chapter.7/Ch.7.html

[10] Y. Rochat, “Closeness Centrality Extended To Unconnected Graphs : The Harmonic Centrality Index,” ASNA, 2009.

[11] S. Robertson and H. Zaragoza, “The probabilistic relevance framework: BM25 and beyond,” *Foundations and Trends in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009, https://doi.org/10.1561/1500000019

[12] H. Y. Modi and M. Narvekar, “A Comparative Study of Various Page Ranking Algorithms”, [Online]. Available: www.ijert.org

[13] S. Zhang, R. Yang, X. Xiao, X. Yan, and B. Tang, “Effective and Efficient PageRank-based Positioning for Graph Visualization,” Proceedings of the ACM on Management of Data, vol. 1, no. 1, pp. 1–27, Nov. 2023, https://doi.org/10.1145/3588930

[14] D. F. Gleich, “PageRank beyond the web,” SIAM Review, vol. 57, no. 3, 2015, https://doi.org/10.1137/140976649

[15] N. Grover and R. Wason, “Comparative Analysis of Pagerank And HITS Algorithms.” [Online]. Available: www.ijert.org

[16] M. Bianchini, M. Gori, and F. Scarselli, “Inside PageRank,” ACM Transactions on Internet Technology (TOIT), vol. 5, no. 1, pp. 92–128, Feb. 2005, https://doi.org/10.1145/1052934.1052938

[17] Mang’are Fridah Nyamisa, Waweru Mwangi, Wilson Cheruiyot. (2017). A Survey of Information Retrieval Techniques. Advances in Networks, 5(2), 40-46. https://doi.org/10.11648/j.net.20170502.12