Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Enhancing Link Evaluation through a Coordinated Structure: A Comparative Analysis of PageRank and HITS Algorithms

S.Jaiganesh[1], Dr.L.R.Aravind Babu[2*]

[1]Department of Computer and Information Science, Annamalai University, Annamalai Nagar, Tamil Nadu, India, jganesh0@gmail.com
[2]Department of Computer and Information Science, Annamalai University, Annamalai Nagar, Tamil Nadu, India, er.arvee@rediffmail.com

Both the World Wide Web and the quantity of data it contains are growing every day. In this scenario, consumers mostly depend on various search engines to locate pertinent information and appropriate responses to their inquiries. The number and usage of search engines have increased as a result of this widespread tendency. The current situation demands that various link analysis algorithms used by search engines to rank web sites in response to user queries be compared and analyzed. This study presents a comparison of the HITS and PageRank algorithms, two widely used web page ranking methods. The research uses simulations created for both of these algorithms to thoroughly assess their differences, strengths, and drawbacks.

Keywords: Web Mining, Web Page Ranking, User Profiles, Page rank, HITS.

1. Introduction

The World Wide Web (www) is essentially a network of connected hypertext pages [1]. An architectural foundation for accessing linked content dispersed over millions of computers on the Internet is provided by WWW [3]. Getting relevant information out of the massive amount of data on the Internet has proven to be one of the hardest things to do. Web search engines have emerged as a helpful tool that facilitates utilizing user-provided search strings to find relevant content on the Internet. A search engine's results are often shown in a list known as search engine results pages (SERPs). In order to get any information on the internet, the user opens his preferred search engine, types in questions, and clicks on the pages that were returned [7].

A significant quantity of pertinent and irrelevant data is mixed together in the search results that search engines return [5]. Every web page that is returned in response to a user's query cannot be seen by any user. Thus, by presenting the resulting pages in a ranked order based on various page rank algorithms, search engines assist visitors in finding pertinent pages that are worthwhile looking at [5]. Search engines utilize web-page ranking, a method of search engine optimization, to rank thousands of web pages according to their relative significance.

The two primary types of search engines that comprise conventional search engine technology are crawler-based engines and human-powered directories-based engines [6]. An Open Directory example of a human-powered directory relies on people for its catalog [2]. In this configuration, the web pages are categorized and kept in separate folders. A query is first classified when it is fired, and then it searches the relevant directory to get the web page. They are created when a website owner submits their site for assessment along with a brief description [6]. Typically, a search is conducted using only the matches found in the given descriptions. Search engines that rely on crawlers, like Google, generate their listings automatically [2]. "Crawl" or "spider" the internet in an attempt to find pages that correspond with user requests. After they provide result sets, users can via the outcomes. Indexers are used by crawler-based search engines to obtain web page contents [2]. Information about retrieved pages is stored and indexed using indexers. The Retrieval Engine looks up entries in index tables, and the Ranker assesses the relevance of the web pages that are returned. The final element is when the web page ranking algorithms come into play [16]. It's unknown what specific information a user may seek. Therefore, the algorithms used to rank web pages are made to predict what the user will need based on a variety of static (like the quantity of textual content or hyperlinks) and dynamic (like popularity) variables [16]. These are crucial elements that distinguish one search engine from another [16]. Algorithms for online search rankings are crucial in determining the order of web pages so that users may get the best a response that is more appropriate for the user's inquiry [8]. The operation of a typical search engine is demonstrated in Figure 1 [4], which displays the flow graph for a question that a web user has searched.

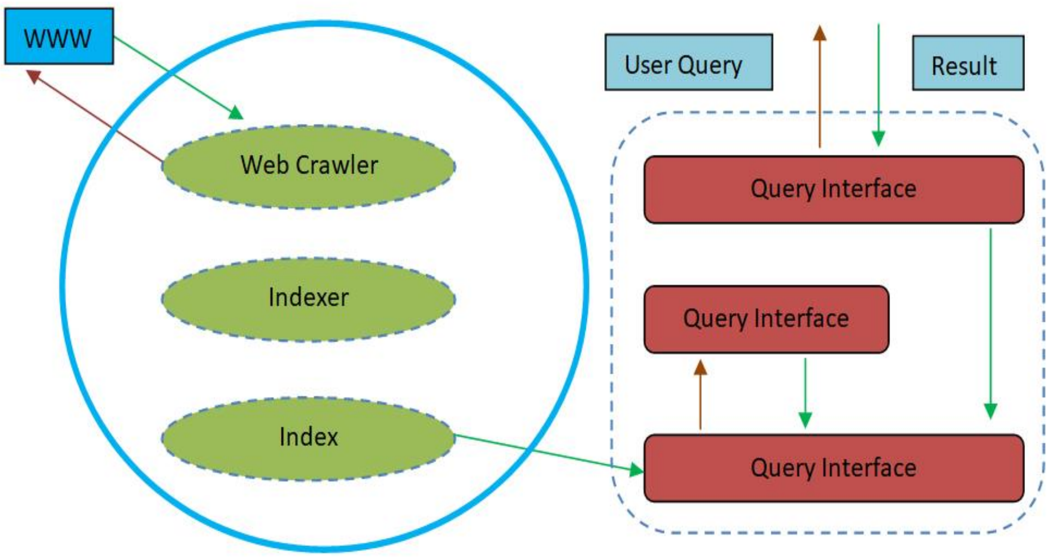


Fig. 1. Show the way a Search Engine Performs

This work aims to investigate the widely used web PageRank and HITS are two web page ranking algorithms, and their modifications. It also provides a comparative analysis of both algorithms, highlighting their respective advantages and disadvantages.

2. Literature Review

By itself, hyperlink analysis is a subset of the larger field of study known as web mining, defined as the use of data mining techniques to extract valuable information from Web data [22]. Internet users can benefit from web mining by learning about future web sites to see [4]. The types of information that can be gathered are utilized in Web Mining analysis, such as usage, structure, and content data [23]. Based on the types of data that need to be mined, Web mining may be roughly classified into three groups [23]:

2.1 Web Content Mining (WCM):

WCM is responsible for exploring the proper and relevant information from the contents of web pages [4]. Content data corresponds to the collection of facts a Web page was designed to convey to the users [22]. It may consist of text, images, audio, video, or structured records such as lists and tables [22].

2.2 Web Structure Mining (WSM):

Web pages are the nodes in a typical Web graph, and hyperlinks are the edges that connect two related pages [22]. WSM processes the web structure to determine the relationship between various web pages [4]. The application of Web Structure Mining is beneficial for taking structural data from the Internet.

2.2.1 WSM is available at two different levels:

i. Document structure analysis: this field examines a document's structure, including the Document Object Model.

ii. Analysis of link types: addresses linkages that might be intra- or inter-document. In the field of web mining, the quantity of outlines—links coming from a page—and the quantity of links—links going to a page—are crucial variables [4]. The inter-document link type structure serves as the foundation for hyperlink analysis, as seen in figure 2 [22]. When combined with Web content, hyperlinks' structural information may be utilized to extract valuable data from the Web and assess the reliability of the data [22]. WSM therefore becomes one of the most crucial areas for web mining study [4].

2.2.2 Web Usage Mining (WUM):

To better understand and meet the demands of Web-based applications, Web Usage Mining (WUM) is the application of data mining techniques to uncover intriguing usage patterns from Web data [23]. WUM is in charge of logging user activity and profile information into the web log file [4]. IP addresses, page references, and user access times are among the common use data that websites gather [22]. Figure 2[22] below shows the high level taxonomy of several research projects in web mining and web structure mining:

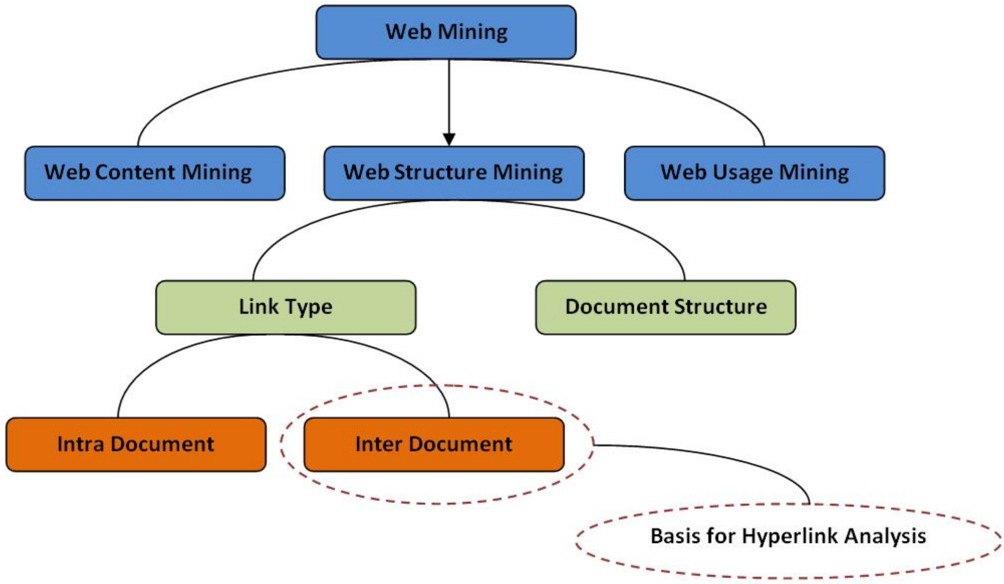


Fig. 2. High level taxonomy of Web Mining

3. Ranking Algorithms

The search results are ranked by web page ranking algorithms based on how relevant they are to the query. Because of this, the search results are ranked by relevancy to the query string being searched, going down in sequence. The relevance of a web page to the terms and concepts in the query, the popularity of its links generally, and other criteria all affect how well it ranks for a given query. These algorithms fall into two categories: text-based and link-based [6].

3.1 Text-Based Ranking

It might appear reasonable that the traditional search engines' text-based ranking system ranks pages according to their textual content. In these methods, a page's rank is determined by the following elements [6]:
The number of terms that the query string matched. Location factors affect a page's rank based on where the search phrase appears on the page. A page's title, the paragraphs that precede it, or even the area closest to the top of the page may include the search query string [6]. The frequency variables determine how many times the search string appears on the page. The page ranking improves with the length of time the string appears [6]. The majority of the time, these components' combined effects are taken into account. For instance, a website should have a high rank if a search phrase occurs close to the top of the page often [6].

3.2 Link-Based Ranking Algorithms

The link-based algorithms are another well-liked family of ranking algorithms. They see the internet as a directed graph, in which the hyperlinks between web sites represent the directed edges connecting these nodes, while the web pages themselves serve as the nodes [6,27]. Through links, link-based ranking algorithms distribute the relevance of a page. Two of the most significant hyperlink-based search algorithms were published between 1997 and 1998. These formulas are as follows:

- PageRanking algorithm
- HITS (Hyperlink Induced Topic Search)

Social networks are connected to both algorithms. They use the Web's linking structure to their advantage, ranking pages based on their degrees of "prestige" or "authority." Section 4 and 5 next individually discuss the above algorithms.

4. HITS Algorithm

4.1. Overview

Jon Kleinberg created the link analysis system known as Hypertext Induced Topic Search (HITS), also known as hubs and authority, in 1998 to rank Web sites. HITS is a search query dependent algorithm that scores a webpage by analyzing all of its inlinks and outlinks. It was developed before PageRank [4]. Consequently, the website's rating determined by comparing the text's contents to a specified query. Following the user's search query, HITS generates two rankings an authority ranking and a hub ranking of the extended collection of relevant pages that the search engine returns. According to this algorithm, a website is classified as an authority if it has a lot of hyperlinks pointing to it, and a website is classified as a hub if it points to a variety of hyperlinks [4]. Two kinds of pages are produced by the algorithm:

- Authority: Pages that offer significant, reliable information on a certain subject
- Hub: Pages with connections to authoritative sources

HITS produced the hubs and authorities shown in Figure 3 [8] below. There is a reciprocal link between authorities and hubs: many good hubs point to a better authority, and many good hubs point to a better authority.

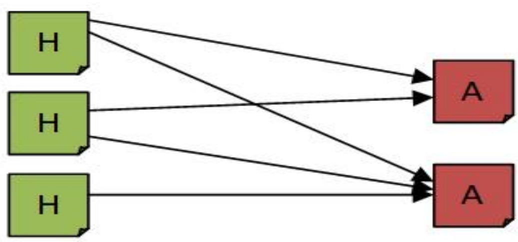


Fig. 3. Hubs and Authorities

To mark a web page as Authority or Update, HITS follows the following rules [27]:

Authority Update Rule: **∀p**, update auth (**p**) as follows:
$\sum_{i=0}^{n}$ Hub i n i=1 (1)

Where n is the total number of pages connected to p. According to (1) the Authority score of a page is the sum of all the Hub scores of pages that point to it [8]. Hub Update Rule: ∀p, we update hub (p) as follows:

$\sum_{i=0}^{n} \text{Auth}\ (i)$ (2)

where p connects to and n is the total number of pages. As per (2), the total Authority ratings of all the connecting pages on a page add up to its Hub score [8]. More specifically, the HITS method first creates the n by n adjacency matrix A, whose m(i, j) element is 1 if page i connects to page j and 0 otherwise, given a collection of web pages (let's say, obtained in response to a search query).

Adjacency Matrix A

$m_{(i,j)} = 1$ if (i,j) exists in graph,

$m_{(i,j)} = 0$ otherwise.

It then iterates the following equations [9]: For each m_i,

$a_i^{(t+1)} = \sum\{j: j \rightarrow i\}\ h_j^{(t)}$ (3)

$h_i^{(t+1)} = \sum\{j: i \rightarrow j\}\ a_j^{(t+1)}$ (4)

(Where "i j" means page i links to page j and ai is authority of ith page and hi is the hub representation of ith page).

Figure 4[4] shows an illustration of HITS process.

Normalization:

After the method is run an unlimited number of times, the final hub-authority scores of the nodes are found. Diverging values are achieved by using the authority update rule and the hub update rule sequentially and directly. Thus, after every iteration, the matrix must be normalized [11].

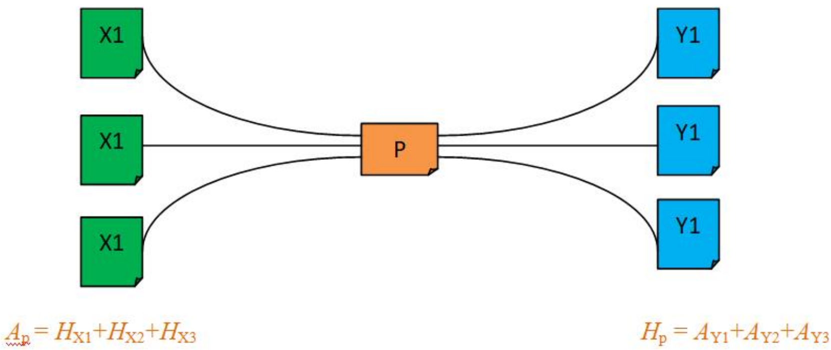


Fig. 4. Illustration of HITS process

4.2 Implementation of HITS algorithm

The first n pages that a text-based search algorithm returns can be used to acquire the root set, or the most relevant pages to the query, in the first phase of the HITS method. By adding all the webpages that link to and from the root set, as well as some of the pages that connect to it, one may create a base set. A focused subgraph is created by the base set's webpages as well as all of the hyperlinks between them. This concentrated subgraph is the sole one on which the HITS computation is done [24]. The goal of creating a base set, according to Kleinberg [25], is to guarantee that the majority (or even most) of the most authoritative sources are included. A node's Hub and Authority scores are determined using the following algorithm [11]:

- Each node should begin with a hub score of 1 and an authority score of 1.
- Use the Authority Update Rule.
- Utilize the Hub Update Rule.
- Divide each Hub score by the sum of the squares of all Hub scores and each Authority score by the sum of the squares of all Authority scores to normalize the numbers.
- As needed, carry out step two again.

Algorithm pseudocode for the HITS [26,27]

```
 Let G be set of pages
 for each page PG in G do
 PG.Auth = 1        // the page's authority score (PG)
 PG.Hub = 1         // hub score of the page PG
function Calc_Hubs_Authorities(G)
for step from 1 to i do       // run the algorithm for i steps
 norm = 0
for each page qg in G do    // update authority values
 PG.Auth = 0
for each page PG in p.inNeighbors do        //set of pages that link to pg
PG.Auth += QG.Hub
norm += square(PG.Auth)   //sum of the squared auth values to normalise
norm = sqrt(normal)
for each page PG in G do // update the auth scores
PG.Auth = PG.Auth / normal        // normalise the auth values
norm = 0
for each page pg in G do    // update hub values
PG.hub = 0
for each page rg in pg.outNeighbors do      // set of pages that pg links to
PG.Hub += RG.Auth
norm += square(PG.Hub)    //sum of the squared hub values to normalise
norm = sqrt(normal)
for each page PG in G do   //update hub values
PG.hub = PG.Hub / normal           // normalise the hub values
```

The pseudocode above shows how the hub and authority values converge. A workaround for this, though, would be to divide each hub value is calculated using the square root of the total squares of all hub values, and each authority value is calculated using the square root of the total squares of all authority values. In order to normalize the hub and authority values after each "step". The above pseudocode does this.

4.3 Simulation

4.3.1 Graph case study 1

Three pages, A, B, and C, are connected by directed edges in a tiny network seen in Figure 4. The HITS algorithm's implementation simulation on the graph is shown in the figure 5:

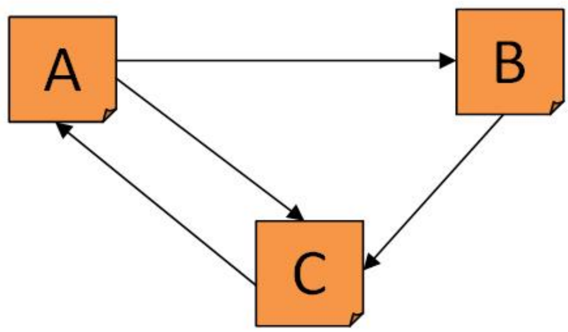


Fig. 5. Connected graph

The hubs and authority scores for each node, as determined by simulation, are shown in Figure 6.

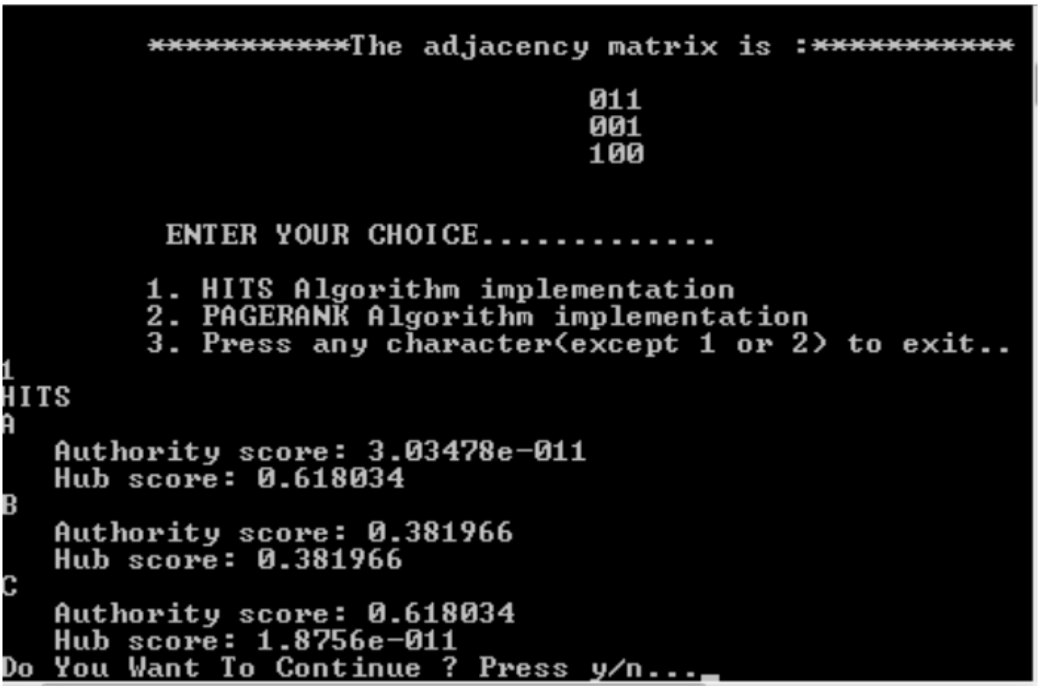


Fig. 6. Simulation for graph1

4.3.2 Graph case study 2

Four pages in a tiny network, A, B, C, and D, are represented by the graph in Figure 7 and are connected by directed edges. Figure 8 depicts the simulation running the HITS algorithm on the graph:

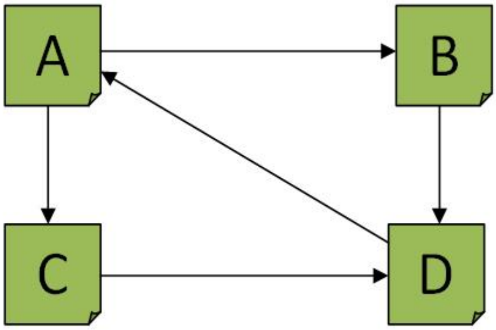


Fig. 7. Connected Graph

The hubs and authority scores for each node, as calculated by simulation, are shown in Figure 8.

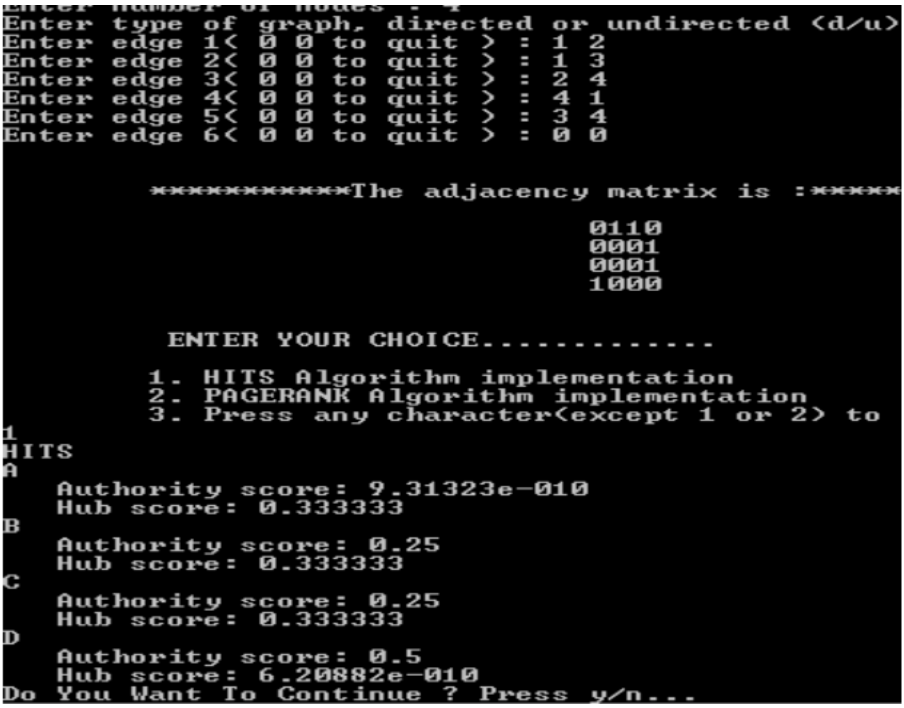


Fig. 8. Simulation for graph2

4.3.3 Case Study 3 on Graphs

A more intricate network with seven pages is seen in the graph in figure 9.

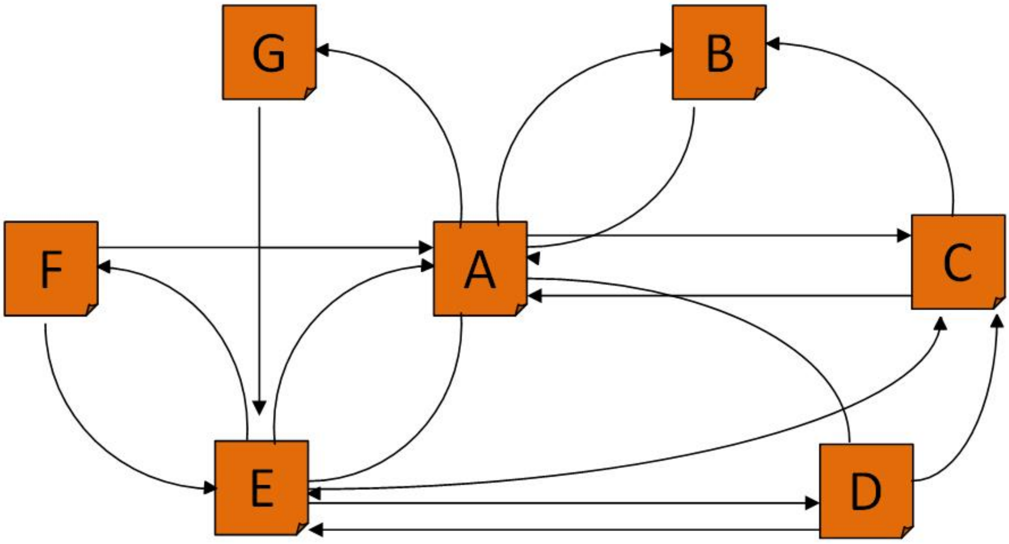


Fig. 9. Connected graph

The hubs and authority scores for each node, as determined by simulation, are shown in Figure 10 below.

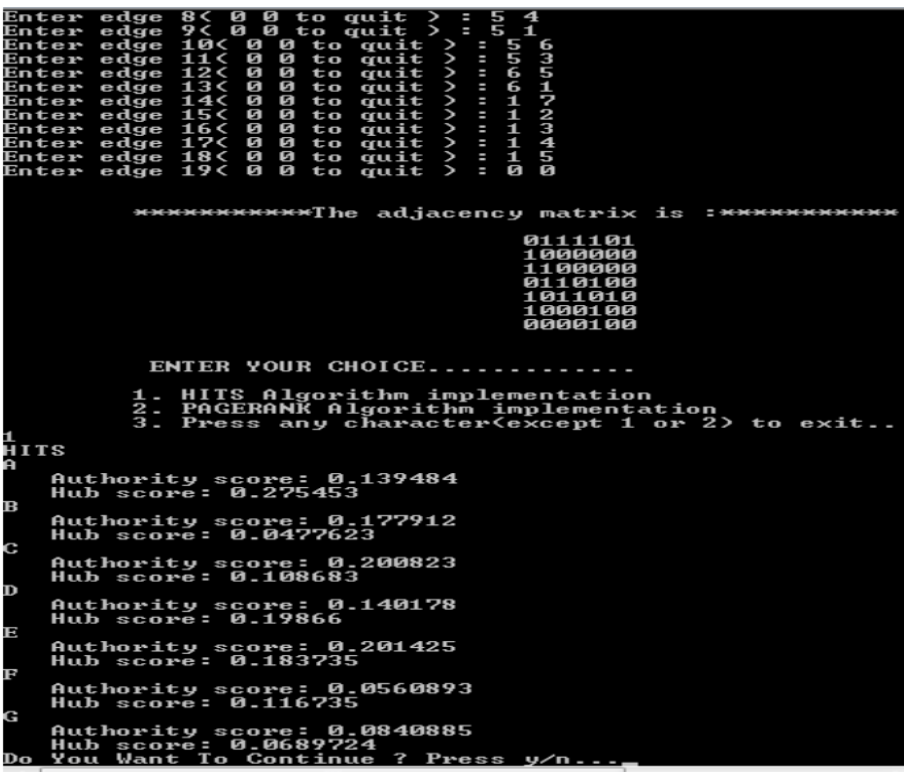


Fig.10. Simulation for graph3

4.4 Advantages of HITS

4.4.1. HITS receives points for its capacity to rank pages based on the query string, producing hub pages and pertinent authority.

4.4.2. The rating can also be paired with other rankings based on information retrieval.

4.4.3. In contrast to PageRank, HITS is responsive to user queries.

4.4.4. Important pages are selected based on hub value and authority calculations.
4.4.5. The HITS algorithm is a generic method for ranking the retrieved material by determining hubs and authority.

4.4.6. Using a specified query string, HITS generates a Web graph by locating a collection of pages.
4.4.7. The results show that HITS accurately determines hubness and authority nodes.

4.5 Drawbacks of HITS algorithm

Cost of Query Time: The assessment of query time is costly. Given that HITS is a query-dependent algorithm, this is a significant disadvantage.

Irrelevant authorities: The web page designer's errors may cause the rating or scores of authorities and hubs to increase. HITS considers that when a user develops a webpage, he does so because he really thinks the authority page is connected to his page (hub) and establishes a hyperlink from his website to another authority page.

Irrelevant Hubs: A page that has links to a lot of different topics may have a high hub rank even if it has nothing to do with the query at hand. If this website links to highly rated authority, it maintains a very high hub rank even when it is not the most important source for any information.

Interactions between hosts that are mutually reinforcing: HITS places a strong emphasis on interactions between hub websites and authorities. A good authority is a page that is linked to by several other good authorities, while a good hub is a page that links to numerous other excellent authorities.

Topic Drift: When there are tightly related but irrelevant sites in the root set, topic drift happens. The pages in the base set will be affected since the root set itself contains irrelevant pages. Additionally, the web graph created from the base set's pages won't contain the most pertinent nodes, making it impossible for the algorithm to identify the hubs and authority that score highest for a specific query.

Less Feasibility: HITS looks for two categories of pages: hubs, which are pages that point to numerous high-quality pages, and authorities, which are high-quality pages. It does this by invoking a typical search engine to gather a collection of relevant pages, then growing this set with its inlinks and outlinks [20]. Today's search engines, which must process tens of millions of requests every day, cannot manage this computation since it is done at query time [20].

5. PageRank Algorithm

5.1 Overview

As part of a study on a new search engine, Larry Page and Sergey Brin created the PageRank algorithm at Stanford University in 1996. Google's search engine uses a link analysis technique called PageRank, which bears Larry Page's name. In order to determine each page's relative value within a hyperlinked set of documents, the algorithm gives each page a numerical weight or rank. The web page link structure is used by the Page Rank algorithm. This method runs on the whole Web, is query agnostic, and gives each web page a PageRank [13]. It is predicated on the ideas that if a page has significant connections pointing to it, then the links of this sites that point toward one another should also be regarded as significant pages; for example, if page A, an authoritative website, has a link to page B, then page B is likewise authoritative. The back link is used by Page Rank to determine the rank score. The page is given a substantial rank if the total of all the backlink rankings is high [4]. In [4], a condensed form of PageRank is also proposed:

$$PR(u) = \sum_{v \in B(u)} PV(v)/L(\mathrm{v})$$

The PageRank value of a web page u in equation (5) is derived by dividing the PageRank values of all the web pages v in the set Bu (which includes all pages that link to web page u) by the total number of links from page v, or L(v). The diagram in figure 11 provides an example of back connections between a group of sites. This is A's backlink, B. D is the back link for E, while A, B, and E are the back links for C.

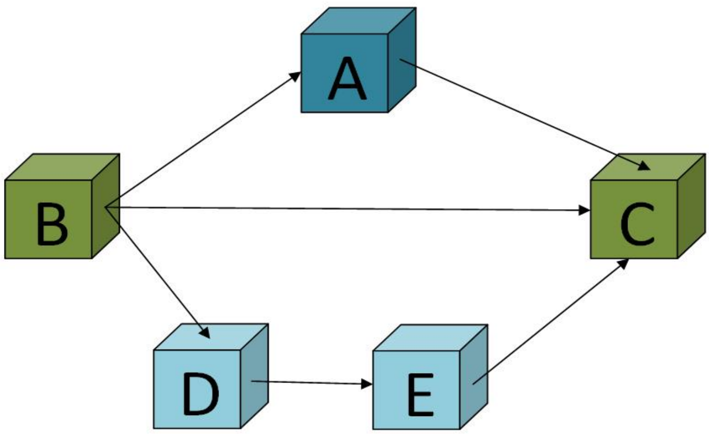


Fig. 11. An Illustration of Back Links

Three parameters were first used by the Google search engine to determine the ranking of online sites [18].

- Page specific factors
- Anchor text of inbound links
- PageRank

Page-specific elements can include the document's URL, body content, and HTML tag weight components (such title preference). Numerous other elements have been incorporated into Google's ranking algorithms. Google uses the anchor text of inbound links on a page and page-specific criteria to calculate an IR score that is weighted by the location and accentuation of the search word within the document [18] in order to deliver search results. The page's overall relevance is then determined by multiplying the IR-score by PageRank [18].

5.2 Implementation of Pagerank Algorithm

The PageRank algorithm ranks each page separately rather than assigning a ranking to the entire website. Moreover, the PageRank of the pages that link to page A defines page A's PageRank recursively. A probability distribution called PageRank is used to show how likely it is for a user to randomly click on links and get on a specific website. A document with a PageRank of 0.5 indicates that there is a fifty percent of it probability that someone clicking on a random link will be taken to it. The PageRank formula was described as follows by Brin S. and L. Page [10]:

PR(A)=(1-d)+d(PR(T1)/C(T1)+………+ PR(Tn)/C(Tn))
Where:
PR(A)= PageRank of page A
T1….Tn=All pages that link to page A
PR(Ti)=Page rank of page Ti
C(Ti)=the number of pages to which Ti links to
d= damping factor which can be set between 0 and 1
PR(Ti)/C(Ti)= PageRank of Ti distributing to all pages that Ti links to.
(1-d)= To make up for some pages that do not have any out-links to avoid losing some page ranks.
Each additional inbound link for a web page always increases that page's PageRank [18]. One may assume that an additional inbound link from page X increases the PageRank of page A by [18]:
d × PR(X) / C(X)

Damping factor: According to the PageRank hypothesis, any hypothetical surfer who clicks on links at random would ultimately give up. A damping factor d is the likelihood that the individual will continue at any given stage. The damping factor is normally fixed at 0.85, however it may be adjusted to any number such that 0.

Pseudocode for PageRank (G) [6,27]

Input: Let G represent set of nodes or web pages

Output: An n-element array of PR which represent PageRank for each web page

1. For i $\leftarrow$ 0 to n-1 do
2. Let A be an array of n elements
3. A[i] $\leftarrow$ 1/n
4. d $\leftarrow$ some value $0<d<1$, e.g. 0.15, 0.85
5. Repeat
6. For i $\leftarrow$ 0 to n-1 do
7. Let PR be a n-element of array
8. PR[i] $\leftarrow$ 1-d

```
For all pages Q such that Q links to PR[i] do
Let O_n be the number of outgoing edge of Q
PR[i] PR[i]+ d * A[Q]/O_n
If the difference between J and PR is small do
Return PR
For i ← 0 to n-1 do
A[i] ← PR[i]
```

5.2 Simulation

5.3.1 Graph case study 1

Three pages are linked together in a network, as seen in the figure 12[14] graph. Every page displays its estimated PageRank inside. This figure shows how PageRank is applied to a condensed three-page internet [14].

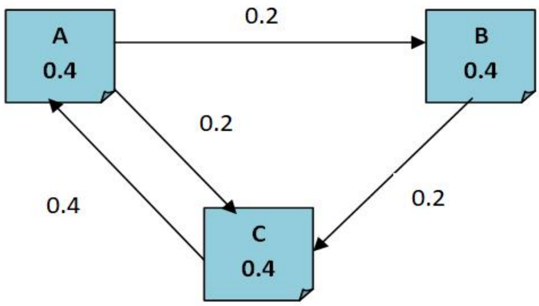


Fig. 12. PageRank in an internet with just three pages.

Figure 13 below displays the PageRank determined by simulation for every page.

```
Enter type of graph, directed or undirected (d/u) : d
Enter edge 1( 0 0 to quit ) : 1 2
Enter edge 2( 0 0 to quit ) : 1 3
Enter edge 3( 0 0 to quit ) : 2 3
Enter edge 4( 0 0 to quit ) : 3 1
Enter edge 5( 0 0 to quit ) : 0 0


        ***********The adjacency matrix is :***********

                                011
                                001
                                100


         ENTER YOUR CHOICE.............

        1. HITS Algorithm implementation
        2. PAGERANK Algorithm implementation
        3. Press any character(except 1 or 2) to exit..
2
PAGERANK
Pagerank
A:      0.390517
B:      0.211397
C:      0.398086

The sum of PageRank is 1
Do You Want To Continue ? Press y/n...
```

Fig. 13. For graph1, a simulation

5.3.2 Graph case study 2

Figure 14 shows a four-node representation of a network that illustrates the PageRank.

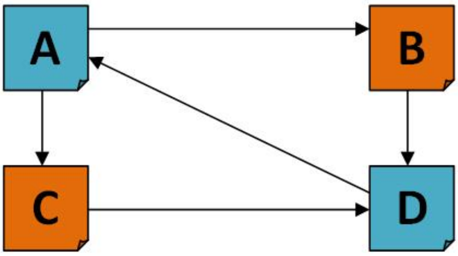


Fig. 14. Graph 4 nodes

The results of the simulation for Graph 2 are displayed below Figure 15.

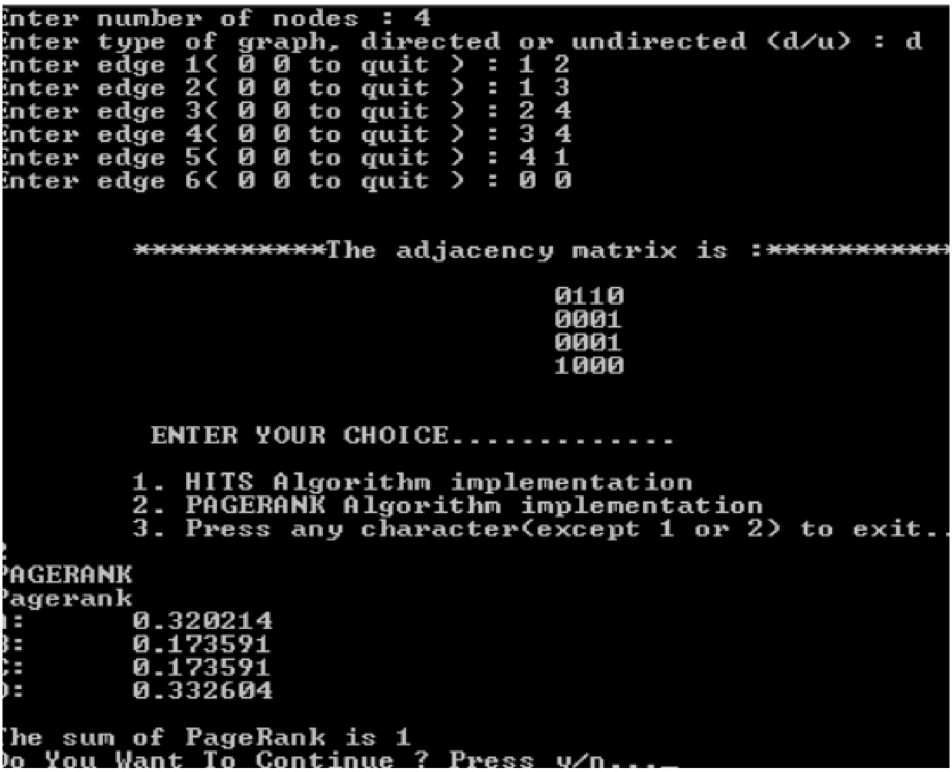


Fig. 15. Imulation for Graph2

5.3.3 Graph case study 3

A more complicated model that better captures the functionality of PageRank is depicted in the graph in figure 16 [14].

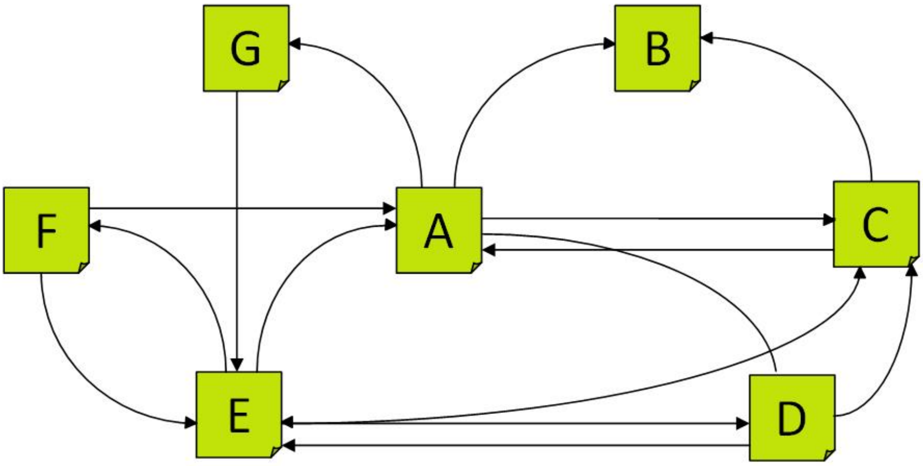
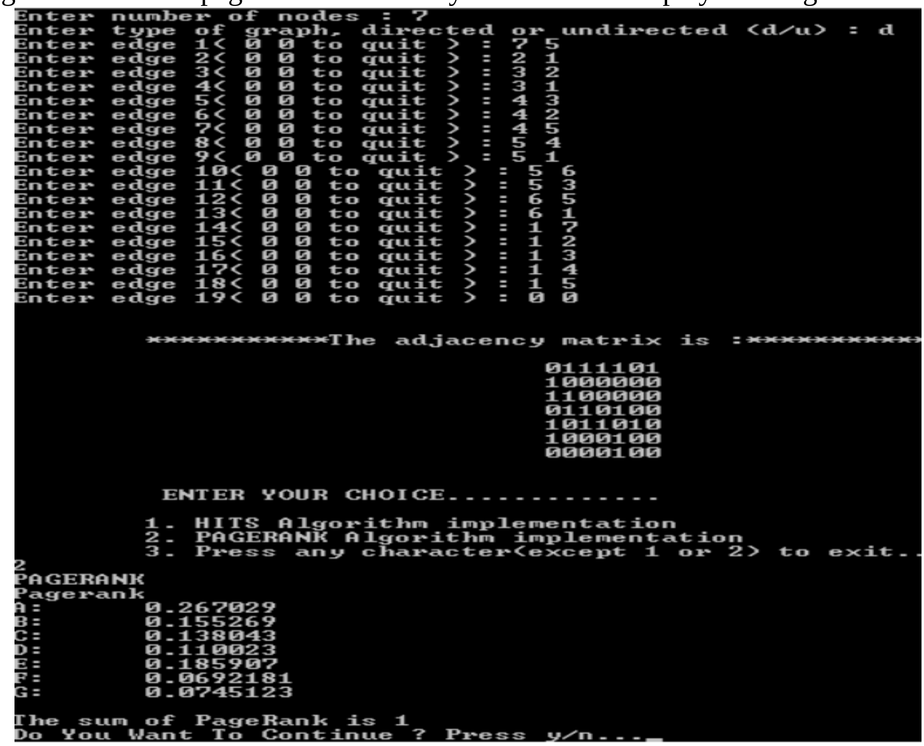


Fig. 16. Graph of a complex network internet.

The PageRank for each page as determined by simulation is displayed in Figure 17 below.



```
Enter number of nodes : 7
Enter type of graph, directed or undirected (d/u) : d
Enter edge 1( 0 0 to quit ) : 7 5
Enter edge 2( 0 0 to quit ) : 2 1
Enter edge 3( 0 0 to quit ) : 3 2
Enter edge 4( 0 0 to quit ) : 3 1
Enter edge 5( 0 0 to quit ) : 4 3
Enter edge 6( 0 0 to quit ) : 4 2
Enter edge 7( 0 0 to quit ) : 4 5
Enter edge 8( 0 0 to quit ) : 5 4
Enter edge 9( 0 0 to quit ) : 5 1
Enter edge 10( 0 0 to quit ) : 5 6
Enter edge 11( 0 0 to quit ) : 5 3
Enter edge 12( 0 0 to quit ) : 6 5
Enter edge 13( 0 0 to quit ) : 6 1
Enter edge 14( 0 0 to quit ) : 1 7
Enter edge 15( 0 0 to quit ) : 1 2
Enter edge 16( 0 0 to quit ) : 1 3
Enter edge 17( 0 0 to quit ) : 1 4
Enter edge 18( 0 0 to quit ) : 1 5
Enter edge 19( 0 0 to quit ) : 0 0

        ***********The adjacency matrix is :**********

                              0111101
                              1000000
                              1100000
                              0110100
                              1011010
                              1000100
                              0000100

         ENTER YOUR CHOICE.............

        1. HITS Algorithm implementation
        2. PAGERANK Algorithm implementation
        3. Press any character(except 1 or 2) to exit..
2
PAGERANK
Pagerank
A:      0.267029
B:      0.155269
C:      0.138043
D:      0.110023
E:      0.185907
F:      0.0692181
G:      0.0745123

The sum of PageRank is 1
Do You Want To Continue ? Press y/n...
```

Fig. 17. PageRank calculated using simulation for graph3.

5.4 Complexity Analysis

The number of iterations (i), the number of web pages (n), and the number of outgoing edges of each web page (On) are the three elements that determine how long the PageRank algorithm takes to execute, according to the pseudocode shown in section 5.2 [6]. The degree of complexity is around

$n + in \sum_{i=1}^{n} oi + n = n\,(2 + i\ \sum_{i=1}^{n} oi\ )$

Given that there are comparatively fewer outgoing edges and iterations per page than there are web pages. O(n) is the PageRank's complexity [6].

5.5 Advantages of the PageRank algorithm

The following are some of the PageRank algorithm's advantages:

5.5.1. Reduced query time cost: PageRank outperforms the HITS method because it requires less query time to include the precomputed PageRank significance score for a page [19].

5.5.2. Less vulnerability to localized links: In addition, PageRank is less vulnerable to localized link spam since it is built utilizing the complete Web graph rather than just a portion of it [19].

5.5.3. More Effective: PageRank, on the other hand, calculates a single quality metric for a page throughout crawl time. At query time, this metric is then blended with a conventional information retrieval score. This has the benefit of being far more efficient than HITS [20].

5.5.4. Feasibility: Because the PageRank algorithm does calculations at crawl time rather than query time, it is more practical in the current environment than the Hits method.

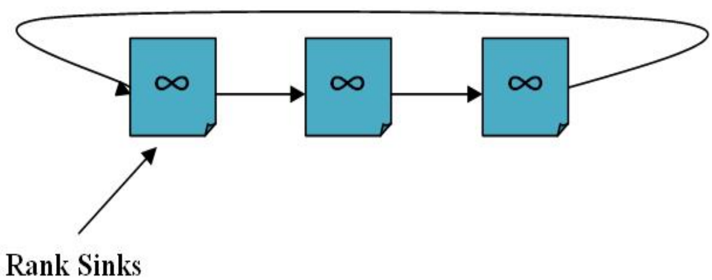


Fig. 18. Illustration of Rank Sinks

5.6 Drawbacks of PageRank algorithm

The following are the problems or disadvantages [17] of PageRank:

5.6.1 Rank Sinks: As seen in figure 18 [17] below, the Rank Sinks problem arises when sites in a network are stuck in endless link cycles:

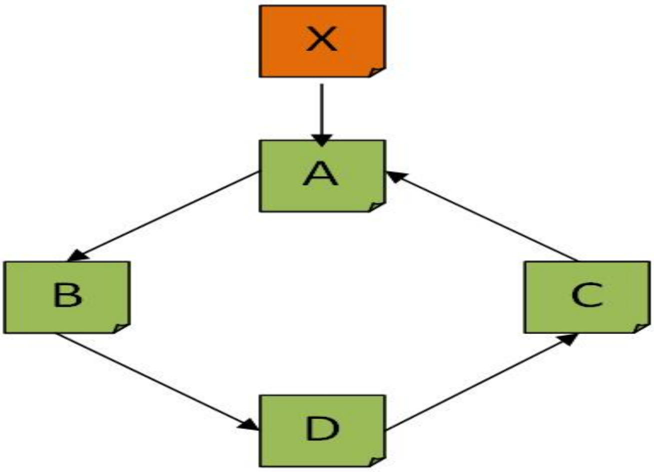


Fig. 19. Illustration of Circular References

5.6.2. Spider Traps: Spider Traps are an additional issue with PageRank. If there are no linkages from one page to another inside the group, then the collection of pages is a spider trap.

5.6.3. Dangling Links: This happens when a link on one website leads to another page without any outbound links. Dangling Links are such kinds of links.

5.6.4. Dead Ends: A dead end is just a page that has no links that lead anywhere.

5.6.5. Pages without outedges are difficult for PageRank to handle as they lower PageRank overall.

5.6.6. Circular References: The PageRank of your first page will be lowered if your website contains circle references [18]. Circular references are seen in the picture 19 [18] below.

5.6.7. Effect of more pages: Adding a page to your website will raise its ranking by around 0.428 [18]. The issue with this strategy is that adding more pages to your home page will result in a decrease in the PageRank of your other pages [18]. Exchange links with websites that have a high PageRank value is the solution. Creating a page with a high PageRank and linking it to your home page is the simplest approach to do this [18].

5.6.8. A page's PageRank score does not take into account if the page is pertinent to the current query or not.

6. A comparison between PageRank and HITS.[27]

Table 1 below enlists the comparison of HITS and PageRank algorithm.

Table 1. Comparison of HITS and PageRank algorithms

Criteria	HITS	Page Rank
Basic Criteria	Link analysis algorithm	Random surfer model-based link analysis method.
Principal Method used	Web Structure Mining, Web	Web Structure Mining

	Content Mining	
Efficiency	HITS looks for hubs and authority after retrieving a list of relevant sites for a given query from a typical search engine. Because of the calculation involved at query time, modern search engines are unable to handle the millions of inquiries they get each day.	PageRank computes a single quality metric for a page at crawl time. This measure is then combined with a traditional information retrieval score at query time. Significantly greater efficiency is the gain.
Mutual Reinforcement	HITS places a strong emphasis on the reciprocal reinforcement of authority and hub sites.	PageRank doesn't try to figure out how to distinguish between authority and hubs. Pages are ranked only by
Neighborhood	HITS is used to the immediate vicinity of the pages that surround a query's results.	PageRank is used across the whole internet.
Dependency of Queries	HITS is query dependent	PageRank is query independent
Stability	Can be erratic: a few link changes may cause drastically different ranks.	can be erratic: a few link changes can result in noticeably different ranks.
Input Parameter(s)	Content, Back and Forward links	Back links
Analysis Scope	Single Page	Single Page
Relevancy	Less. Given that the pages are ranked by this algorithm on the indexing time	More because this algorithm takes into account the page's content in addition to using hyperlinks to provide quality results
Quality of Results obtained	Less than PageRank algorithm	Medium
Merits	• The calculation of Hub and Authority values yields the pertinent and significant pages. • To rank the obtained data, the authority and hubs are determined using the generic technique known as HITS. • That algorithm's main goal is to generate the Web graph by locating a collection of sites that have a search on a specific subject (query). • The hubness and authority node calculations show that it performs well, according to the results.	• Applied in academic writing and journal citations • Google's algorithm for page ranking. • The precomputed PageRank significance score for a page has a low query-time cost. • Because PageRank is derived from the complete Web graph, not just a slice of it, it is less vulnerable to localized link spam. • PageRank may be applied as a way to assess how much an online community such as the blogosphere affects the Web as a whole.
Limits	• Dependency of Queries • The issue of irrelevant authorities Unrelated Hubs issue • Problems with hosts'	• Rank Sinks • Spider Traps • Dangling Links • Dead Ends • Circular References

	mutually reinforcing ties • Subject Drift	• Effect of additional pages

7. Conclusion

We infer from this study that the HITS algorithm and page rank are distinct link analysis methods that use separate models to determine web page rank. The more well-known algorithm Page Rank serves as the foundation for the widely utilized Google search engine. The absence of certain qualities in the HITS algorithm, such as efficiency, feasibility, reduced query time cost, and reduced susceptibility to localized linkages, are the reason for its popularity. Even though the HITS algorithm hasn't been all that popular, several variations of it has been used on a variety of websites.

References

1. Charanjit Singh,Vijay Laxmi,Arvinder Singh Kang,A New Ranking Algorithm for Search Engine: Content's Weight based Page Ranking International Journal of Computer Applications (0975 – 8887) Volume 152 – No.7, October 2016
2. Vinamrata Singh1*,Kailash Patidar2 and Rajendra Prasad Sahu3 A survey and analysis of page ranking through data mining and advanced techniques International Journal of Advanced Technology and Engineering Exploration, Vol 5(39) ISSN (Print): 2394-5443 ISSN (Online): 2394-7454 ©2018
3. Neha Agrawal, A Web Service Discovery Approach Based on Operation Discovery with Ranking Algorithm International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; Volume 11 Issue II Feb 2023
4. Shubham Goela , Ravinder Kumara , Munish Kumarb,⁎ , Vikram Chopra. An efficient page ranking approach based on vector norms using sNorm(p) algorithm 0306-4573/ Information Processing and Management 56 (2019) 1053–106 © 2019.
5. E.Preethi1,Dr. R.Arumugam2 Applications Of Stochastic Models In Web Pageranking, International Journal of Recent Trends in Engineering & Research (IJRTER) Volume 03, Issue 03; March - 2017 [ISSN: 2455-1457] @ IJRTER-2017
6. Atul Kumar Srivastava1 , Mitali Srivastava2 , Rakhi Garg3 , P. K. Mishra4, Mahila Maha Vidyalaya, Comparative Study of Web Page Ranking Algorithms, International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS) ISSN (Print): 2279-0047 ISSN (Online): 2279-0055 IJETCAS 13- 106; © 2014,
7. Prof.Er.Dr.G.Manoj Someswar, Design and Development of an Efficient Data Mining Algorithm for Ranking Problems, Research Inventy: International Journal of Engineering And Science Vol.9, Issue 1 (Jan 2019), PP -01-18 Issn (e): 2278-4721, Issn (p):2319-6483, ; © 2019
8. Devendra Tanaji Rane and Ganesh R. Pathak, Focused Web Crawler With Genetic Algorithm-An Approach To Web Mining, International Journal of Applied Engineering and Technology ISSN: 2277-212X (Online) An Open Access, © 2017
9. Prabhat Kumar Sahu, Dr. Rajendra Gupta, Frequent Sequential Traversal Pattern Mining for Next Web Page Prediction Int. J. Advanced Networking and Applications Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290, , © 2021
10. Neetu Narwal*,Sanjay Kumar Sharma, Amit Prakash Singh, Fuzzy rule-base optimisation using genetic algorithm for mobile web page adaptation, Int. J. Information and Decision Sciences, Vol. 10, No. 4, © 2018

11. Pallavi *, Dushyant Singh, Hybrid Algorithm For Page Ranking In Information Retrieval Systems, INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, ISSN: 2277-9655, IC™ Value: 3.00 © 2016
12. 1 Shubham P. Sharma, 2Krishna M. Gupta, 3Akshay D. Pabale, 4 Prof. Deepti Vijay Chandran, Implementation of Semantic and Synaptic Web Mining- An Integrated Web Mining Algorithm, International Journal for Research in Engineering Application & Management (IJREAM) ISSN : 2454-9150 Vol-03, Issue 02, © Apr 2017.
13. Mohamed Nadjib MEADI1,∗ , Mohamed Chaouki BABAHENINI1 , Abdelmalik TALEB AHMED2, New use of the HITS algorithm for fast web page classification, Turkish Journal of Electrical Engineering & Computer Sciences, Turk J Elec Eng & Comp Sci (2017) 25: 2015 – 2032 TUB¨ ˙ITAK doi:10.3906/elk-1501-236 ©2017
14. Dr.S. Haseena 2Dr.M. Charles Arockiaraj 3 Dr.Radha. C, OPTIMIZATION OF WEB PAGE RANKING OF PERSONALIZED SEARCH USING IMPROVED BM25F MODEL, NeuroQuantology ,Volume 20 | Issue 10 | Page 924-935| doi: 10.14704/nq.2022.20.10.NQ55070 © 2022.
15. Albi Dode1 , Silvester Hasani2, PageRank Algorithm, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 1, Ver. III ,PP 01-07© 2017
16. Debajyoti Mukhopadhyay , Pradipta Biswas , Young-Chon Kim “A Syntactic Classification based Web Page Ranking Algorithm”, 6th International Workshop on MSPT Proceedings
17. “Modeling and Optimizing Hypertextual Search Engines” Based on the Reasearch of Larry Page and Sergey Brin, Yunfei Zhao Department of Computer Science, University of Vermont Slides from Spring 2009 Presenter: Michael Karpeles
18. Google and the Page Rank Algorithm, slides by Székely Endre 2007. 01. 18.
19. Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search Taher H. Haveliwala Stanford University
20. The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank Matthew Richardson Pedro Domingos Department of Computer Science and Engineering University of Washington Box 352350 Seattle, WA 98195-2350, USA
21. Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization Rada Mihalcea Department of Computer Science University of North Texas
22. Hyperlink Analysis: Techniques and Applications Prasanna Desikan, Jaideep Srivastava, Vipin Kumar, and PangNing Tan Department of Computer Science, University of Minnesota, Minneapolis, MN, USA {desikan, srivastava, kumar, ptan}
23. J. Srivastava, R. Cooley, M. Deshpande, and P. –N. Tan. “Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data” (2000), SIGKDD Explorations, Vol. 1, Issue 2, 2000.
24. HAR: Hub, Authority and Relevance Scores in Multi-Relational Data for Query Search, Xutao Li, Michael K. Ng, Yunming Ye
25. Authoritative Sources in a Hyperlinked Environment, JON M. KLEINBERG Cornell University, Ithaca, New York
26. Analysis of data mining techniques for increasing search speed in web, B.Chaitanya Krishna, C.Niveditha, G.Anusha , U.Sindhu , Sk.Silar
27. Comparative Analysis Of Pagerank And HITS Algorithms Nidhi Grover MCA Scholar Institute of Information Technology and Management