

# Comparative Study of HITS and PageRank Link based Ranking Algorithms

Pooja Devi<sup>1</sup>, Ashlesha Gupta<sup>2</sup>, Ashutosh Dixit<sup>3</sup>

M.Tech Student, CE Department, YMCA University of Science and Technology, Faridabad, India<sup>1</sup>

Assistant Professor, CE Department, YMCA University of Science and Technology, Faridabad, India<sup>2,3</sup>

**Abstract:** World Wide Web is a huge repository of information resources that include text, audio, video etc. As the amount of information available on web is increasing it is difficult to acquire information on web. Therefore users today mainly depend upon various search engines for finding suitable answers for their queries. Search engines may return millions of pages in response to a query. It is not possible for a user to preview all the returned resultset. So search engine make use of ranking algorithm to display the resultant pages in a ranked order using different page ranking algorithms. In this paper, we compare two popular Link based ranking algorithms namely: HITS algorithm and PageRank algorithm. Relative strengths and limitations of these two algorithms are explored to find out further scope of research.

**Keywords:** Search Engine, PageRank, HITS, Hub, Authority, Link Based Search.

## I. INTRODUCTION

World Wide Web is a vast resource of hyperlinked and heterogeneous information including text, audio, video and metadata. It is estimated that WWW is doubling in size every six to ten months. Due to the rapid growth of information resources on World Wide Web it is difficult to manage the information on the web. Therefore it has become necessary for the users to use efficient information retrieval techniques to find and order the desired information. Search engines play an important role in searching web pages. The search engine [1] gathers, analyzes, organizes the data from the internet and offers an interface to retrieve the network resources. Search engines [1] are “programs” that search documents for specified keywords and returns a list of the documents where the keywords were found. The search results are generally presented in a line of results often referred to as search engine results pages (SERPs). Figure 1 represents the general architecture of a Search Engine. The major components of a Search Engine are the Crawler, Indexer, Query Processor. A crawler or spider is a program that traverses the web by following hyperlinks and storing downloaded pages in a large database. The crawler starts with a seed URL and collects documents by recursively fetching links and storing the extracted URL’s into a local repository. Indexer extracts the terms from each web page and records the URL where each word has occurred. Query Processor is responsible for receiving and filling search requests from the user. When a user fires a query, query engine searches the web page in the index created by the indexer and returns a list of URL’s of the web pages that match with the user query.

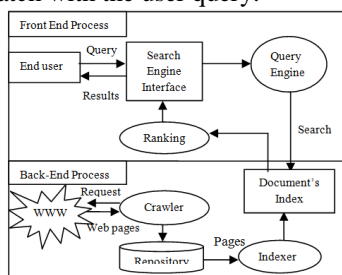


Fig 1 Architecture of search engine

In general Query Engine may return several hundreds and thousands of URL in response to a user query which includes a mixture of relevant and irrelevant information. Since no user can read all web pages returned in response to the user query, Page Ranking mechanisms are used by most search engines for putting the important pages on top leaving less important in the bottom of the result list. Popular Page Ranking algorithms used are Page Rank algorithm, Hypertext Induced Topic Search (HITS), Weighted Page Rank algorithm, Page Content Rank etc.

## II. RANKING ALGORITHMS

Web-page ranking [3] is an optimization technique used by search engines for ranking hundreds and thousands of web pages in a relative order of importance. To rank a web page different criteria are used by ranking algorithms. For example some algorithms consider the link structure of the web page while others look for the page content to rank the web page. Broadly Page Ranking algorithms can be classified into two groups Content-based Page Ranking and Connectivity-based Page Ranking [7, 8].

**Content-based Page Ranking:** In this type of ranking the pages are ranked based on their textual. Factors that influence the rank of a page are :

- Number of matched terms with the query string
- Frequency of terms i.e the number of times the search string appears in the page. The more time the string appears, the better is the page ranking
- Location of terms i.e query string could be found in the title of a page or in the leading paragraphs of a page or even near the head of a page.

**Connectivity-based Page Ranking (Link based):** This type of ranking work on the basis of link analysis technique. They view the web as a directed graph where the web pages form the nodes and the hyperlinks between the web pages form the directed edges between these nodes. There are two famous link analysis methods:

- PageRank Algorithm
- HITS Algorithm and

Section III and IV discusses these two algorithms respectively.

### III. PAGERANK

The PageRank algorithm was developed at Stanford University by Larry Page [4] and Sergey Brin in 1996. PageRank algorithm, named after Larry Page and used by the Google Internet search engine uses the link structure of the web to determine the importance of the web page. Here a page obtains a higher rank if sum of its back-links is high. This algorithm is based on random surfer model. The random surfer model assumes that a user randomly keeps on clicking the links on a page and if she/he gets bored of a page then switches to another page randomly. Thus, a user under this model shows no bias towards any page or link. PageRank(PR) is the probability of a page being visited by such user under this model. For each web page, Page Rank value is pre-computed. For this over 25 billion web pages on the WWW are considered to assign a rank value.

A Simplified version of PageRank is defined in Equation (1)

$$PR(A) = c \sum_{v \in T_A} \frac{PR(v)}{Q_v} \dots \dots \dots (1)$$

Where A is a web page whose PageRank is to be calculated.  $T_A$  is the set of pages A points to and  $S_A$  is the set of pages that point to A.  $Q_v$  is the number of links from A and c is a factor used for normalization (so that the total rank of all web pages is constant).

The rank of a page is divided among its forward links evenly to contribute to the ranks of the pages they point to. Note that  $c < 1$  because there are a number of pages with no forward links and their weight is lost from the system. The presented equation is recursive but it may be computed by starting with any set of ranks and iterating the computation until convergence.

Page Rank algorithm assumes that if a page has a link to another page then it votes for that page. Therefore, each inlink to a page raises its importance. Following is a modified version of the PageRank algorithm.

$$PR(A) = (1 - d) + d \left( \frac{PR(T_1)}{Q(T_1)} + \dots \dots \dots \frac{PR(T_n)}{Q(T_n)} \right) \dots \dots \dots (2)$$

Where:

- PR(A) = PageRank of page A
- $T_1 \dots T_n$  = All pages that link to page A
- PR( $T_i$ ) = Page rank of page  $T_i$
- Q( $T_i$ ) = the number of pages to which  $T_i$  links to
- d = damping factor which can be set between 0 and 1
- PR( $T_i$ )/Q( $T_i$ ) = PageRank of  $T_i$  distributing to all pages that  $T_i$  links to.
- (1-d) = To make up for some pages that do not have any out-links to avoid losing some page ranks.

Damping factor: The PageRank theory holds that any imaginary surfer who is randomly clicking on links will eventually stop clicking. The probability, at any step, that the person will continue is called a damping factor *d*. The

damping factor can be set to any value such that  $0 < d < 1$ , nominally it is set around 0.85. The damping factor is subtracted from 1 and this term is then added to the product of the damping factor and the sum of the incoming PageRank scores.

#### A. Implementation of Page Rank Algorithm

The following steps explain the method for implementing Page Rank Algorithm.

Step 1: Initialize the rank value of each page by  $1/n$ . Where n is total no. of pages to be ranked. Suppose we represent these n pages by an Array of n elements. Then  $A[i] = 1/n$  where  $0 \leq i < n$

Step 2: Take some value of damping factor such that  $0 < d < 1$ . e.g 0.15, 0.85 etc.

Step 3: Repeat for each node i such that  $0 \leq i < n$ . Let PR be an Array of n element which represent PageRank for each web page.

$$PR[i] \leftarrow 1-d$$

For all pages Q such that Q Links to PR[i] do

$$PR[i] \leftarrow PR[i] + d * A[Q]/Q_n$$

where  $Q_n$  = no. of outgoing edges of Q

Step 4: Update the values of A

$$A[i] = PR[i] \text{ for } 0 \leq i < n$$

Repeat from step 3 until the rank value converges i.e. values of two consecutive iterations match.

#### B. Advantages of PageRank

The strengths of PageRank algorithm are as follows:

- Less Query time: PageRank has a clear advantage over the HITS algorithm, as PageRank compute ranking at crawling time so response to user query is quick.
- Less susceptibility to localized links: Furthermore, as PageRank is generated using the entire Web graph, rather than a small subset, it is less susceptible to localized link.
- More Efficient [6]: In contrast, PageRank computes a single measure of quality for a page at crawl time. This measure is then combined with a traditional information retrieval score at query time. Compared with HITS, this has the advantage of much greater efficiency.
- Feasibility: As compared to Hits algorithm the PageRank algorithm is more feasible in today's scenario since it performs computations at crawl time rather than query time.

#### C. Disadvantages of PageRank

The following are the problems or disadvantages of PageRank [3]:

- Less Relevant to user Query: PageRank score of a page ignores whether or not the page is relevant to the query at hand.
- Rank Sinks: The Rank sinks problem occurs when in a network pages get in infinite link cycles.

- It is a static algorithm that, because of its cumulative scheme, popular pages tend to stay popular generally. Popularity of a site does not guarantee the desired information to the searcher.
- In Internet, available data is huge and the algorithm is not fast enough.
- Spider Traps: Another problem in PageRank is Spider Traps. A group of pages is a spider trap if there are no links from within the group to outside the group.
- Dangling Links [6]: This occurs when a page contains a link such that the hypertext points to a page with no outgoing links. Such a link is known as Dangling Link.
- Dead Ends: Dead Ends are simply pages with no outgoing links. PageRank doesn't handle pages with no outedges very well, because they decrease the PageRank overall.
- Circular References: If you have circle references in your website, then it will reduce your front page's PageRank.

#### IV. HITS

Hypertext Induced Topic Search (HITS) or hubs and authorities is a link analysis algorithm developed by Jon Kleinberg [11] in 1998 to rate Web pages. The HITS algorithm is an iterative algorithm developed to quantify each page's value as an authority and as a hub. The premise of the algorithm is that a web page serves two purposes: to provide information on a topic, and to provide links to other pages giving information on a topic. So it categorizes a web page in two ways:

- Authority: pages that provide important and trustworthy information on a given topic. So an authority is a page that is pointed by many hubs.
- Hub: pages that contain links to authorities' i.e pointing to many pages.

Figure below depicts the hubs and authorities created by HITS.

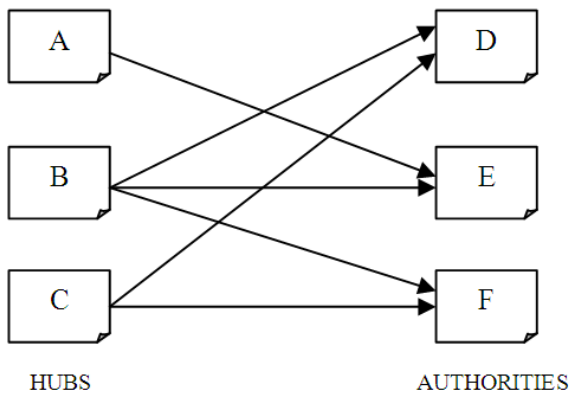


Fig 2 Hubs and Authorities

In HITS [12] algorithm, ranking of the web page is decided by analyzing their textual contents against a given query. After collection of the web pages, the HITS algorithm concentrates on the structure of the web only, neglecting their textual contents. HITS [6] applied on a subgraph after a search is done on the complete graph.

#### A. Implementation of HITS Algorithm

The following steps explain the method for implementing HITS Algorithm.

Step 1: In the first step of the HITS algorithm we determine a base set S.

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

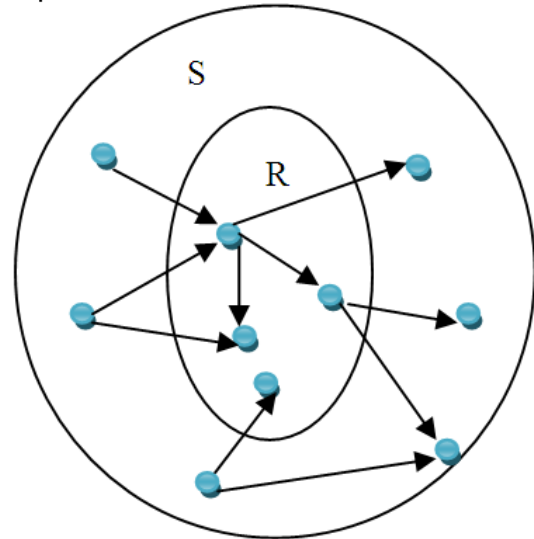


Fig 3 Expansion of the root set R

Step 2:

- Add to S all pages pointed by any page in R.
- Add to S all pages that point to any page in R.
- For each node p initialize the a(p) and h(p) to 1.

Step 3: In each iteration update the authority weight and the hub weight for each node in S. We can represent the subgraph in the form of matrix.

Say, n pages are retrieved in response to a search query, then HITS algorithm forms the n by n adjacency matrix A, whose m(i, j) element is 1 if page i links to page j and 0 otherwise.

It then iterates the following equations

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

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

Where "i → j" means page i links to page j and  $a_i$  is authority of  $i^{th}$  page and  $h_i$  is the hub representation of  $i^{th}$  page.

For eg:

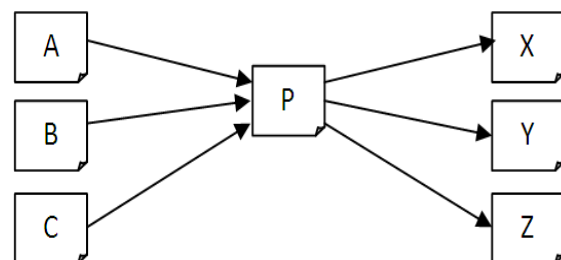


Fig 4 authority and hub of a page P

Authority of page P is given as:

$$a(P) = h(A) + h(B) + h(C)$$

Hub of page P is given as:

$$h(P) = a(X) + a(Y) + a(Z)$$

Step 4: Normalization: The final hub-authority scores of nodes are determined after infinite repetitions of the algorithm. In each iteration diverging values of authority and hub are obtained. So, it is necessary to normalize the values after each iteration. Normalization [12] is done by dividing each Hub score by the square root of sum of the squares of all the Hub scores, and dividing each Authority score by the square root of sum of the squares of all the Authority scores.

### B. Advantages of HITS

Here are some considerable advantages of HITS [6]:

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

### C. Disadvantages of HITS

Here are some notable disadvantages of HITS algorithm[2]:

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

nodes and as a result the algorithm will not be able to find the highest ranked authorities and hubs for a given query.

- Less Feasibility: As HITS compute Rank value at query time, it is not feasible for today's search engines, which need to handle tens of millions of queries per day.

## V. IMPLEMENTATION RESULTS

For implementation purpose, the top most web pages returned by a popular search engine for a user query are being considered to calculate Rank values for both the algorithms. These web pages are being represented as a web graph as shown in fig 5. In this graph nodes are representing the web pages and edges represent the links between the web pages The functionality of HITS and PageRank algorithm are being demonstrated with the help of this web graph.

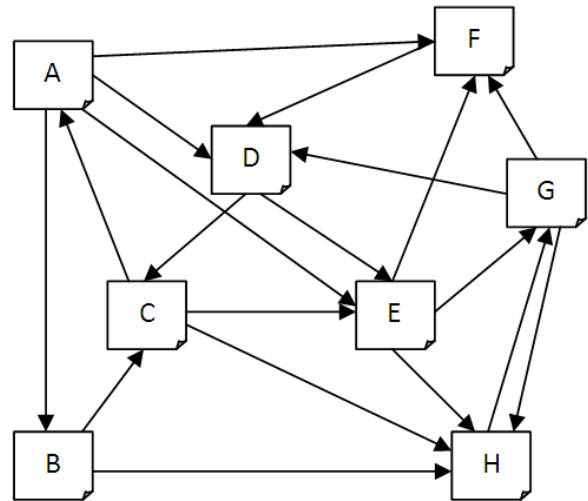
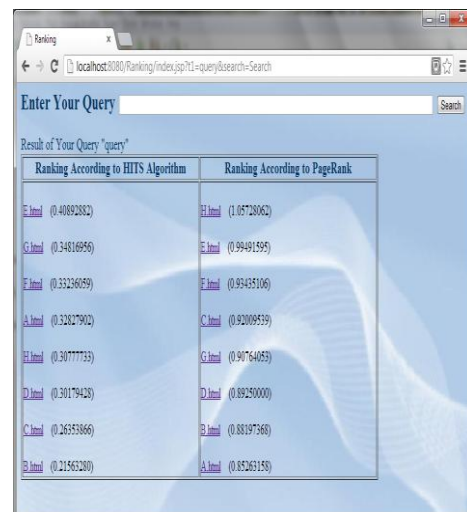


Fig 5 Graph for implementation

Result of Ranking according to HITS Algorithm (average of Authority score and Hub score) and Page Rank algorithm of the above graph is shown below at iteration 5 at damping factor  $d=0.85$



Ranking According to HITS Algorithm		Ranking According to PageRank	
E.html	(0.40892882)	F.html	(1.05728062)
G.html	(0.34816956)	E.html	(0.99491595)
F.html	(0.33246059)	F.html	(0.89435106)
A.html	(0.33027902)	C.html	(0.82008539)
E.html	(0.30777133)	G.html	(0.90764053)
D.html	(0.30179420)	D.html	(0.89250000)
C.html	(0.26535866)	B.html	(0.88197368)
B.html	(0.21565280)	A.html	(0.85263155)

Fig 6 Comparison of Rank values According to HITS and PageRank

A. PageRank Results:

TABLE I  
PAGERANK OF DIFFERENT PAGES IN GRAPH AT  
ITERATION 1, 3 AND 5 AT D=0.85

	Iteration 1	Iteration 3	Iteration 5
A	0.16491228	0.24568575	0.31025027
B	0.18504385	0.20220822	0.21592818
C	0.25101206	0.46449374	0.65434887
D	0.24469298	0.76801554	1.10409663
E	0.36015846	0.66022139	0.87056810
F	0.30200103	0.59063419	0.76383872
G	0.29678173	0.91374562	1.18073432
H	0.48589678	0.81350237	1.00837068

TABLE 2  
PAGERANK OF DIFFERENT PAGES IN ITERATION  
1, 3 AND 5 AT D=0.15

	Iteration 1	Iteration 3	Iteration 5
A	0.85263157	0.89916037	0.89987763
B	0.88197368	0.88371851	0.88374541
C	0.92009539	0.99646400	0.99764925
D	0.89250000	1.08362490	1.08493926
E	0.99491595	1.01481358	1.01499831
F	0.93435106	0.98742116	0.98755703
G	0.90764053	1.06109662	1.06124534
H	1.05728062	1.06989759	1.06997555

B. HITS Algorithm Results:

Authority Score:

TABLE 3  
AUTHORITY SCORE OF DIFFERENT PAGES IN  
GRAPH IN ITERATION 1, 2 AND 3.

	Iteration 1	Iteration 2	Iteration 3
A	0.13736056	0.13732008	0.13492265
B	0.13736056	0.17165010	0.17919414
C	0.27472113	0.18881511	0.16443698
D	0.41208169	0.39479524	0.40687611
E	0.41208169	0.39479525	0.38579445
F	0.41208169	0.49778531	0.51228443
G	0.27472113	0.18881512	0.17708597
H	0.54944226	0.56644536	0.56077225

Hub Score:

TABLE 4  
HUB SCORE OF DIFFERENT PAGES IN GRAPH AT  
ITERATION 1, 2 AND 3.

	Iteration 1	Iteration 2	Iteration 3
A	0.48853197	0.51148238	0.51984395
B	0.29311918	0.26476735	0.25401466
C	0.39082557	0.38511614	0.37880674
D	0.24426598	0.20459295	0.19272624
E	0.43967877	0.43927310	0.43787992
F	0.14655959	0.13840111	0.14251404
G	0.48853197	0.51148238	0.51836712
H	0.09770639	0.06619184	0.06202683

VI. COMPARISON OF PAGERANK AND HITS  
[2, 5]

Criteria	PageRank	HITS
Mining Techniques	Web Structure	Web Structure and Web Content
Working Process	Computes Rank values at index time and results are sorted on the priority of pages.	'n' highly relevant pages rank are computed.
Input Parameters	Inlinks to a page.	Inlinks, outlinks and content
Relevance	Less( as this algo ranks the page at indexing time)	More( as this uses hyperlink structure and also consider the content of the page.
Quality of Result obtained	Medium	Less than PageRank algorithm
Advantages	<p>→Query-time cost of incorporating precomputed PageRank importance score for a page is low.</p> <p>→PageRank generated using the entire Web graph, rather than a small subset, it is Less susceptible to localized link spam.</p> <p>→PageRank may be used as a methodology to measure the impact of a community like</p>	<p>→HITS is a general algorithm used for calculating the authority and hubs in order to rank the retrieved data</p> <p>→The basic aim of that algorithm is to induce the Web graph by finding set of pages with a search on a given topic (query).</p> <p>→Results demonstrates that it is good in</p>

	the blogosphere on the overall Web itself.	calculating the authority nodes and hubness
Disadvantages	<ul style="list-style-type: none"> <li>→Rank Sink</li> <li>→Spider Traps</li> <li>→Dangling Links</li> <li>→Dead Ends</li> <li>→Circular References Effect of additional pages</li> </ul>	<ul style="list-style-type: none"> <li>→Irrelevant authorities</li> <li>→Irrelevant Hubs problem</li> <li>→Mutually reinforcing relations between hosts problematic</li> <li>→Topic Drift</li> </ul>
Search Engine	Used in Google	Used in IBM search engine Clever

[6]. Ritika Wason, Nidhi Grover, Comparative Analysis of Pagerank And HITS Algorithms, International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 8, October – 2012

[7]. Ricardo Baeza-Yates and Emilio Davis, Web page ranking using link attributes, In proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, PP.328-329, 2004.

[8]. Aallan borodin, Link Analysis Ranking: Algorithms, Theory, and Experiments, University of Toronto

[9]. L. Page, S. Brin, R. Motwani, and T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, Stanford Digital Libraries SIDL-WP-1999-0120, 1999.

[10]. Székely Endre Google and the Page Rank Algorithm, slides 2007. 01. 18.

[11]. J.Kleinberg, Authoritative sources in a hyperlinked environment, Journal of ACM (JASM), 1999

[12]. L. Li, Y. Shang, and W. Zhang, Improvement of HITS-based algorithms on web documents, in Proceedings of the Eleventh International Conference on the World Wide Web, May 2002.

## VII. CONCLUSION

On the basis of this study we conclude that both page rank and HITS algorithm are different link analysis algorithms that employ different models to calculate web page rank. The PageRank and HITS algorithm give importance to links rather than the content of the pages. According to PageRank algorithm, rank score of a web page is divided evenly over the pages to which it links whereas HITS algorithm rank pages according to authority and hubness of a page. Page Rank is a more popular algorithm used as the basis for the very popular Google search engine. This popularity is due to the features like efficiency, feasibility, less query time cost, less susceptibility to localized links etc. which are absent in HITS algorithm. However though the HITS algorithm itself has not been very popular, different extensions of the same have been employed in a number of different web sites. Results demonstrate that HITS calculates authority nodes and hubness correctly. HITS may also be combined with other information retrieval based rankings. After going through exhaustive analysis of PageRank and HITS algorithms for ranking of web pages against the various parameters such as methodology, input parameters, relevancy of results and importance of the outcome, it is concluded that these techniques have limitations particularly in terms of time response, accuracy of results, importance of the outcome and relevancy of results. An efficient web page ranking algorithm should meet out these challenges efficiently with compatibility with global principles of web technology.

## REFERENCES

[1]. C. Ridings and M. Shishigin, Pagerank Uncovered Technical report,2002.

[2]. Dilip Kumar Sharma, A Comparative Analysis of Web Page Ranking Algorithms, (IJCE) International Journal on Computer Science and Engineering Vol. 02, No. 08, 2010, 2670-2676

[3]. Mercy Paul Selvan, Ranking Techniques for Social Networking Sites based on Popularity, Journal of Computer Science and Engineering (IJCE).

[4]. C. Ridings and M. Shishigin, Pagerank Uncovered, Technical report,2002.

[5]. Neelam Duhan ,A.K.Sharma and Komal Kumar Bhatia , Page Ranking Algorithms : A Survey, In proceedings of the IEEE International Advanced Computing Conference (IACC),2009