# NOVEL TECHNIQUE PAGE RANKING FOR EFFICIENT INFORMATION RETRIEVAL

**[1]Mr. Nidheesh Sharma, [2] Mrs. Nidhi Tyagi, 3Mr. Rajesh Pandey**

*[1]MCA, Dr. K.N. Modi Institute of Engg & Tech, Modinagar (India)*

*[2]B.Tech CSE, Shobhit University, Meerut (India)*

*[3]B.Tech CSE/IT, Dr. K.N. Modi Institute of Engg & Tech, Modinagar (India)*

## ABSTRACT

*Page Rank is probability distribution used to represent the likelihood that a person randomly clicking on links will arrive at any particular page. Search engines calculate the page rank of a page with the help of different algorithms. The proposed solutions for the page ranking, count all the inner and outer links of the page, and simultaneously count the hits on a particular page to find out whether that the page is popular or not in user environment and on the bases of the two calculated parameters assigns the page rank value. This novel technique helps the user to retrieve the relevant pages faster.*

## I INTRODUCTION

The World Wide Web (WWW) [1] is an interlinked collection of documents formatted using Hyper Text Markup Language (HTML). These documents contain hyperlinks to other documents. The links can point to a document on the same machine or to one on the other side of the world. In order to extract information from the WWW, there is need of a tool to search the Web. The tool is called a search engine [2]. The queries are submitted by the user through the search engine, a computer program that searches for particular keywords and return a list of documents in which they were found. A crawler[3] (also referred as robots spiders, worms, wanderers) is a program that automatically collects web pages to create a local index and or a local collection of web pages, for a web search engine. Page Rank is probability distribution used to represent the likelihood that a person randomly clicking on links will arrive at any particular page. After Web pages are gathered to the site of a search engine, they are pre-processed into a format that is suitable for effective and efficient retrieval by search engines. Search engines calculate the Page Rank [4][5] of a page with the help of different algorithms. Many engines calculate the Page Rank of a page by counting the inner and outer links. Some calculate the Page Rank by counting the hits on that page and some by the weight of the page. If all the inner and outer links of a page are counted to calculate the page rank, it is not necessary that the page has more number of users using that page. The proposed solutions for the page ranking, count all the Inner and Outer links of the page, and simultaneously count the hits on a particular page to find out whether that the page is popular or not in user environment and on the bases of the two calculated



IJARSE

parameters assigns the page rank value. This novel technique helps the user to retrieve the relevant pages faster. The paper is divided into four sections , the section 2 discusses the related work in this direction, section 3 gives the proposed work, with the algorithm and the related example, finally conclusion id discussed in section5.

## II RELATED WORK

Page Rank [6] reflects the view of the importance of web pages by considering more than 500 million variables and 2 billion terms. Pages that we believe are important pages receive a higher Page Rank and are more likely to appear at the top of the search results. Page Rank also considers the importance of each page that casts a vote, as votes from some pages are considered to have greater value, thus giving the linked page greater value. We have always taken a pragmatic approach to help improve search quality and create useful products, and our technology uses the collective intelligence of the web to determine a page's importance. A Page Rank results from a mathematical algorithm based on the graph created by all World Wide Web pages as nodes and hyperlinks, The rank value indicates an importance of a particular page. A hyperlink to a page counts as a vote of support. The Page Rank of a page is defined recursively and depends on the number and Page Rank metric of all pages that link to it ("incoming links"). A page that is linked to by many pages with high Page Rank receives a high rank itself. If there are no links to a web page there is no support for that page.

Recently, new algorithms have been created that greatly improve rankings using the network structure of the web. Kleinberje's hits algorithm a method of link analysis used the link structure [9]of a network of web pages to assign authority and hub weight to each page. They have found that certain tree link web structures can lead the hits algorithm to return either arbitrary or non intuitive result. They present two modifications to the adjacency matrix input to the hits algorithm exponentiated input our on direct links but also on longer paths between pages. It resolves both limitations mentioned above. The HITS algorithm is a iterative algorithm develop to quantify is a iterative algorithm develop to quantify each page value as a hub and an authority.

The SALSA (Stochastic Approach for Link Structure Analysis) algorithm, developed by Lempel and Moran [7], combines the random walk idea of Page Rank with the hub/authority idea of HITS. A significant advantage of SALSA is that the weightings can be computed explicitly without the iterative process described in [10]. Although the initial paper provides an explicit calculation only in the case of uniform initial weights, this quick method of calculating the weightings can be easily generalized to accommodate non uniform initial weights.

The critical look at the available literature reveals that

- If all the inner and outer links of a page are counted to calculate the page rank, it is not necessary that the page has more number of users using that page
- If the hit are counted of a particular page, it will contain all the information of the related topic.
- If the Keyword is used more in a given text on a particular page. It does not indicate that page is a popular page.



IJARSE

The above stated reason make it importance to develop an improve technique to the Page Rank of web page to improve the efficiency of retrieval system of term of time and relevancy

## III PROPOSED WORK

In the proposed work, we have built a Ranking System which first of all learn and built an index from the seed URL. For subsequent resources on the web, a component has been made that determines whether the resource is relevant thereby also classify the links embedded in the resource. The proposed work used the html tags URL name and similarity for the classification of the pages. It means the proposed work consider the link base and hit base technique, which result better precision than the ranking system. Reputation and Ranking System are an essential part of web search. No able example are Hits and Page Rank, these algorithm use the global Hyperlink structure of the web to determine the importance of individual pages. Each use notation of endorsements among pages, so we also refer to them as link based reputation system. The proposed works used the inner and outer link and count the hits of the page. It means the proposed work consider the link based classification technique. We develop this technique in .net platform. Which show the result on net platform and results in better precision than the ranking system that are only based on standard classification.

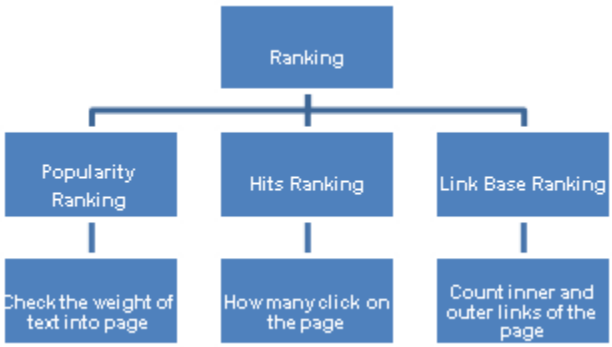


**Fig. 1 Ranking Architecture**

### 3.1 Steps of Ranking Algorithm

- Any user type the text in the search engine which he/she wants to search. After writing the text in searching text box the user click on search button.
- The crawler retrieves resources from the seed URL and requests the classifier to build a knowledge base.
- The classifier user the purpose algorithm to approved or rejects a page.
- Firstly the user calculates the Rank of a page by following the instruction given below.
  - ❖ The user count all the links on a given page which include both the types of link i.e. inner and outer

- The user counts the hits given on a page.
- The value of hits & links are added and then divided by 2 for calculating the Page Rank.

### 3.2 Algorithm for the set as default value

Step 1 - [Initialize default values with page name, hit count and link count of the page as 0]

STR=Page1:0,0#Page2:0,0#Page3:0,0#Page4:0,0#Page5:0,0#Page6:0,0#Page7:0,0#Page8:0,0#Page9:0,0#Page10:0,0

Step 2 - [Write this string in the file saving the information about the page]

[Pages are separated by #, Pages and their corresponding value is separated by : and, Link Count and hit count are respectively separated by , ]

File.AppendAllText(StringFilePath,"Page1:0,0#Page2:0,0#Page3:0,0#Page4:0,0#Page5:0,0#Page6:0,0#Page7:0,0#Page8:0,0#Page9:0,0#Page10:0,0")

### 3.3 Algorithm for the Page Count

Step I - [Initialize the Function With the Page URL]

Step 2 - OURI =[URL Parameter1]
OURI=Get page name from this URL
ICount=0

Step 3 - PageStrem= Get all response of the Page

Step 4 - Repeat Step 6 while PageStrem contains '<A href'

Step 5 - PageStrem= Remaining stream except the previous one

Step 6 - Icount=ICount + 1

Step 7 - Return Icount

Step 8 - EXIT

IJARSE

### 3.4 Algorithm for Page log with hit Count and Link Count

Step 1 - [Take all string from file which contains Page hit count and link count detail, StringFilePath is the File Path on the Disk] sText = File.ReadAllText(StringFilePath)

Step 2 - [Split sTest to seperate the pages and their values. These are separated by #]

PageArray = sText.Split("#")

Step 3 - Repeat each Page and take it into PageName type array.

Step 4 - [Split Pages and their values, Store it in HitsCountArray] HitsCountArray =

PageName.Split(":")(1).Split(",")

Step 5 - [Now Seperate these HitCountArray into the HitCount and Link Counnt]

Step 6 - [Get Link Count by using Page Reference Count Algorithm and get it into a

variable.] LinkCount=Page Reference Count Algorithm[Page URL]

Step 7 - [Replace the values of Page with corresponding HitCount and Link Count]

Step 8 - [Save it back into the File]

Step 9 - EXIT.

**Table 1: Result after the implementation of proposed algorithm**

| Page | Hit Count | Link Count | Rank |
|---|---|---|---|
| Page 1 | 6 | 147 | 79.5 |
| Page 2 | 7 | 142 | 78 |
| Page 3 | 0 | 0 | 0 |
| Page 4 | 4 | 132 | 70 |
| Page 5 | 0 | 0 | 0 |
| Page 6 | 0 | 0 | 0 |
| Page 7 | 0 | 0 | 0 |
| Page 8 | 1 | 146 | 73.5 |
| Page 9 | 0 | 0 | 0 |
| Page 10 | 1 | 145 | 71 |

IJARSE

**Table 2. Sequentially arranged pages according to evaluated ranks**

| Page | Hit Count | Link Count | Rank |
|---|---|---|---|
| Page 1 | 6 | 147 | 79.5 |
| Page 2 | 7 | 142 | 78 |
| Page 8 | 1 | 146 | 73.5 |
| Page 10 | 1 | 145 | 71 |
| Page 4 | 4 | 132 | 70 |
| Page 3 | 0 | 0 | 0 |
| Page 5 | 0 | 0 | 0 |
| Page 6 | 0 | 0 | 0 |
| Page 7 | 0 | 0 | 0 |
| Page 9 | 0 | 0 | 0 |

The example given for the proposed work reveals that the evaluation of the page rank is more appropriate and the page which has the more hits, links count is considered to be for important for the user, as in Table 1 and Table 2.

## IV CONCLUSION

In the proposed work, solution to remove the various drawback of the existing problem have been suggested. If we count all the inner and outer links of the page and count the hits on a particular page to find out whether that the page is popular or not in user environment. Add the value of hits and obtained links to calculate the page rank of a page. This novel technique will improve the efficient of the Page Ranking system in term of relevance and time of retrieval. The example given for the proposed work reveals that the evaluation of the page rank is more appropriate and the page which has the more hits, links count is considered to be for important for the user. In the future work the similarity of the key world with the title of the pages may be implemented and added to the ranking system which will further improve the efficiency of web page ranking system.

## REFERENCES

[1] Duglas E.Comer,"The Internet Book" , Prentice Hall of India , New Delhi ,2001.

[2] Sunny Lam :"Overview of Web Search Engine", 2001.

[3] Gray, M., 1996. Internet Statistics: Growth and Usage of the Web and the Internet, http://www.mit.edu/people/mkgray/net/.



IJARSE

[4] Burner,M.,1997.Crawling towards Eternity : Building An Archive of The World Wide Web , in Web Techniques Magazine, Vol. 2.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd. The Page Rank Citation Ranking: Bring Order to the Web. Technical Report, Stanford University, 1998.

[6] http://en.wikipedia.org/wiki

[7] R. Lempel and S. Moran, The stochastic approach for link-structure analysis (SALSA).

[8] Balamurugan,Newlin Rajkumar,J.Preethi " Design and Implementation of a New Model Web Crawler with Enhanced Reliability "2008

[9] Kleinberg. Authoritative sources in a hyperlinked environment. J. ACM, 46:604–632, 1999.

[10] Junghoo Cho and Hector Garcia – Molina, "The Evolution of the Web and implementation for an incremental crawler" , Prc. of VLDB Conf.,2000.

IJARSE