Review Article | Impact Factor: 0.621 | ISSN: 2319-507X
PB Niranjane, IJPRET, 2014; Volume 2 (8): 734-744 | IJPRET

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

REVIEW OF SEARCH ENGINE ALGORITHM

PROF. P. B. NIRANJANE, RACHANA A. AGRAWAL

1. Assistant Professor, Department Of CSE, B.N.C.O.E., Pusad, India.
2. M. E. Student, Department Of CSE, B.N.C.O.E., Pusad, India.

Accepted Date: 15/02/2014 ; Published Date: 01/04/2014

Abstract: The use of the web resources increased every day. Searching techniques are needed for to extract appropriate information from the web, as the users require correct and complex information from the web. Search algorithm means the billions of web pages and other information it has, in order to return what it believes are the best answers. Search engine contain many algorithm to find related data of the keywords. Web mining contains three categories that also explain. Ranking means arrange link in ascending or descending order with respect to our keywords. These paper described the algorithm for search engine. Searching algorithm such as page rank algorithm and HITS algorithm. Both algorithms are link analysis algorithm. The working of algorithms is explained in the paper.

Keywords: Information Retrieval, Search Engine, algorithm.

PAPER-QR CODE

Corresponding Author: PROF. P. B. NIRANJANE

Access Online On:

www.ijpret.com

How to Cite This Article:

PB Niranjane, IJPRET, 2014; Volume 2 (8): 734-744

INTRODUCTION

Many kinds of web search engines, such as Yahoo, fresh eye, Google and so many others[7], have been developed. The World Wide Web (WWW) has evolved into a massive collection of information. The web resources increase everyday and that in turn increases the complexity of accessing the web data[1]. Web sources usage increase every day. People uses internet in day to day life. There is a need for efficient searching techniques to extract appropriate information from the web, as the users require correct and complex information from the web. There are three web mining categories such as web content mining, web structure mining and web usage mining [2]. Web usage mining searches the user logs and identifies the user's usage pattern of the web. The web structure mining analyses the connections in the web pages. Web content mining is involved in finding useful information from the content of the web such as text, audio, video, images and etc. Various web mining techniques are discussed in [3], [5] to search WWW.

A. Web Search Engine

A Web Search Engine is a Search Engine that searches for information on the WWW and returns a list documents in which the search query's key words are found. Web Search Engines are classified into general purpose search engine such as Google and Vertical Search Engine. A web search query is a query that a user enters into web search engine to satisfy his or her information needs. Web search queries are unstructured and often ambiguous. Search engines assume that users are capable of creating appropriate search queries. Search Engines such as iMed creates search queries for the user by interacting with theuser [6]. The Web Search Engines simple interface is shown in fig.

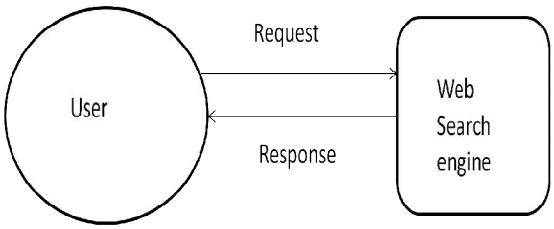

Fig.1 Simple interface

B. Information Retrieval

Information Retrieval is the science of searching for information and retrieving information. Web search is also a type of information retrieval as the user search for information from the web and receive information from the web. The application used to search and locate

appropriate documents on selected topics from a database of texts is a text-based application [5]. The efficiency of a search facility is measured using two metrics Precision and Recall. Precision specifies whether the documents retrieved are relevant and Recall specifies whether all the relevant documents are retrieved.

Precision = Relevant and Retrieved / Retrieved

Recall = Relevant and Retrieved / Relevant

Precision can be calculated as the ratio of the relevant web pages that are retrieved to the number of web pages that are retrieved. It is not possible to calculate the recall as it is not possible to find the number of relevant web pages out of the millions of web pages present in the WWW. The recall values can be calculated when the number of documents to be accessed is known in advance such as the files in an organization. Ranking means arrange link in ascending or decending order with respect to our keywords. Search engine contain different algorithm such as Hilltop algoritm,Trust algorithm, HITS algorithm and Page rank algorithm.

I. RANKING METHODS:

A. HITS algorithm

The current search engine ranking algorithm Hyperlink-Induced Topic Search (HITS; also known as hubs and authorities) is a link analysis algorithm that rates Web pages, developed by Jon Kleinberg. HITS is a ranking algorithm which ranks “hubs” and “authorities[4]. It is an iterative algorithm based on the linkage of the documents on the web, like Page and Page Rank. The algorithm assigns two scores for each page. One is authority, which estimates the value of the content of a page. The other is hub value, which estimates the value of its links to other pages. The idea behind Hubs and Authorities stemmed from a particular insight into the creation of web pages when the Internet was originally forming that is, certain web pages, known as hubs, served as large directories that were not actually authoritative in the information that it held, but were used as compilations of a broad catalog of information that led users directly to other authoritative pages. In other words, a good hub represented a page that pointed to many other pages, and a good authority represented a page that was linked by many different hubs. The reason why the algorithm of HITS cannot be used in large-scale search engine and satisfy search demands not only because of slower query speed and poorer real time but also it is a partial analysis of web pages depending on users’ demands.

Authority

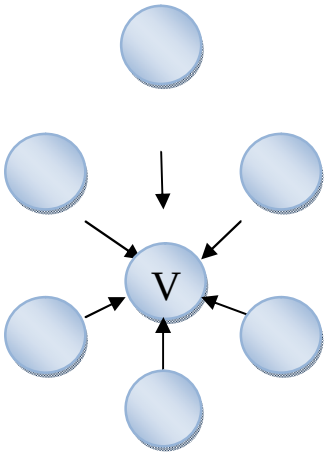

Fig.4 Authority

Hub

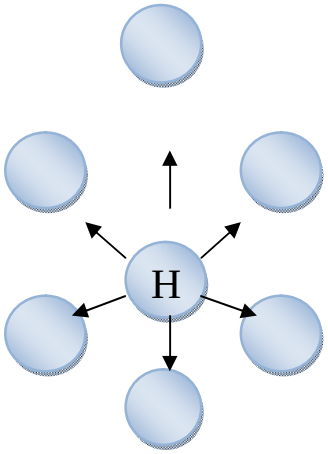

Fig.3 Hub

Each page has two weights

1. Authority weight a(v)

A good authority has many edges from good hubs.

$a(v) = \sum_{u \to v} h(u)$

2. Hub weight h(v)

A good hub has many outgoing edges to good authorities

$h(v) = \sum_{v \to u} a(u)$

a. Algorithm

In the HITS algorithm, following step are follow

Step-1- Sampling Step

Given a user query with several terms collect a set of pages that are very relevant called the base set.

How to find base set?

We retrieve all web pages that contain the query terms. The set of web pages is called the root set.

Next, find the link pages, which are either pages with a hyperlink to some page in the root set or some page in the root set has hyperlink to these pages.

All pages found form the base set.

Step 2 – Iteration Step

Goal: to find the base pages that are good hubs and good authorities. The algorithm performs a series of iterations, each consisting of two basic steps:

- Authority Update: Update each node's Authority score to be equal to the sum of the Hub Scores of each node that points to it. That is, a node is given a high authority score by being linked to by pages that are recognized as Hubs for information.
- Hub Update: Update each node's Hub Score to be equal to the sum of the Authority Scores of each node that it points to. That is, a node is given a high hub score by linking to nodes that are considered to be authorities on the subject.

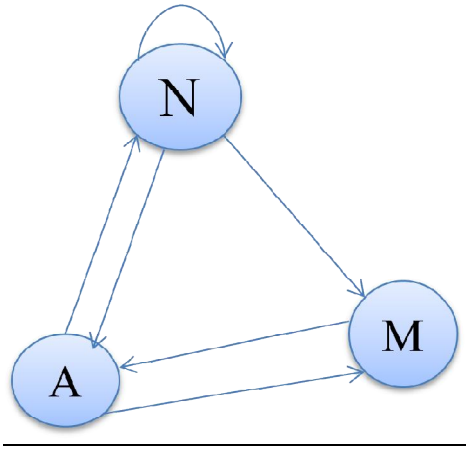

Fig.4 Example of HITS algorithm

N: Netscape

MS: Microsoft

A: Amazon.com

For hub:

h(N) = a(N) + a(MS) + a(A)

h(MS) = a(A)

h(A) = a(N) + a(MS)

$$\begin{pmatrix} h(N) \\ h(MS) \\ h(A) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} a(N) \\ a(MS) \\ a(A) \end{pmatrix}$$

$$\vec{h} = \begin{pmatrix} h(N) \\ h(MS) \\ h(A) \end{pmatrix} \quad \vec{a} = \begin{pmatrix} a(N) \\ a(MS) \\ a(A) \end{pmatrix}$$

$$\vec{h} = M\vec{a}$$

For authority:

a(N) = h(N) + h(A)

a(MS) = h(N) + h(A)

a(A) = h(N) + h(MS)

$$\begin{pmatrix} a(N) \\ a(MS) \\ a(A) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} h(N) \\ h(MS) \\ h(A) \end{pmatrix}$$

$$\vec{h} = M\vec{a}$$

We have,

$\vec{h} = M\vec{a}$, $\vec{a} = M^T\vec{h}$

We derive

$\vec{h} = MM^T\vec{h}$, $\vec{a} = M^TM\vec{a}$

b. Disadvantage of HITS:

Since there are two concepts, namely hubs and authorities, we do not know which concept is more important for ranking.

B. Page Rank algorithm:

Modern search engines employ methods of ranking the results to provide the "best" results first that are more elaborate than just plain text ranking. One of the most known and influential algorithms for computing the relevance of web pages is the Page Rank algorithm used by the Google search engine. It was invented by Larry Page and Sergey Brin while they were graduate students at Stanford.

PageRank is a link analysis algorithm, used by the Google search engine. It is a search engine calculating technology based on mutual hyperlinks between pages. Its principle is “The pages come from a number of quality links to pages, must be high-quality pages. . If we create a web page *i* and include a hyperlink to the web page *j*, this means that we consider *j* important and

relevant for our topic. If there are a lot of pages that link to *j*, this means that the common belief is that page *j* is important.[4] If on the other hand, *j* has only one backlink, but that comes from an authoritative site *k*, (like www.google.com, www.cnn.com, www.cornell.edu) we say that *k* transfers its authority to *j*; in other words, *k* asserts that *j* is important. Whether we talk about popularity or authority, we can iteratively assign a rank to each web page, based on the ranks of the pages that point to it.

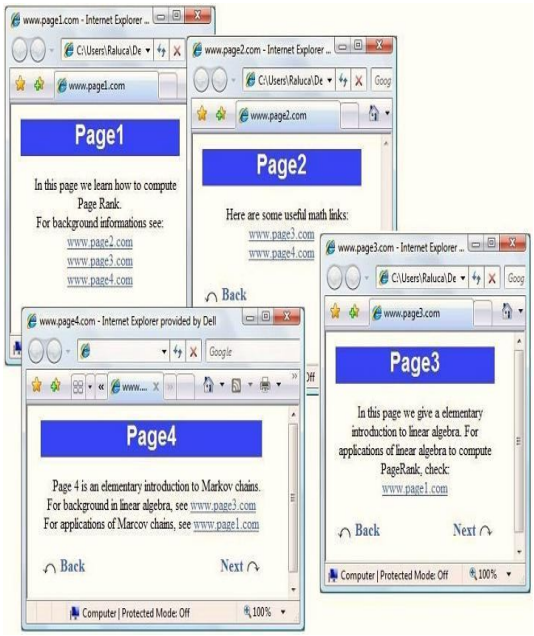

Fig.5 Example of page rank algorithm

We "translate" the picture into a directed graph with 4 nodes, one for each web site. When web site *i* references *j*, we add a directed edge between node *i* and node *j* in the graph. For the purpose of computing their page rank, we ignore any navigational links such as back, next buttons, as we only care about the connections between different web sites. For instance, Page1 links to all of the other pages, so node 1 in the graph will have outgoing edges to all of the other nodes. Page3 has only one link, to Page 1, therefore node 3 will have one outgoing

edge to node 1. After analyzing each web page, we get the following graph:

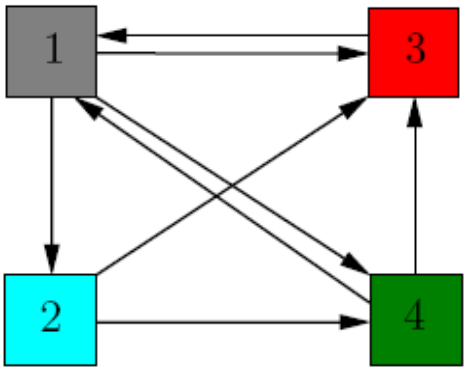

Fig.6 Directed graph of page rank example

In our model, each page should transfer evenly its importance to the pages that it links to. Node 1 has 3 outgoing edges, so it will pass on $\frac{1}{3}$ of its importance to each of the other 3 nodes. Node 3 has only one outgoing edge, so it will pass on all of its importance to node 1. In general, if a node has *k* outgoing edges, it will pass on $\frac{1}{k}$ of its importance to each of the nodes that it links to. Let us better visualize the process by assigning weights to each edge.

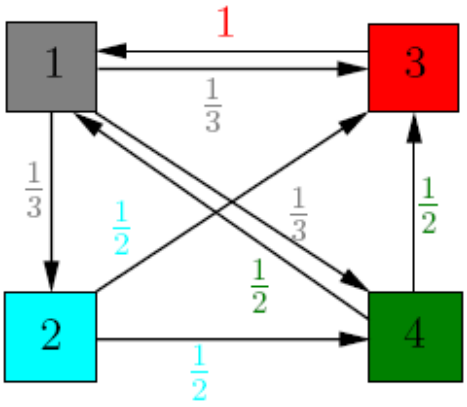

Fig 6. Directed graph with weights

Let us denote by *A* the transition matrix of the graph, A =

$$\begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}.$$

a. Dynamical systems point of view:

Suppose that initially the importance is uniformly distributed among the 4 nodes, each getting ¼. Denote by *v* the initial rank vector, having all entries equal to ¼. Each incoming link increases the importance of a web page, so at step 1, we update the rank of each page by adding to the current value the importance of the incoming links. This is the same as multiplying the matrix *A* with *v* . At step 1, the new importance vector is $v_1 = Av$. We can iterate the process, thus at step 2, the updated importance vector is $v_2 = A(Av) = A^2v$. Numeric computations give:

$$v = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}, \quad Av = \begin{pmatrix} 0.37 \\ 0.08 \\ 0.33 \\ 0.20 \end{pmatrix}, \quad A^2v = A(Av) = A\begin{pmatrix} 0.37 \\ 0.08 \\ 0.33 \\ 0.20 \end{pmatrix} = \begin{pmatrix} 0.43 \\ 0.12 \\ 0.27 \\ 0.16 \end{pmatrix}$$

$$A^3v = \begin{pmatrix} 0.35 \\ 0.14 \\ 0.29 \\ 0.20 \end{pmatrix}, \quad A^4v = \begin{pmatrix} 0.39 \\ 0.11 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad A^5v = \begin{pmatrix} 0.39 \\ 0.13 \\ 0.28 \\ 0.19 \end{pmatrix}$$

$$A^6v = \begin{pmatrix} 0.38 \\ 0.13 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad A^7v = \begin{pmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad A^8v = \begin{pmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{pmatrix}$$

We notice that the sequences of iterates *v*, *Av*, ..., A^kv tends to the equilibrium value $v^* = \begin{pmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{pmatrix}$.

We call this the PageRank vector of our web graph.

b. Advantage of PageRank:

PageRank involves only one concept for ranking

III. CONCLUSION:

In these paper, two searching algorithms are explained. HITS contain authority and hub webpages, while PageRank ranks pages just by authority. HITS is query dependent but PageRank is query-independent. HITS is applied to the local neighborhood of pages surrounding the results of a query whereas PageRank is applied to the entire web page.Finally, the HITS algorithm is time consuming since it first gets the root set for a query, then expands it and performs eigenvector computation. On the other hand, PageRank is an off-line process.

REFERENCES

1. D. Minnie, S. Srinivasan(2011),"Intelligent Search Engine Algorithms on Indexing and Searching of Text Documents using Text Representation.

2. Raymond Kosla, Hendrik Blockeel, *Web Mining Research: A Survey*, SIGKDD Explorations, July, 2000, Volume 2, Issue 1, pages 1-15.

3. N. Jeyaveeran, A. Haja Abdul Khader, R. Balasubramaniyan, *E-Learning and Web Mining: An Evaluation*, proceedings of the 2nd International

4. Qing Zhang, Zhiqiang Wei, Dongning Jia, Xiang Xiao(2012)," All-In-One Search Engine Algorithms of Smart environments-based on Android"

5. Margaret H Dunham & S. Sridar, *Data Mining*, Pearson Education,2007.

6. Gang Luo, *Design and Evaluation of the iMed Intelligent Medical Search Engine*, ICDE'09,Proceedings of the 2009 IEEE International Conference on Data Engineering, pp 1379 – 1390 (2009).

7. Hiroyuki Kawano(2001)," Overview of Mondou web search engine using text mining and information visualizing technologies.

8. http://www.math.cornell.edu/~mec/winter2009/RelucaRemus/lecture3\ lecture3.html