

WEB PAGE RANKING ALGORITHMS FOR TEXT-BASED INFORMATION RETRIEVAL

By

¹Abass, D. A., ²Folorunso, O and ³Yisau, N. B.^{1,2,3}Dept. of Computer Science,

Tai Solarin College of Education, Omu-Ijebu, Ogun State

²Dept. of Computer Science,

Federal University of Agriculture, Abeokuta, Ogun State

lolaabas@gmail.com, folorunso@gmail.com, yisautunde@yahoo.com**ABSTRACT:**

In this era of information volume explosion on the web, great number of web users use information retrieval (IR) tools like search engines (e.g. popular Google, Yahoo, Bing etc.) to find information from the world wide web (WWW) because of their crawling and ranking methodologies. Ranking mechanism is a very important factor for search engines as they daily answer millions of queries and subsequently download, index and store hundreds of millions of web pages. For a user query, ranking determines the sequence (arranged in descending order) of the return pages in result based on the popularity of the pages and the Web pages with high popularity come up in the returned results. Therefore, this paper attempts to review various ranking algorithms the search engines use to return relevant documents to the user's query. This review serves as a background for our further study.

Keywords: Information Retrieval, Ranking Algorithms, Search Engines, Query

INTRODUCTION

Information Retrieval (IR) is the science of searching for information within relational databases, documents, text, multimedia files, and the World Wide Web (Gupta et al, 2013). IR is becoming an important research area in the field of computer science, generally concerned with the searching and retrieving of knowledge-based information from large database (Sharma and Patel, 2013) and is considered as a subfield of computer science that deals with the representation, storage, and access of information (Mohameth-François, et. al., 2012). IR is the process by which a collection of data is represented, stored, and searched for the purpose of knowledge discovery in response to a user request, i.e. query (Alhenshiri, 2013) and it involves stages initiated by representing data and ending with returning relevant information to the user. Intermediate stage includes filtering, searching, matching and ranking operations

(Sharma and Patel, 2013). An ideal information retrieval system (IRS) is expected to retrieve only the relevant documents while irrelevant ones are ignored. The term "IR", as used in this paper, refers to the retrieval of *unstructured* records which consists of mainly free-form natural language text in which reasonable emphasis of IR research is focused on (Greengrass, 2000). IR is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections usually stored on computers (Manning et al, 2009). According to Ruthven and Lalmas (2003), IRS allow users to access large amounts of electronically stored information objects where a user submitting a request (query) to the system will receive, in return, a number of objects relating to his request.

Currently, great number of web users use IR tools like search engines (e.g. popular Google,

Yahoo, Bing etc.) to find information from the world wide web (WWW) because of their crawling and ranking methodologies. Ranking mechanism is a very important factor for search engines as they daily answer millions of queries and subsequently download, index and store hundreds of millions of web pages. For a user query, ranking determines the sequence (arranged in descending order) of the return pages in result based on the popularity of the pages and the web pages with high popularity come up in the returned results.

The remaining parts of this paper are organized as follows: Section 2 discusses the architecture for efficient IRS and the processes involved in retrieving relevant information. Section 3

deals with critical review of different algorithms commonly used by search engine to achieve meeting web user's need. Finally, Section 4 concludes the review of the algorithms and highlights the future research directions.

BASIC ARCHITECTURE OF INFORMATION RETRIEVAL SYSTEM

There are three basic processes an IRS has to support: the representation of the content of the documents, the representation of the user's information need, and the comparison of the two representations. The processes are visualized in Figure 1. In the figure, squared boxes represent data and rounded boxes represent processes.

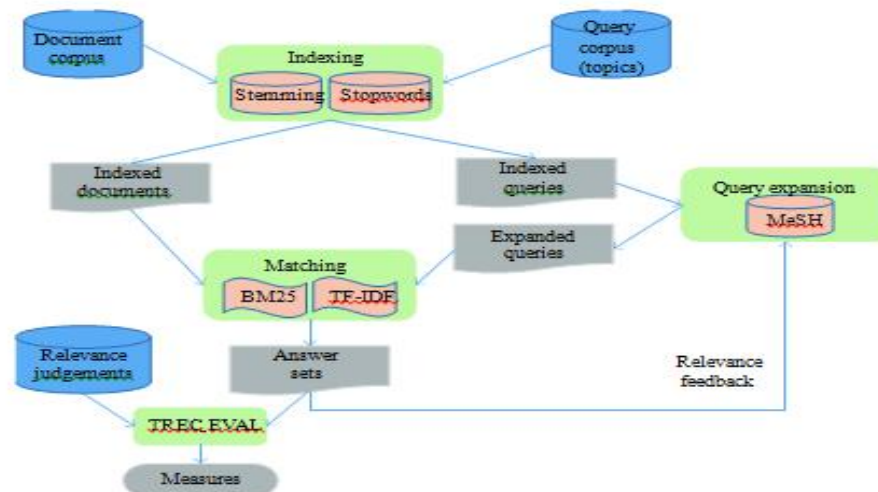


Figure 1: The Information Retrieval Process (Rivas et al, 2014)

Representing the documents is usually called the *indexing* process. The process takes place off-line, that is, the end user of the information retrieval system is not directly involved. The indexing process results in a representation of the document. Users do not search just for fun as they have a need for information. The process of representing their information need is often referred to as the *query* formulation process. The resulting

representation is the query (Hiemstra, 2009). Comparing the two representations is known as the *matching* process. Retrieval of documents is the result of this process.

A document collection consists of many documents containing information about various subjects or topics of interests. Document contents are transformed into a document representation (either manually or automatically). Document

representations are done in a way such that matching these with queries is easy. Another consideration in document representation is that such a representation should correctly reflect the author's intention. The primary concern in representation is how to select proper index terms. Typically, representation proceeds by extracting keywords that are considered as content identifiers and organizing them into a given format (Pathak, et. al, 2000).

RANKING ALGORITHMS

Ranking module plays an essential role in web search engine. It is used by search engine for ranking search results. With respect to user queries, the ranking module determines the order of the return pages in result. Generally, the order of the web pages depends on popularity of the pages where web pages having high popularity come up in the returned result. Popularity is also known as *Page Ranking*. Search engine's results for a query should be arranged on descending order of page ranking. Page ranking calculation is very critical part of search engine. For this purpose, search engine uses page ranking algorithms (Rawat et al, 2014). Basically, page ranking algorithms are used to present the search result by considering the relevance, importance and content score (Ashutosh and Ravi, 2009).

Singh and Ravi (2009) opine that with the increasing number of web pages and users on the web, the number of queries submitted to the search engines are also increasing rapidly. Therefore, the search engines need to be more efficient in its process. Web mining techniques are employed by the search engines to extract relevant documents from the web database and provide the necessary information to the users. The search engines become very successful and popular if they use efficient ranking mechanism.

Ranking algorithm was introduced to bring the concept of ranking. Since Boolean model in IR do not have ranking mechanism because it may skip important data. Hence, there was a need for ranking.

The result is ranked on the basis of occurrence of terms in the queries. This method eliminates the often-wrong Boolean syntax used by the end-users, and provides some results even though a term of the query is incorrect. This methodology also works well for the complex queries that may be difficult for users to express using Boolean operators. For example, "human factors and/or system performance in medical databases" is without any clear necessary Boolean syntax but the ranking model would do well with this query (Bhatia, et. al. 2013). Page ranking algorithms are used by the search engines to present the search results by considering the relevance, importance, score of content and techniques of web mining to order them according to the user interest. Some ranking algorithms depend only on the link structure of the documents while some use a combination of both, that is they use document content as well as the link structure to assign a rank value for a given document (Ashutosh and Ravi, 2009).

Page Rank-based Algorithms

Five Page Rank-based algorithms exist. These are *PageRank (PR)*, *Weighted PageRank (WPR)*, *Hypertext Induced Topic Search (HITS)*, *DistanceRank* and *DirichletRank* algorithms (Ashutosh and Ravi, 2009).

PageRank Algorithm

In 1996 at Stanford University, Larry Page and Sergey Brin developed a link analysis algorithm which is known as *PageRank* algorithm (Brin, and Page, 1998). PageRank algorithm was named after Larry Page, who is co-founder of Google search engine. Google is very popular because of its PageRank algorithm. PageRank is Google's method for calculating importance of web pages in web by using the hyperlinked structure of web pages. Suppose we have two pages A, B and page A links to Page B. It means that Page A is saying that Page B is important. If page A links to Page B, then we will say that Page A have forward link to

Page B and Page B have backlink from Page A. PageRank algorithm states that if a web page has some important backlinks to it, then its outgoing links to other

web pages also become important (Duhan et al, 2009). A simplified version of PageRank is shown in the equation below:

$$PR(A) = \left(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \dots + \frac{PR(T_n)}{C(T_n)} \right) \quad \text{Eq. 1}$$

Where $PR(A)$ is the PageRank of Page A, $PR(T_i)$ is the PageRank of a site having backlink to Page A, $C(T_i)$ is the number of forward links from that page and $PR(T_n)/C(T_n)$ is calculated for each page which have backlink of page A. PageRank formula is recursive. To calculate PageRank of a page, we need to know the PageRank of other pages. Therefore, the calculation starts with any random value (normally with 1 for all pages) of PageRank and iteratively updates these ranks using above formula and wait until they converge (Jain et al, 2013). But there is a

problem with this simplified ranking function. Suppose we have two pages that point to each other but there is no outgoing links from this loop and there are some pages which have incoming links to one of them, then during iteration this loop will accumulate rank but will never distribute any rank since there is no outgoing link. The loop forms a sort of trap which is known *rank sink* (Page et al, 1999). To overcome this problem a modification was made on PageRank formula. Modified PageRank formula is:

$$PR(A) = (1 - d) + d \left(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \dots + \frac{PR(T_n)}{C(T_n)} \right) \quad \text{Eq. 2}$$

Where d is damping factor and its value can be set between 0 and 1, we usually set the value of d to 0.85. The damping factor is use to stop the other pages having too much influence. Wang et al (2008) proved that PageRank has a zero-one gap flaw which can be potentially exploited by spammers to easily spam PageRank results. This zero-one gap problem occurs from the ad-hoc way of computing the transition probabilities in the random surfing model adopted in the current ranking. The probability that the random surfer clicks on one link is solely given by the number of links on that page. This is why one page's PageRank is not completely passed on to a page it links to, but is divided by the number of links on the page. So, the probability for the random surfer reaching one page is the sum of probabilities for the random surfer following links to this page. Now, this probability is reduced by the damping factor d . The justification within the Random Surfer Model is that the surfer does not click on an infinite number of links, but gets bored sometimes and jumps to another page

at random. The zero-one gap problem refers to the unreasonable dramatic difference between a page with no out-link and one with a single out-link in their probabilities of randomly jumping to any page. Alternatively, a novel DirichletRank algorithm based on the Bayesian estimation was proposed by Wang et al. (2008) with a Dirichlet prior to solve the zero-one gap problem specially the transition probabilities.

Weighted PageRank Algorithm

Xing and Ghorbani (2004) proposed a *Weighted PageRank (WPR)* algorithm which is an extension of the *PageRank* algorithm. This algorithm assigns a larger rank values to the more important pages rather than dividing the rank value of a page evenly among its outgoing linked pages. Each outgoing link gets a value proportional to its importance. The importance is assigned in terms of weight values to the incoming and outgoing links and are denoted as $W^m(m, n)$ and $W^{out}(m, n)$ respectively. $W^m(m, n)$ as

shown in (a) is the weight of $link(m, n)$ calculated based on the number of incoming links of page n and

$$W_{(m,n)}^{in} = \frac{I_n}{\sum_{p \in R(m)} I_p} \tag{Eq. 3(a)}$$

$$W_{(m,n)}^{out} = \frac{O_n}{\sum_{p \in R(m)} O_p} \tag{Eq. 3(b)}$$

Where I_n and I_p are the number of incoming links of page n and page p respectively. $R(m)$ denotes the reference page list of page m . $W^{out}(m, n)$ is as shown in (b) is the weight of $link(m, n)$ calculated based on the number of outgoing links of page n and the number of outgoing links of all reference pages of m . Where O_n and O_p are the number of outgoing links of page n and p respectively.

PageRank versus Weighted PageRank Algorithms

Choudhary and Burdak (2012) opine that we can easily differentiate the WPR from the PageRank, categorized the resultant pages of a query into four categories based on their relevancy to the given query. They are:

- i. Very Relevant Pages (VR): The pages contain very important information related to a given query.
- ii. Relevant Pages (R): The pages do have important information about given query.
- iii. Weak Relevant Pages (WR): The pages do not have the relevant information but may have the

the number of incoming links of all reference pages of page m .

- query keywords.
 - iv. Irrelevant Pages (IR): The pages do not have both relevant information and query keywords.
- Both the PageRank and WPR algorithms provide pages in the sorting order according to their ranks to users for a given query. So, the order of relevant pages and their numbering are very important for users in the resultant list.

The HITS Algorithm - Hubs and Authorities

Kleinberg (1999) identifies two different forms of web pages called *hubs* and *authorities*. Authorities are pages having important contents. Hubs are pages that act as resource lists, guiding users to authorities. Thus, a good hub page for a subject point to many authoritative pages on that content, and a good authority page is pointed by many good hub pages on the same subject. Kleinberg says that a page may be a good hub and a good authority at the same time. This circular relationship leads to the definition of an iterative algorithm called HITS (Hyperlink Induced Topic Search).

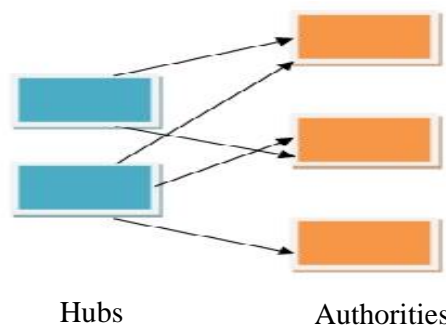


Figure 2: Illustration of hubs and authorities (Choudhary and Burdak, 2012)

According to Choudhary and Burdak (2012), in HITS algorithm, the hub and authority are calculated using the following algorithm:

HITS Algorithm

1. Initialize all weights to 1
2. Repeat until the weights converge:
3. For every hub $p \in H$
4. $H_p = \sum_{q \in I(p)} H_q$
5. For every authority $p \in A$
6. $A_p = \sum_{q \in B(p)} H_q$
7. Normalize

The HITS algorithm treats WWW as a directed graph $G(V, E)$, where V is a set of vertices representing pages and E is a set of edges that correspond to links. There are two major steps in the HITS algorithm. The first step is the *Sampling* step and the second step is the *Iterative* step. In the sampling step, a set of relevant pages for the given query are

collected i.e. a sub-graph S of G is retrieved which is high in authority pages. This algorithm starts with a root set R , a set of S is obtained, keeping in mind that S is relatively small, rich in relevant pages about the query and contains most of the good authorities. The second step, iterative step, finds hubs and authorities using the output of the sampling step using (a) and (b).

$$H_p = \sum_{q \in I(p)} A_q \quad Eq. 4(a)$$

$$A_p = \sum_{q \in B(p)} H_q \quad Eq. 4(b)$$

where H_p is the hub weight, A_p is the Authority weight, $I(p)$ and $B(p)$ denotes the set of reference and referrer pages of page p . The page's authority weight is proportional to the sum of the hub weights of pages that it links to it, (Kleinberg, 1999).

The following are the constraints of HITS algorithm (Chakrabarti, et. al., 1999):

- i. *Hubs and authorities*: It is not easy to distinguish between hubs and authorities because many sites are hubs as well as authorities.
- ii. *Topic drift*: Sometime HITS may not produce the most relevant documents to the user queries because of equivalent weights.

- iii. *Automatically generated links*: HITS gives equal importance for automatically generated links which may not produce relevant topics for the user query.
- iv. *Efficiency*: HITS algorithm is not efficient in real time.

Distance Rank Algorithm

DistanceRank algorithm is proposed by Ali Mohammad Zareh Bidoki and Nasser Yazdani (Bidoki and Yazdani, 2008). They proposed a novel recursive method based on reinforcement learning (Sutton and Barto, 1998) which considers distance between pages as punishment, called "DistanceRank" to compute ranks of web pages. The number of 'average clicks'

between two pages is defined as *distance*. The main objective of this algorithm is to minimize distance or punishment so that a page with smaller distance to have a higher rank. Most of the current ranking algorithms have the "rich-get richer" problem (Bidoki and Yazdani, 2008) i.e. the popular high rank web pages become more and more popular and the young high quality pages are not picked by the ranking algorithms. There are many solutions (Sutton and Barto, 1998; Bidoki and Yazdani, 2008) suggested for this "rich-get-richer" problem. The DistanceRank algorithm is less sensitive to the "rich-get-richer" problem and finds important pages faster than others. This algorithm is based on the reinforcement learning such that the distance between pages is treated as punishment factor. Normally related pages are linked to each other so the distance based solution can find pages with high qualities more quickly.

In PageRank algorithm, the rank of each page is defined as the weighted sum of ranks of all pages having back links or incoming links to the page. Then, a page has a high rank if it has more back links to the pages that have higher ranks. These two properties

$$d_b = \frac{\sum_{a=1}^v d_{ab}}{v}$$

The purpose of the *Distance Rank* is to compute the average distance of each page and there is a dependency between the distance of each page and its incoming links or back links. For example, if

$$d_b = d_a \log(O(a)) \tag{Eq. 6}$$

In general, suppose $O(a)$ denotes the number of forwarding or outgoing links from page a and $B(b)$ denotes the set of pages pointing to page b . The

$$d_b = \min(d_a + \log O(a)), a \in B(b) \tag{Eq. 7}$$

Ashutosh and Ravi (2009) opine that *DistanceRank* algorithm is less sensitive to "rich-get-richer" problem. *DistanceRank* algorithm also provides good prediction of pages for future ranking. The convergence speed of this algorithm is fast with a

are true for Distance Rank also. A page having many incoming links should have low distance and if pages pointing to this page have low distance then this page should have a low distance. The above point is clarified using the following definitions.

Definition 1. If page a points to page b , then the weight of link between a and b is equal to $\log(O(a))$ where $O(a)$ shows a 's out degree or outgoing links.

Definition 2. The distance between two pages, a and b , is the weight of the shortest path (the path with the minimum value) from a to b . This is called *logarithmic distance* and is denoted as d_{ab} .

Definition 3. If d_{ab} shows the distance between two pages a and b as in Definition 2, then d_b denotes the average distance of page b and is defined as the following where V shows number of web pages:

Eq. 5

page b has only one backlink and it is from page a , to compute the average distance for page b , d_b is as follows.

DistanceRank of page b denoted by d_b is given as follows.

little number of iterations. In *DistanceRank*, it is not necessary to change the web graph for computation. Therefore, some parameters like damping factor can be removed and can work on the real graph.

DirichletRank Algorithm

DirichletRank algorithm is proposed by Xuanhui Wang et al in 2008. This algorithm eliminates the zero-one gap problem found in the *PageRank* algorithm proposed by Brin and Page (Page et al, 1999). The zero-one gap problem occurs due to the current ad-hoc way of computing transition probabilities in the random surfing model. The zero-one-gap problem allows spammers to manipulate the ranking results of PageRank. That is, zero-one gap problem can be exploited to spam *PageRank* results and make the state-of-art link-based anti-spamming

techniques ineffective. The authors proposed a novel *DirichletRank* algorithm which calculates the probabilities using the Bayesian estimation of Dirichlet prior. *DirichletRank* is a form of *PageRank* and the authors have shown that *DirichletRank* algorithm is free from the zero-one gap problem. They have also proved that this algorithm is more robust against several common link spams and is more stable under link perturbations. The authors also claim that this is as efficient as *PageRank* and it is scalable to large-scale web applications.

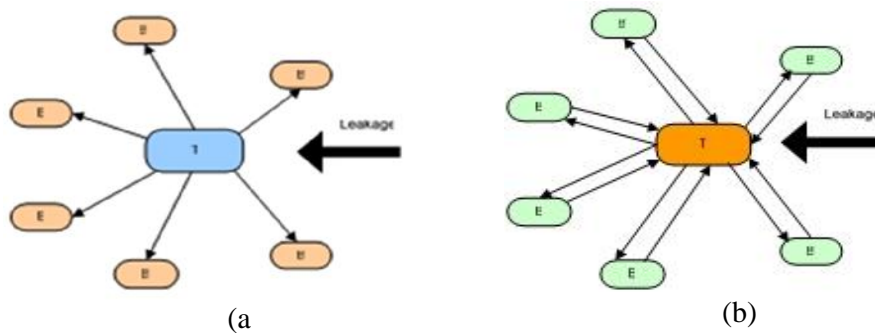


Figure 3: Sample Contrast Structures (Ashutosh and Ravi, 2009)

Considering the Figure 3 below where (a) is a structure without link spamming (only out-link) and (b) shows a typical spamming structure with all bogus pages B's having back links to the target page T. In *DirichletRank*, a surfer would more likely follow the outlinks of the current page if the page has many out-

links. Bayesian estimation provides a proper way for setting the transition probabilities and the authors showed that it not only solves the zero-out-link problem but also solves the zero-one-gap problem. The random jumping probability of *DirichletRank* is:

$$\omega(n) = \frac{\mu}{n + \mu}, 0 \leq n \leq \infty \tag{Eq. 8}$$

where n is the number of out-links and μ is the Dirichlet parameter. DirichletRank scores $d_o(\cdot)$ and $d_s(\cdot)$ for the structures shown in Fig. 3 (a) and (b) can be calculated using the following formula.

$$d_o(T) = \sigma + \frac{\tau}{N} \tag{Eq. 9}$$

$$d_s(T) = \left[1 + \frac{k}{\mu^2 + (k+1)\mu} \right] \left[\sigma + \frac{k + \mu + 1}{\mu + 1} \frac{\tau}{N} \right] \tag{Eq. 10}$$

and $d_s(T) \leq d_o(T)$ for any positive integer k.

Ashutosh and Ravi (2009) claimed that they obtained a similar score of PageRank i.e. $d_s(T)$ is constantly larger than or equal to $d_o(T)$ but $d_s(T)$ is in fact close to $d_o(T)$. It also shows that no significant change in T 's DirichletRank scores before and after spamming. Hence, DirichletRank is more stable and less sensitive to the change of local structure. DirichletRank also do not take extra time cost and this makes it suitable for web-based applications. The authors also proved that DirichletRank is more stable than PageRank during link perturbation i.e. removing a small number of links or pages. Stability is an important factor for a reliable ranking algorithm. Their experiment results showed that DirichletRank is more effective than PageRank due to its more reasonable allocation of transition probabilities.

CONCLUSION AND FUTURE WORK

The work purposely review different ranking algorithms used for text-based information retrieval on the Web. A typical search engine should use web page ranking techniques based on the specific needs of the users because the ranking algorithms provide a definite rank to resultant web pages. The future work will further describe a large-scale evaluation of the effectiveness of DirichletRank algorithms in web-based information retrieval in comparison with other popular link-based ranking algorithms. Emphasis will also be laid on determining their effectiveness using three common performance measures: the mean reciprocal rank, the mean average precision, and the normalized discounted cumulative gain measurements using two major criteria for evaluating IRS performance.

REFERENCES

Alhenshiri Anwar A. (2013). Web Information Retrieval and Search Engines Techniques. *Al-Satit Journal*, PP: 55-92.

Ashutosh Kumar Singh, Ravi Kumar P. 2009. "A Comparative Study of Page Ranking

Algorithm for Information Retrieval". *World Academy of Science, Engineering and Technology* Vol:3, 4-20.

- Bhatia Parul Kaira, Mathur Tanya and Gupta Tanaya. 2013. Survey Paper on Information Retrieval Algorithms and Personalized Information Retrieval Concept. *International Journal of Computer Applications* (0975 - 8887) Volume 66, No. 6.
- Bidoki A. M. Z, and Yazdani N. (2008). DistanceRank: An intelligent ranking algorithm for web pags. *Information Processing Management*, Vol. 44, No. 2, pp. 877-892.
- Brin, S. and Pagem L. (1998). Anatomy of of a large-scale hypertextual web search engine. Proc. 7th Intl. World-Wide-Web Conference, pp. 107-117.
- Chakrabarti Soumen, Martin van den Berg, and Byron Dom (1999). Focused crawling: a new approach to topic-specific web resource discovery. *Computer Networks*, 31:1623{1640}
- Choudhary Laxmi and Burdak Bhawani Shankar (2012). Role of Ranking Algorithms for Information Retrieval. *International Journal of Artificial intelligence & Applicatins (IJAAI)*, Vol.3, No.4, pp203-220.
- Duhan Neelam, A. K. Sharma, Komal Kumar Bhatia. 2009. Page Ranking Algorithms: A Survey. *IEEE International Advance Computing Conference*, 978-1-4244-1888-6/08.
- Greengrass, E. D. (2000). Information Retrieval: A Survey. CSCI 561.
- Gupta Yogesh, Saini Ashish and Saxena, A.K. (2013). A survey on important aspects of information retrieval. *Journal of Engineering and Technology*, Vol. 4 No. 2, pp. 49-72
- Hiemstra Djoerd (2009). Information Retrieval Models. Published in Goker, A., and Davies, J. Information Retrieval: Searching in the 21st Century. John Wiley and Sons, Ltd., ISBN-13: 978-0470027622, pp. 1-2.
- Jain Ashish, Sharma Rajeev, Dixit Gireesh, Tomar Varsha. 2013. Page Ranking Algorithms in Web

- Mining, Limitations of Existing methods and a New Method for Indexing Web Pages. *CSW: IEEE*.
- Kleinberg, Joh M.. Authoritative sources in a hyperlinked environment. In *Journal of the ACM*, pp. 604-632
- Manning Christopher D., Raghavan Prabhakar & Schütze Hinrich.(2009). *An Introduction to Information Retrieval*. Cambridge University Press Cambridge, England. Ppl-3.
- Mohameth-François Sy, Sylvie Ranwez, Jacky Montmain, Armelle Regnault, Michel Crampes, Vincent Ranwez Pezzoli. 2012. User-centered and ontology based information Retrieval system for life sciences, *BMC Bioinformatics*, 1471-2105.
- Page L., Brin S., Motwani R. and Winograd T. (1999). The PageRank Citation: Bringing order to the web., Technical Report, Stanford Digital Libraried SIDL-WP-1999-0120.
- Pathak Praveen, Gordon Michael and Fan Weiguo. 2000. Effective Information Retrieval using Genetic Algorithm-based Matching Functions Adaptation. *Proceedings of the 33^d Hawaii International Conference on System Sciences*.
- Rawat Poonam, Dwivedi Shri Prakash, Mandoria Haridwari Lal. 2014. An Adaptive Approach in Web Search Algorithm. *International Journal of Information & Computation Technology*. Vol. 4, No. 15, pp. 1575-1581
- Rivas, A. R., Iglesias, E. L., and Borrajo, L. (2014). Study of Query Expansion Techniques and Their Application in the Biomedical Information Retrieval, *The Scientific World Journal*. Volume 2014, Article ID 132158 10 pages
<http://dx.doi.org/10.1155/2014/132158>
- Ruthven, I. and Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *Knowl. Engin. Rev.* 18, 2, 95-145.
- Sharma Manish and Patel Rahul. 2013, A Survey on Information Retrieval Models, Techniques and Applications. *International Journal of Emerging Technology and Advanced Engineering*, volume 3, Issue 11, pp:542-545.
- Singh, Ashutosh Kumar, Ravi, Kumar P. (2009). A Comparative Study of Page Ranking Algorithms for Information Retrieval, *International Journal of Electrical and Computer Engineering* 4:7 2009.
- Sutton R. S. and Barto A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.
- Wang X., Tao T., Sun J. T., Shakery A., and Zhai C. (2008). Dirichlet Rank: Solving the Zero-One Gap Problem of PageRank. *ACM Transaction on Information Systems*, Vol. 26, Issue 2.
- Xing W. and Ali Ghorbani, 2004. "Weighted PageRank Algorithm", *Proc. Of the Second Annual Conference on Communication Networks and Services Research (CNSR 04)*, IEEE.