A New Context Driven Page Ranking Algorithm

Aditi Humad
Department of Computer Science and Engineering and IT,
Mangalayatan University,Aligarh(U.P)

Vikas Solanki
Department of Computer Science and Engineering and IT,
Mangalayatan University,Aligarh(U.P)

Abstract- The size of web is growing day by day. People rely on search engines to find the information on the web. Now it is a challenge for the search engine to provide quality and relevant information to the information seekers. In search engines, the quality and relevancy of the results depends upon the performance of page ranking algorithms. In this paper a new algorithm for page ranking has been proposed. The proposed algorithm is ranking the searched results on the basis of the context of the user query. The algorithm works in two steps, step 1 is to find the context of user query with the help of thesaurus. In step-2 the context of different documents is also determined with the help of thesaurus. Then only those documents which are in the context of user query are ranked and returned for the query result. The number of documents ranked by the ranking algorithm will get reduced by matching the context of the query and context of the documents. This decrease in no of documents ranked will improve the precision and recall values of the search results.

Keywords - context, information retrieval, page ranking, search engine, thesaurus.

I. INTRODUCTION

The web as we all know is the largest source of data. During the past few years the World Wide Web has become the foremost and most popular way of communication and information dissemination. It serves as a platform for exchanging various kinds of information, ranging from research papers, and educational content, to multimedia content, software and personal logs. Every day, the web grows by roughly a million electronic pages, adding to the hundreds of millions pages already on-line. So with the rapid growth of information sources available on the World Wide Web, it has become increasingly necessary for users to use automated tools to find the desired information resources, and to track and analyze their usage patterns. These factors give rise to the requisite of creating server side and client side intelligent systems that can effectively mine for knowledge. Web mining can be broadly defined as the extraction and mining of useful information from the World Wide Web.

Web Structure Mining is the process of discovering information from the Web, finding information about the web pages and inference on hyperlink, finding authoritative web pages, retrieving information about the relevance and the quality of the web page. Thus Web structure mining focuses on the hyperlink structure of the web. We review two approaches: HITS concept and Page Rank method. Both approaches focus on the link structure of the Web to find the importance of the Web pages. Mainly In links to the pages and out links from the page can give idea about the context of the page. PageRank does not rank web sites as a whole, but it calculates the rank of individual web page and Hypertext Induced Topic Search (HITS) depends upon the hubs and authority framework.

A web search engine typically consists of:

1) Crawler: used for retrieving the web pages and web contents
2) Indexer: stores and indexes information on the retrieved pages
3) Ranker: Measure the importance of Web Pages returned
4) Retrieval Engine: performs lookups on index tables against query

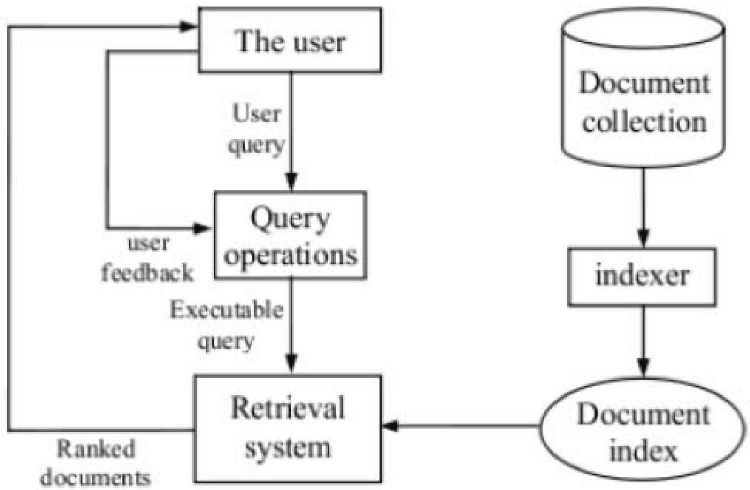

Figure-1. A search engine system during a search operation

A user issues a query which is first checked before it is forwarded and compared to documents indexes.

Nowadays searching on the internet is most widely used operation on the World Wide Web. The amount of information is increasing day by day rapidly that creates the challenge for information retrieval. There are so many tools to perform efficient searching. Due to the size of web and requirements of users creates the challenge for search engine page ranking [19]. Ranking is the main part of any information retrieval system Today's search engines may return million of pages for a certain query It is not possible for a user to preview all the returned results So, page ranking is helpful in web searching. Rankers are classified into two groups: - Content-based rankers and Connectivity-based rankers. Content-based rankers works on the basis of number of matched terms, frequency of terms, location of terms, etc. Connectivity-based rankers work on the basis of link analysis technique, links are edges that point to different web pages.

II. PROPOSED ALGORITHM

In this paper a new architecture has been proposed to find the ranks of the documents. The ranking system has been divided into two components pre rank module and post rank module. The proposed architecture has been given in figure-2. The components are as follows:

1. Crawler : This module is responsible to download the pages from the web and storing the documents in a repository.

2. Context Finder :- This module find the context of a document using thesaurus.

3. Thesaurus :- It is a module that will decide the context of a document. It may generate a list of contexts to which the document is related above a threshold value.

4. Indexer :- This module will parse the documents and creates an index to facilitate the ranking module to find the relevant documents for a user query.

5. Context Based Index :- The indexer module creates a context based index. The context based index stores all the available contexts and a list of documents for every context.

6. Context Repository :- It is a repository which stores the knowledge about a particular domain or context. This repository can also be created with the help of thesaurus. It stores all available context and the list of words which are related to that context. A sample of context repository is shown in Table-1.

Table-1 Am example of a context repository

Sr No	Available Context	List of words related to that context
1.	Car and Automobiles	auto, car, automobiles, motor, transport, bus, motorcycle, engine, gear, wheel, travel, vehicles, petrol, diesel
2.	Medicine	injury, biomedical, doctor , patient, diagnosis, treatment, medicine, hospital, health , healthcare, drug , nurse , pharmacists , radiographers , disease , blood , sick
3.	Space Science	space , science, stars, universe, gravity, supernovae, planetary, planet, hydrogen, helium, earth, moon, astronomy, solar, milky, galaxy , aeronautics, aerospace, , satellite, rocket, aircraft
4.	Weapon	weapon, crime, missiles, ballistic, military, attacks, guns, rockets, tank, war, bomb, atom, aircraft, bullet

7. Pre Rank Module :- This module will calculate the pre importance weight of a document. This weight is a measure how good a document is in a given context.

8. Post Rank Module:- This module will calculate the post importance weight i.e. calculate weight after user query. The post importance weight is a measure of how much a document is related to user query.

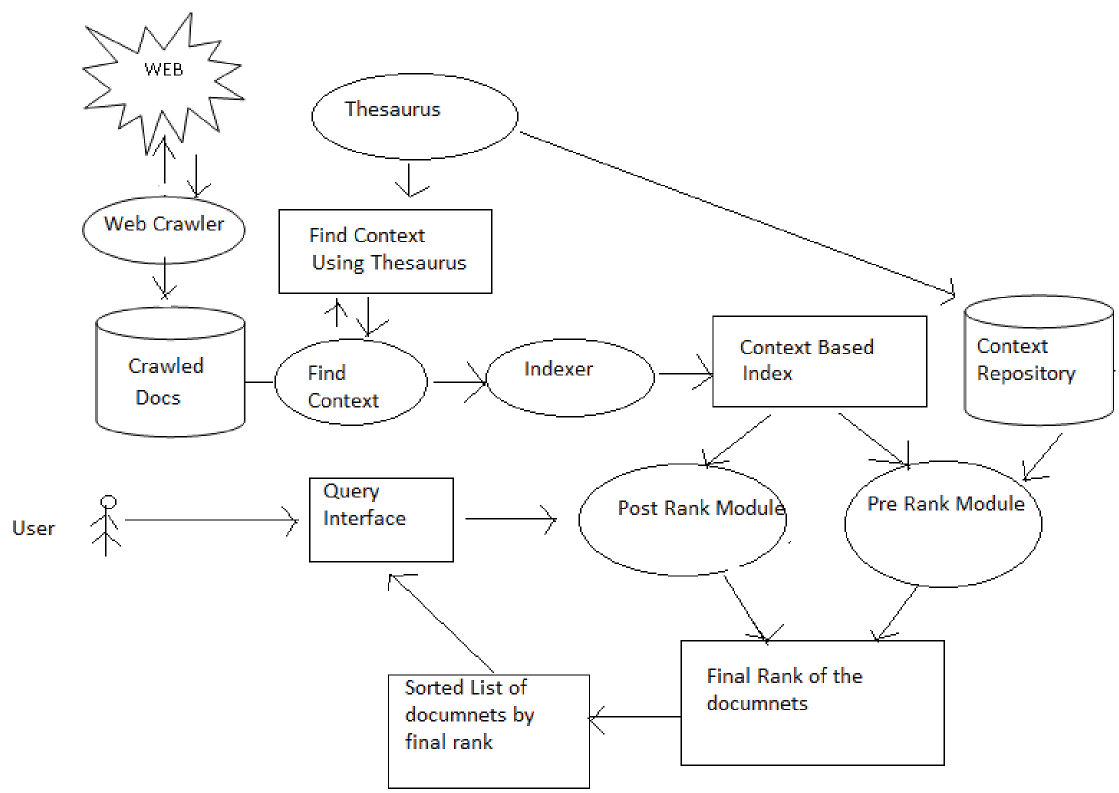

Figure-2. Proposed architecture of context based ranking system

In this paper a new algorithm for page ranking has been proposed. The algorithm mainly based on calculation two things namely pre-importance-weight and post-importance-weight. The pre importance weight is a measure to calculate importance of document in a given context. The post importance weight is a measure of to calculate the importance of a document for the user query. For the calculation of pre importance weight the algorithm needs a list of words which are related to the given domain or context. This list can be found using the thesaurus and WorldNet. The proposed algorithm for calculating the pre-importance-weight is as follows:

A. Algorithm Calculate-pre-importance-weight

Input : Topic, List of related words, Document

Output : pre-importance-weight

1. Collect a list files called corpus.
2. Initialize a context repository mentioned above.
3. For all files in the corpus find the context of the file from thesaurus.
4. Parse all documents which are in the corpus to find a list of words with their frequency.
5. Find the list of words for this context in context repository.
6. Find common words which are in the parsed file and which also exist in the list of words in the context repository for the context of given file. Say it is a list of common words.
7. Calculate the document-importance-weight of the document in the given context by using the following formula

   Count-1 = sum of the frequencies of all words in the common words list of step 6.
   Count-2 = no of words in the common words list of step 6.
   document-importance-weight = (count-1)/(Count-2)

B. The algorithm to calculate post-importance-weight is as follows:

Input : Query, Documents

Output : post-importance-weight

1. Read a query from the user.
2. Read/Find the context of the query.
3. Parse the query to find keywords to be searched.
4. For all documents which are related to the context of the query

```
{
    Count-1 = Sum of frequencies of common words in user query and the document.
    Count-2 = (count1/No of common words in user query and the document)
    Post-importance-weight = Count-1/Count-2
}
```

C. The algorithm to calculate the final rank of the documents is as follows:

Input : Documents, pre-importance-weight, post-importance-weight

Output : Ranked Documents

1. For all documents which are related to the context of the query

```
{
    Weight-1 = pre-importance-weight
    Weight-2 = post-importance-weight
    Final-Weight = Weight-1 + Weight-2
}
```

2. Sort all the documents (which are in the context of user query) on Final-Weight value
3. Return the sorted list to the user.

III. Experimental Result

The proposed method has been implemented in java and a snapshot is shown in figure-3. It has been implemented for a sample corpus of 16 text files. The files belong to different contexts. In the experiment 3 files belongs to context “car and automobiles”, 5 files belongs to context “Medicines”, 4 files belongs to context “Weapon” and 4 files belong to context “Space Science”. These files has been parsed to retrieve tokens. Stop words has been removed and stemming has been performed on each token. The document pre important weight has been calculated and assigned to each document. Then a query has been fired. The user has to specify the query and the context of the query. A list of available contexts has been provided to the user.

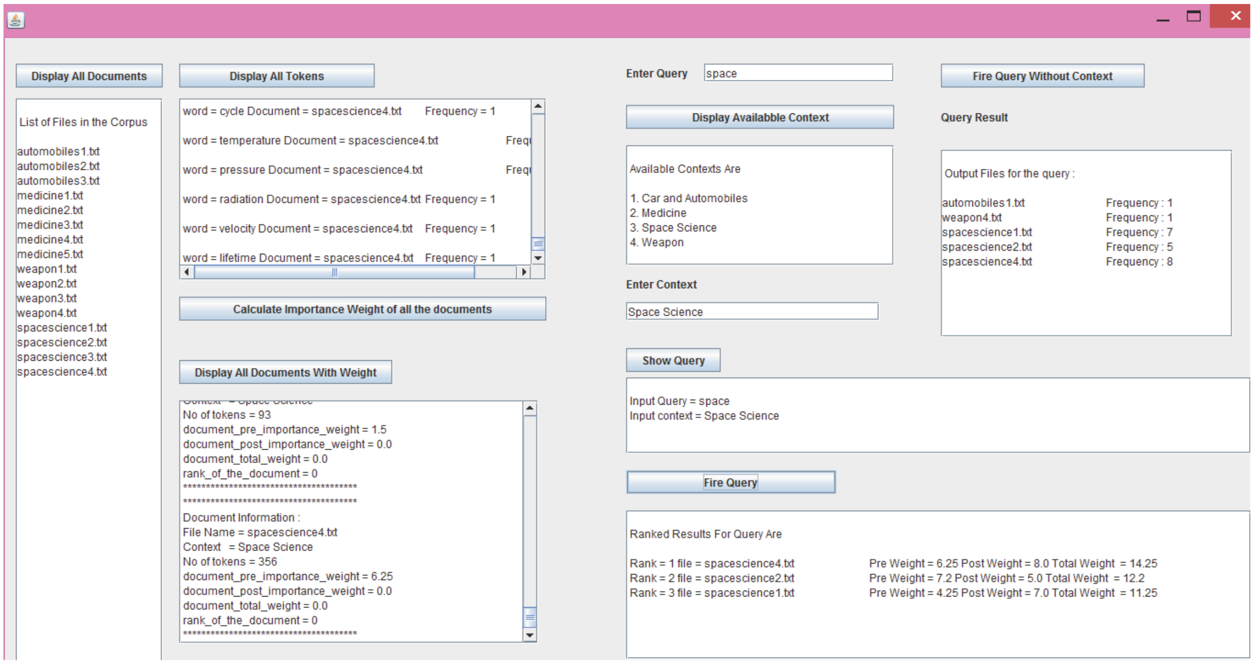

Figure-3. Experimental results

The cluttering of documents in different context is not the are of research of this paper. It has been assumed that a context generator module will assign context sense to each document and this module cluster the documents according to their context. The query has been fired and result has been evaluated as follows:

Query : space
Context : Space Science
Result : Three files has been retrieved which belongs to context “Space Science” sorted according to their weight.

When the same query has been fired on the whole index without entertaining the context, five files has been retrieved. One document from the automobile context, one from Weapon context and three from space science context.

Comparative Analysis
The results has been compared for precision value for both context based searching and normal searching.
Precision is the fraction of the documents retrieved that are relevant to the user's information need.

Precision = (|{Relevant Documents}∩{Retrieved Documents}|) / |{Retrieved Documents}|
Without context:
Relevant Documents = 3
Retrieved Documents = 5
(|{Relevant Documents}∩{Retrieved Documents}| = 3
Precision = 3/5 = 0.6

With Context
Relevant Documents = 3
Retrieved Documents = 3
Precision = 3/3 = 1.0

So almost all the documents retrieved will be relevant to user in context based retrieval.

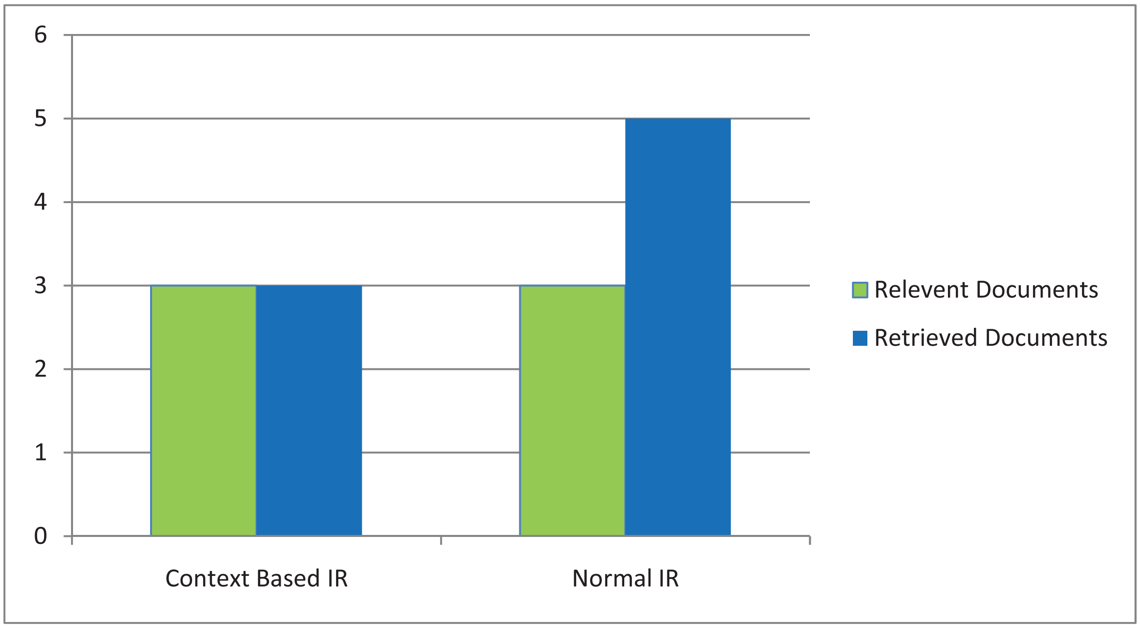

Figure-4. Comparative Analysis of Context Based IR and Normal IR.

IV.CONCLUSION

In this paper a new algorithm for page ranking has been proposed. The aim of this research is to improve the precision and recall value of the page ranking system. The paper divided the page repository on the basis of the context. Documents which are in the context of user query have been ranked by the ranking system. The pre importance weight is a measure to calculate importance of document in a given context and the post importance weight is a measure to calculate the importance of a document for the user query has been calculated. Finally a procedure will combine these two weights to generate the rank of the documents. Because only those documents which are in the context of the user query has been selected for the ranking, so it will returned all the related documents to the user. It will improve the precision and recall measures of the page ranking system. In this paper a new algorithm has been proposed for page/document ranking. However in future efforts are needed to implement the given algorithms and to test the efficiency of the proposed work on a corpus having large number of documents with different contexts.

REFERENCES

[1] BHARAT, K., AND HENZINGER, M. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In Proceedings of ACM SIGIR'98 (Melbourne, Australia, 1998).
[2] A.K. Singh, Ravi Kumar and Alex Goh Kwang Leng "Efficient Algorithm for Handling Dangling Pages using Hypothetical node".
[3] Wenpu Xing and Ali Ghorbani, "Weighted PageRank Algorithm" Proceedings of the Second Annual Conference on Communication Networks and Services Research (CNSR'04) 2004 IEEE.
[4] S. Al-Saffar and G. Heileman, Experimental bounds on the usefulness of personalized and topic-sensitive pagerank, International Conference on Web Intelligence, pp. 671-675, 2007.
[5] A. Rungsawang, K. Puntumapon, B. Manaskasemsak, Un-biasing the link farm effect in pagerank computation, 21th International Conference on Advanced Networking and Applications, pp. 924-931, 2007.
[6] Cooper, C. Frieze A., "A general model of Web graphs", In ESA, 2001, pp. 500-511. CERN Common Log Format,
[7] http://www.w3.org/Daemon/User/Config/Logging.html# common-log file-format.
[8] CALADO, P., RIBEIRO-NETO, B., ZIVIANI, N., MOURA, E., AND SILVA, I. Local Versus Global Link Information in the Web. ACM Transactions on Information Systems 21, 1 (January 2003), 42–63.
[9] CRASWELL, N., CRIMMINS, F., HAWKING, D., AND MOFFAT, A. Performance and cost tradeoffs in web search. In ADC'04 (Dunedin, New Zealand, January 2004), pp. 161–170. http://es.csiro.au/pubs/craswell adc04.pdf.
[10] A. L. Barabasi and R. Albert, Emergence of scaling in random networks, Science Magazine, Vol. 286. no. 5439, pp. 509-512, 199
[11] Kleinberg, J. M. Authoritative sources in a hyperlinked environment, Journal of the ACM, Vol.46 (5). (Sept. 1999). 604-632.

[12] Lerman, K., Getoor, L., Minton, S., and Knoblock, C.Using the Structure of Web Sites for Automatic Segmentation of Tables. SIGMOD (2004) 119-130.
[13] Eiron, N. McCurley, K., and Tomlin, J. Ranking the web frontier. Proceedings of the international conference on World Wide Web, (WWW'04). Pp.309-318, 2004.
[14] C. Guo and Z. Liang, An improved BA model based on the pagerank algorithm, 4th WiCOM International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1-4, 2008.
[15] NG, A. Y., ZHENG, A. X., AND JORDAN, M. I. Link analysis, eigenvectors, and stability. In Proceedings of IJCAI'01 (Seattle, USA, 2001), ACM Press.
[16] Deng Cai, Shipeng Yu, and et al. Block-based Web Search. In Proceedings of ACM SIGIR'04, July 25-29, 2004, Sheffield, South Yorkshire, UK. 465-463.
[17] HAVELIWALA, T. H. Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search. In IEEE Transactions on Knowledge and Data Engineering (July 2003).
[18] Ziv Bar-Yossef and Sridhar Rajagopalan. Template Detection via Data Mining and its Applications. In: Proceedings of WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA. 580-591.
[19] L. Page, S. Brin, R. Motwani, and T. Winograd. The Pagerank citation ranking: Bring order to the web. Technical report, Stanford University, 1998.
[20] B. Liu. Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data. Springer, 2006.
[21] L. Getoor, Link Mining: A New Data Mining Challenge. SIGKDD Explorations, vol. 4, issue 2, 2003.
[22] C. Ding, X. He, P. Husbands, H. Zha, and H. Simon, Link analysis: Hubs and authorities on the world. Technical report: 47847, 2001
[23] J. Wang, Z. Chen, L. Tao, W. Ma, and W. Liu, Ranking user's relevance to a topic through link analysis on web logs, Proceedings of the Second Annual Conference on Communication Networks and Services research (CNSR'04), 2002, pp. 49–54.
[24] Xianchao Zhang, Hong Yu, Cong Zhang, and Xinyue Liu "An Improved Weighted HITS Algorithm Based on Similarity and Popularity", 2007 IEEE
[25] Sekhar Babu Boddu, V.P Krishna Anne, Rajesekhara Rao Kurra, Durgesh Kumar Mishra, " Knowledge Discovery and Retrieval on World Wide Web Using Web Structure Mining", 2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation 978-0-7695-4062-7/10 $26.00 © 2010 IEEE DOI 10.1109/AMS.2010.108 532
[26] ABITEBOUL, S., PREDA, M., AND COBENA, G. Adaptive On-Line Page Importance Computation. In Proceedings of WWW2003 (Budapest, Hungary, May 2003).
[27] ARASU, A., NOVAK, J., TOMKINS, A., AND TOMLIN, J. PageRank Computation and the Structure of the Web: Experiments and Algorithms. In Proceedings of WWW2002 (Hawaii, USA, May 2002).
[28] LEMPEL, R., AND MORAN, S. (SALSA) the stochastic approach for link-structure analysis. ACM Transactions on Information Systems (2001).
[29] TIAN Chong "A Kind of Algorithm For Page Ranking Based on Classified Tree In Search Engine", 2010 International Conference on Computer Application and System Modeling (ICCASM 2010).
[30] Shiguang Ju, Zheng Wang, Xia Lv "Improvement of Page Ranking Algorithm Based on Timestamp and Link", 2008 International Symposiums on Information Processing.