RESEARCH ARTICLE OPEN ACCESS

Optimized Page Ranking Algorithm for Online Digital Libraries

Jigyasa Sidhu
M.Tech Research Scholar
Department of Computer Science and Engineering
Amity University, Noida
U.P - India

ABSTRACT

Digital libraries are emerging as a significant source for serving the queries of researchers for relevant document. With the growing digital content and the user's needs, it increases the complexity in the ranking mechanism utilized in digital libraries. Ranking plays an important role in digital libraries as it makes the user's search for scientific literature, research papers, or other academia based documents fruitful and avoids unnecessary navigation to find the desired content. Many ranking algorithms based on different parameters have already been proposed. The parameters like citations to a research paper, content of paper, impact factor of publication venue, age of the paper, bookmarks etc are utilized for ranking the documents in the result list of the digital libraries. The existing ranking algorithms sometimes provide irrelevant results due to certain shortcomings, which indicate a scope for further improvement in ranking mechanism. In this paper an optimized ranking algorithm is proposed that carries out static as well as dynamic ranking to rank the documents in digital libraries. The proposed algorithm considers the link structure of the papers i.e citations, bookmarks of the paper, paper age, and user's feedback via number of downloads of a paper as parameters and clustering process for producing efficient and relevant search results.

Keywords:- Digital library, Web Mining, Page ranking, PageRank, TDCC

I. INTRODUCTION

World Wide Web is composed of huge and massive volumes of information in the form of text, audio, video, images and meta data. It can be thought as a large database possessing unstructured or semi-structured chunks of data. For satisfying the information needs of the researchers and reducing the irrelevant search navigation, digital libraries have been introduced. A *digital library* [1][2]is an integrated collection of various services including catching, cataloging, storing, searching, guarding and retrieving digital content or information and provides clear and logical organization and convenient and easy access to typically huge amounts of digital information [1][2]. Today digital libraries are being utilized for various communities and in variety of different fields like academic, science, culture, health, and many more. Thus, the introduction of digital libraries has made the creation, storing, sharing and retrieving of information attractive and easy for the web users.

The complete architecture of the online digital library search system [1][2] is shown in figure 1. The main component of the architecture shown above is a crawler. Crawler like a spider crawls and traverses the hypertext structure of the World Wide Web. While traversing the web, crawler downloads the relevant web pages or gathers the required research papers that are published in some specific venue or publication like either in a conference or in a journal and lastly it stores the extracted contents in a database. Generally the publications or papers existing on web are in the form of either postscript files or PDF. Thus, whenever a user fires a search query, a new instance of the user or client agent is created which is responsible for locating and downloading the postscript files having either".ps" or ".ps.Z" or ".ps.gz" extensions [1]. After the files are downloaded, they are passed to the document parsing sub-agent. The document parsing sub-agent carry outs the extraction mechanism which extracts the meta data and the semantic features from each downloaded document and then save them into a database. These documents are known as the parsed documents. In the next step, these parsed documents are transferred to an indexing module. The indexing module is responsible for building the index utilizing the keywords of the documents or papers. Whenever the user submits a query to the digital library search interface in the form of keywords or terms, the query is accepted and processed by the database search and browsing sub agent. After processing the user query in appropriate syntax, the module returns an HTML formatted result to the user. The result set generated by the online digital library search engine is ordered by employing page ranking algorithms. Hence, an efficient search results are provided to the user against the query fired.

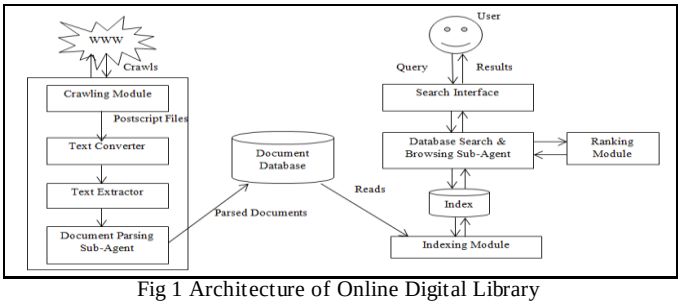

Fig 1 Architecture of Online Digital Library

Some of the significant benefits of digital libraries are listed below,

- *Improved access*

  Digital libraries are usually accessed via the Internet and thus any user can access the information content, provided by the digital library, from anywhere and anytime. They are free from physical location and the operating hours of traditional digital library.

- *Serve wide range of user's request*

  A digital library is capable of serving multiple information access requests for a document by variety of users across the globe. This is achieved because the digital libraries on receiving the simultaneous request for any document, creates multiple instances or copies of that document.

- *Improved information sharing*

  A digital library is capable of sharing its information with other similar digital libraries. This is done by utilizing the meta data content and appropriate information exchange protocols. The sharing of information among the digital libraries enhances the user's access.

The amount of digital content in digital libraries is rapidly growing which somewhere degrading the results of the ranking mechanism utilized by the search engines [2]. The existing ranking algorithms possess some shortcomings and provide a scope of enhancement in ranking of documents in digital libraries. In this paper, an optimized ranking mechanism for online digital libraries has been proposed and its comparison with some of the existing page ranking algorithm is done. This paper is structured as follows: in Section II, a review of some existing page ranking algorithms has been discussed. Section III introduces with an optimized ranking method with its architecture and illustration. Section IV presents a comparison of the proposed algorithm with some of the existing ranking algorithms. Finally in Section V, conclusion is drawn along with future suggestions.

II. REVIEW WORK

Many ranking algorithms for online digital libraries have been proposed. A detailed study of various page ranking mechanisms utilized for online digital libraries have been done. Some of the work done till now in the related field is mentioned below with its brief explanation.

A. PageRank Algorithm: Surgery Brin and Larry Page [3][4] presented a ranking mechanism which considers the backlinks along with the outgoing links of a research paper. It is one of the commonly utilized ranking algorithm and laid the basis for other ranking algorithms. This algorithm assigns a higher weightage to the incoming link that comes from an important research paper. This method calculates the rank of the paper u using the equation (1) [3][4]shown below,

$$PR(u) = (1-d) + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \qquad (1)$$

Where u denotes the paper whose rank is to be calculated, B(u) denotes the set of papers that points to u, PR (u) and PR (v) are page rank of paper u and v respectively, N_V denotes the number of outgoing edges of the paper v and d is the damping factor with value ranging from 0 to 1 and is generally assumed to be 0.85 [4].

B. Content Based Citation Count Ranking Algorithm: Shikha Singla et al. [1] presents an algorithm named C3 ranking algorithm for ranking papers in digital libraries, which considers two critical elements i.e. citations or references to the paper and the importance of the content with the user's query [1]. In this proposed algorithm, two parameters are considered for calculating the rank score of each paper. These parameters are: citations to the paper and the similarity among the paper's contents with the other papers which cited that research paper. Instead of reading the complete paper, summary of the paper is computed which is then used to determine the similarity score [1]. Therefore, it efficiently saves space and time. Also this paper lists the limitations in the existing ranking algorithms for the digital libraries for future improvement.

C. Popularity Weighted Ranking Algorithm: Yang Sun et al. [7] introduces a ranking method which takes into account a new popularity factor that is the venue of the publication (where the research paper is published) [7]. A popularity weighted ranking score of a publication in this proposed ranking method is defined by the two parameter namely, citations from other publications and the popularity factor of its publication [7]. Also in this paper, there is a comparison of this new method with the traditional PageRank, citation count algorithm and HITS algorithm. The new ranking method is implemented on the CiteSeer metadata [7]. As for future scope, user recommendations are not highly utilized in this method which can be added as a further improvement of this method.

D. Citation Count Ranking Algorithm: Joeran Beel et al. [2] proposed a ranking method named as Citation Count ranking algorithm for determining author's reputation [2]. This algorithm determines the significance of a research paper

according to the number of citations of a publication. In other words, greater the number of citations to a publication, higher will be its page rank. According to this algorithm, the citation count of a paper p is given using the equation (2) [2],

$$CC_p = |I_p| \qquad (2)$$

Where CC_p denotes the citation count of paper p and $|I_p|$ denotes total number of citations to paper p [2].

E. Time Dependent Citation Count Ranking Algorithm: Ludmila Marian [2] proposed a new ranking approach namely Time Dependent Citation Count Ranking Algorithm which extends Citation Count Algorithm [2]. This approach takes into account the time factor i.e the time of the citations of a paper along with the total number of citations for determining the paper rank. This paper highlights the significance of publication age as a ranking parameter. The paper is marked as important if its age resides within the selected time interval. The paper weight according to this algorithm is determined using equation (3) [2],

$$Weight_i = e^{-w(t_p - t_i)} \qquad (3)$$

Where t_p represents the current time i.e year, t_i denotes the year in which the paper i is published and w denotes the time-decay factor whose value lies between 0 and 1 and distinguishes between old and new publications [2].

F. Result Optimization Technique Based on Learning from Historical Query Logs: A.K. Sharma et. al proposed a optimized ranking technique that considers query logs for enhancing the search result list. This method predicts the user's behavior and navigation pattern over web so as to minimize navigation time within the search result list. The method involves clustering process for forming query clusters in query logs on the basis of a similarity measure. The query clustering is performed according to the query keywords and user's browsing history. It then discovers the sequential patterns of user's browsing of various web pages within each cluster utilizing sequential pattern mining technique. And thus this method re-ranks the result list according to the updated PageRank of the pages on the basis of extracted sequential patterns. Thus, this optimized method reduces the user's navigation time for retrieving the relevant information content and enhances the search engine's search result list.

G. SIMRANK, Page Rank Approach Based on Similarity Measure: Shaojie Qiao et. al [10] proposed an approach for ranking the web pages in the search engine result list according to the similarity measure named as SimRank [10]. This algorithm determines the similarity among pages and utilizes this similarity measure to partition the entire web database into numerous web social networks (WSNs) [10]. This method takes into account the social annotations for enhancing the ranking process. The web annotators [10] assign set of textual content with each page over web that provide a brief description regarding the web page to the user before visiting that page. This eliminates irrelevant user's navigation to random and least significant pages. These set of words are termed as annotations. Thus, this method computes a relevancy score by determining the similarity between the keywords of query and the annotations. This method computes the overall term weight which is calculated using equation (4) [10],

$$w_{ij} = \left\{0.5 + \frac{0.5 \times f_{ij}}{\max\{f_1, f_2, .. f_{|V|j}\}}\right\} \times \log\frac{N+1}{df_i} \qquad (4)$$

Where w_{ij} denotes the term weight of term i in page j, f_{ij} denotes the frequency of the term i in the page j, *N* denotes the number of pages in the web database, df_i denotes the number of pages in which the term i comes atleast once and |V| is the vocabulary size.

Thus the similarity measure is computed according to the equation (5) [10],

$$sim(p_a, p_b) = \frac{\sum_{i=1}^{n} w_{ipa} \times w_{ipb}}{\sum_{i=1}^{n} w_{ipa}^2 + \sum_{i=1}^{n} w_{ipb}^2 - \sum_{i=1}^{n} w_{ipa} \times w_{ipb}} \qquad (5)$$

where $sim(p_a, p_b)$ represents the similarity score between pages p_a and p_b.

The major challenges in the existing ranking algorithms provide a scope for improvement and development of an optimized page ranking algorithm for online digital libraries. They are listed below,

- The existing page ranking algorithms are not capable enough to utilize the relevance implied by user surfing patterns to improve the ranking of web pages.
- The ranking algorithm considers only link structure which reduces their effectiveness by displaying fake and irrelevant links to the user as they only depends on the link structure of citation graph instead of the query.
- User's session timestamp for a particular document is a significant parameter used for ranking purpose but there are chances that the complete session is not utilized for exploring the particular web page, rather the user's system is simply left unused for long period of time. This results into inappropriate rank to that web page.
- In some research documents the annotations may be sparse and incomplete hence it creates a gap between annotations and queries which affect the relevance score [10].

- The effectiveness of the ranking algorithms sometimes gets affected by the capabilities of the web crawler being utilized [5].

III. PROPOSED PAGE RANKING ALGORITHM

An optimized page ranking algorithm has been proposed that utilizes the bookmarks for computing the overall rank score of the documents in the online digital libraries and perform ranking on the basis of this rank score. Bookmarks are set of keywords that identify the document or some part of the document. For instance, the title, headings and the sub-headings are by default the bookmarks of the research document. This ranking algorithm carry outs two types of ranking namely, static ranking and dynamic ranking. Static ranking performs the ranking procedure on the basis of significant parameters. This algorithm takes into account number of downloads, paper posted time and initial page rank for static ranking. And the dynamic ranking performs the ranking procedure on the basis of the user query in the form of keywords. The complete architecture of the proposed algorithm is shown in figure 2.The proposed algorithm works in the following two phases mentioned and explained below,

- Paper Upload
- Paper Search

A. Paper Upload

When a user uploads a new paper via upload interface, it is stored in a paper repository along with the existing research papers. Each paper in the paper repository undergoes various mechanisms. Firstly, the *Text Extractor Module* extracts the complete text from the paper including title, author, keywords and bookmarks. If the paper has no bookmarks attached to it, then the *Bookmark Creator* creates the same for that particular paper. The extracted content is stored in a content store for future usage. Secondly, the *Similarity Analyzer* retrieves the keywords from the content store to compute a similarity score between them. This score indicates the similarity among the uploaded paper and the existing papers in the paper repository. Once this score is calculated, it is utilized by the *Clustering Tool* for forming the paper clusters which are stored in the clustering database. Within each cluster formed, *Static Ranking* is performed to determine paper weight and to rearrange the documents in the cluster according to the weights calculated. Lastly, each cluster with proper sequence of papers is maintained in the clustering database.

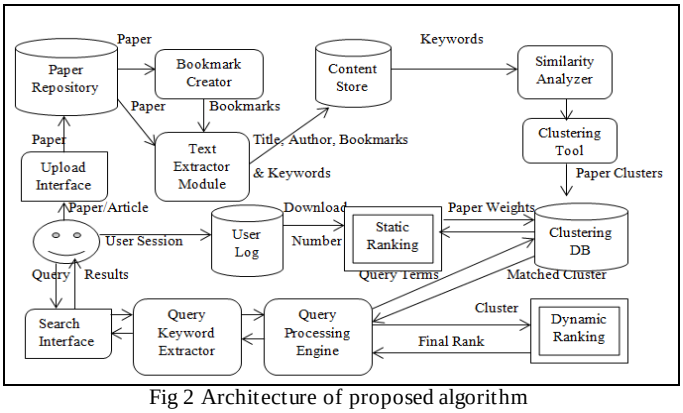

Fig 2 Architecture of proposed algorithm

B. Paper Search

When a user fires a query to the search engine via search interface, the *Query Keyword Extractor* extracts the keywords from the user's query. The Query Processing Engine accepts these keywords and selects an appropriate cluster against the user's query from the clustering database. This is done by computing similarity between the query keywords and the cluster keywords. Once a suitable cluster is selected from the database, the final rank of each paper is determined by performing *Dynamic Ranking* on the selected cluster. The papers in the cluster selected is re-ordered on the basis of the final rank of each paper and are displayed to the user in the form of search result set against the query entered.

The working of different functional components of the proposed algorithm is described below:

1) Bookmark Creator

The proposed algorithm utilizes bookmarks of the research paper to enhance the ranking mechanism. The bookmark creator component extracts the title, headings and sub-headings from the research paper not having bookmarks by default. The contents retrieved are taken as bookmarks of that paper.

2) Text Extractor Module

This module parses the research paper completely and extracts the text from the paper. The text retrieved is title, authors, keywords and attached bookmarks of the paper. Keywords along with their frequency of occurrence are extracted from the bookmarks by excluding the stop words. The data gained from this module is saved in the content store for future use.

Similarity analyzer takes the most commonly occurring keywords of every paper from the content store as input and outputs a value indicating similarity between the research papers. The similarity measure lies in the range of 0 to 1 with 0 representing no similarity and 1 being exactly similar. Similarity analyzer utilizes the algorithm shown in fig 3.

Algorithm: Similarity_Calculation (paper1, paper2)

1. $P1 \leftarrow Keyword\_Extract(paper1)$
2. $P2 \leftarrow Keyword\_Extract(paper2)$
3. $Score \leftarrow Sim_{keyword}(P1, P2)$
4. Return Score

Fig 3 Algorithm for Similarity Calculation between the research papers

The algorithm takes two papers from the paper repository as input to compute the similarity score between them. The functions used in the algorithm are described below:

Keyword_Extract ():
This function takes the paper as input and extracts the keywords of that paper from the content store. Then it checks the frequency of the keywords and return only top most occurring keywords of the paper.

$Sim_{keyword}()$:
The keywords of the papers to be matched are taken as input in this function. This function computes the similarity score between the keywords using the equation (6) shown below,

$$sim(q,d) = \frac{\sum W_{q,i} \times W_{d,i}}{\sqrt{\sum W^2_{q,i}} \times \sqrt{\sum W^2_{d,i}}} \qquad (6)$$

where $W_{q,i}$ and $W_{d,i}$ denotes the weight of term t_i in the user's query q and paper d respectively. These weights can be computed by calculating the frequency of occurrence of term t_i in q and d [9].

3) Clustering Tool

The clustering of papers is carried out by the clustering tool as shown in the algorithm in figure 4. Similarity score calculated by the similarity analyzer is utilized by the clustering tool for forming the clusters. The algorithm takes the paper to be uploaded and its similarity score as input to form clusters. It calculates the distance of the score of the paper uploaded with the existing papers in the repository. According to this distance and the threshold the clusters are formed and are saved in the clustering database.

Algorithm: Cluster (Paper, Score)

1. Set n = 0 // n is the number of clusters
2. Set $\tau = 0.3$ // Similarity threshold
3. Let $C = \{C_k, C_{k+1}, \ldots, C_{k+n}\}$ // Set of clusters
4. Set $C_k \leftarrow null$ & $n = n + 1$ // Initial cluster & increment n
5. For each paper p $\in P_k$ // P_k is the paper in the paper repository
6. $D_k \leftarrow Distance(Paper, p)$
7. If $D_k \geq \tau$ $C_k = C_k \cup p$
8. Else initialize new cluster, $C_{k+1} = p$ & $n = n + 1$
9. Return C & n

Fig 4 Algorithm for Clustering process

The function used in the algorithm is described below:

$Distance(Paper, p)$:
This function computes a value indicating the distance between the two papers taken as input. The value obtained will be compared to the threshold value assumed and accordingly the cluster for paper p will be decided. The distance between the papers is calculated using the equation (7) shown below,

$$Distance(Paper, p) = |Score(paper) - Score(p)| \qquad (7)$$

where Score(paper) and Score(p) are the similarity score of paper and p respectively computed by similarity analyzer.

After the clusters are formed and stored in the database, set of keywords are assigned to each cluster. For each cluster an intersection operation is performed between the keywords of the papers stored within that cluster to gain the cluster's keywords.

4) Static Ranking

Once the clusters are formed, the static ranking is performed within each cluster to arrange the papers according to the three parameters namely,

- Number of Downloads
- Paper Posted Time
- Initial Page Rank

Number of Downloads
User Log is maintained that stores every user's session and is utilized to gain the number of downloads of every paper. Thus, this parameter generates a download score for each paper P in the database by using the equation (8) shown below.

$$Download\ Score(P) = \frac{Number of\ Downloads(P)}{Maximum\ Downloads} \qquad (8)$$

Paper Posted Time
Paper posted time is computed by using *Time Dependent Citation Count Algorithm (TDCC)*. This algorithm is the most commonly used for ranking the documents in the online digital libraries. It utilizes the citation graph of nodes interconnected with each other through edges. Hence, using equation (3) the TDCC rank of each paper is computed.

Initial Page Rank
Initial page rank of the paper is calculated by using the *PageRank Algorithm*. This method computes the rank of a paper by considering the number of citations (i.e backlinks) that paper has. Thus this method calculates the rank of the paper using the equation (1) described in the previous section.

Paper Weight
Hence, for each paper in a cluster, a paper weight is computed by using the equation (9) and is stored in the clustering database. The papers within each cluster are re-arranged on the basis of this paper weight.

$$Paper\ Weight = Download\ Score + Posted\ Time + PageRank \qquad (9)$$

5) Query Keyword Extractor

This module extracts the keywords from the user's query by removing the stop words. The keywords extracted are passed to the query processing engine for selecting an appropriate cluster by matching extracted query keywords and the cluster keywords.

6) Dynamic Ranking

Dynamic ranking involves re-ordering of the papers according to the user's query within the selected cluster. The final rank of each paper within the cluster that will form the sequence in which the papers will be displayed to the user is determined by computing similarity between the query keywords and the cluster keywords. The similarity measure shown in equation (6) is used for finding out the similarity score. The final rank is obtained using the equation (10) shown below,

$$Final\ Rank = Paper\ Weight + Similarity\ Score \quad (10)$$

The papers in the result set provided to the user are ranked according to the final rank of each paper computed.

C. Illustration of Proposed Algorithm

An example is taken to explain the ranking mechanism of the proposed algorithm. The existing papers in the database are shown in table I.

TABLE I
PAPER REPOSITORY

S.No	Paper Title
A	Web Mining Research: A Survey
B	Web Crawler Architecture
C	Network Security: History, Importance, and Future
D	Page Ranking Algorithms for Web Mining
E	A Survey- Link Algorithm for Web Mining
F	How search engines work and a web crawler application
G	Network Security: it's time to take it seriously
H	Network Security Attacks Solution and Analysis
I	Application of Page Ranking Algorithm in Web Mining
J	Weighted Page Rank Algorithm Based on Number of Visits of Links of Web Page
K	A Crawler-based Study of Spyware on the Web
L	Network Security Using Cryptographic Techniques
M	Cybercrime: A threat to Network Security
N	Comparative study of Page Ranking Algorithms for Web Mining
O	Mercator: A scalable, extensible Web crawler
P	Web Crawler: Extracting the Web Data
Q	Analysis of Various Web Page Ranking Algorithms in Web Structure Mining
R	Design and Implementation of a High-Performance Distributed Web Crawler*
S	A Review of types of Security Attacks and Malicious Software in Network Security
T	Significances and Issues of Network Security

Firstly, bookmarks from each paper in the database are extracted which are utilized for determining the frequently occurring keywords for each paper. Table II lists the frequently occurring keywords of the papers in the database.

TABLE II
MOST FREQUENT KEYWORDS IN EACH PAPER IN THE DATABASE

S.No	Keywords
A	Mining, Web, View, Research, Survey, Overview, Categories, Agent, Paradigm, Content
B	Web, Crawler, Architecture, Historical, Background, Foundation, Key, Application, Future, Directions
C	Security Network, Internet, History IPv4, IPv6, Architecture, Current, Protocol, Attacks
D	Web, Mining, Page, Algorithms, Ranking, Categories, Content, Structure, Usage, Link
E	Web, Mining, Page, Rank, Weighted, Content, Link, Algorithm, Algorithms, Survey
F	Web, Search, Indexing, engines, crawler, crawling, application, content, work, popular
G	Security, Network, Internet, Architecture, IPv4,developments, IPv6, history, future, time
H	Network, Security, Solution, Attacks, Analysis, various, attacking, methods, effective
I	Algorithm, Page, Ranking, Web, Mining, Application, Methodologies, Weighted, Rank,

	HITS
J	Weighted, Page, Visits, Links, Rank, Algorithm, Number, Web, PageRank, Result
K	Spyware, Web, Crawler, Study, Infected, Executables, Crawling, examining, changing, environment
L	Network, Security, Cryptographic, Techniques, Related, Survey
M	Research, threat, Network, Security, Analysis, Cybercrime, Aims, Objectives, Design, Data
N	Ranking, Algorithms, Page, Comparative, study, Web, Mining, Text, Link, Analysis
O	Crawler, scalable, Web, Mercator, extensible, Architecture, Extensibility, traps, hazards, Results
P	Web, Crawler, Crawling, Extracting, Data, Literature, Survey, Architecture, Types, Algorithms
Q	Web, Page, Ranking, Algorithms, Analysis, various, Structure, mining, Comparison
R	System, Crawler, Application, Crawl, Manager, Performance, Implementation, Crawling, Structure, Network
S	types, Security, Attacks, Malicious, Software, Review, Network
T	Security, Authentication, Issue, Network, Integrity, Significances, Services, Confidentiality, Peer, Entity

Now, the similarity analyzer will compute a similarity score between the already existing papers in the database. This will be done by matching the keywords listed in table 2 among each other and computing a similarity score. The similarity score is computed using the equation [9] (6),

The similarity scores gained after the comparison of the research papers are presented using a matrix namely similarity matrix which an n x m matrix where n and m represents a specific paper in the database and each entry corresponds to the similarity between them. The threshold value is assumed to be 0.2. The similarity matrix is shown in table III.

TABLE III
SIMILARITY MATRIX

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
A	1	0.198	0	0.886	0.726	0.360	0	0	0.353	0.114	0.397	0.036	0.059	0.224	0.225	0.486	0.510	0	0	0
B	0.198	1	0.040	0.204	0.163	0.323	0.127	0	0.124	0.057	0.25	0	0	0.052	0.391	0.438	0.186	0.216	0	0
C	0	0.40	1	0	0	0	0.924	0.598	0	0	0	0.466	0.375	0	0.022	0.016	0	0.086	0.415	0.515
D	0.886	0.204	0	1	0.830	0.376	0	0	0.455	0.198	0.411	0	0	0.488	0.233	0.519	0.744	0.024	0	0
E	0.26	0.163	0	0.830	1	0.325	0	0	0.576	0.518	0.326	0.035	0	0.401	0.185	0.422	0.607	0.019	0	0
F	0.360	0.323	0	0.376	0.325	1	0	0	0.165	0.100	0.436	0	0	0.091	0.424	0.635	0.324	0.215	0	0
G	0	0.127	0.924	0	0	0.376	1	0.497	0	0	0	0.430	0.346	0	0	0.030	0	0	0.343	0.479
H	0	0	0.598	0	0	0	0.497	1	0	0	0	0.384	0.387	0.039	0	0	0.046	0.107	0.362	0.358
I	0.353	0.124	0	0.455	0.576	0.165	0	0	1	0.383	0.166	0	0	0.393	0.094	0.198	0.463	0.044	0	0
J	0.114	0.057	0	0.198	0.518	0.100	0	0	0.383	1	0.115	0	0	0.091	0.065	0.138	0.250	0	0	0
K	0.397	0.25	0	0.411	0.326	0.436	0	0	0.166	0.115	1	0	0	0.131	0.340	0.597	0.372	0.090	0	0
L	0.036	0	0.466	0	0.035	0	0.430	0.384	0	0	0	1	0.268	0	0	0.051	0	0.093	0.235	0.345
M	0.059	0	0.375	0	0	0	0.346	0.387	0	0	0	0.268	1	0	0	0.020	0.064	0.074	0.189	0.277
N	0.224	0.052	0	0.488	0.401	0.091	0	0.039	0.393	0.091	0.131	0	0	1	0.059	0.188	0.751	0	0	0

O	0.225	0.391	0.022	0.233	0.185	0.424	0	0	0.094	0.065	0.340	0	0	0.059	1	0.656	0.211	0.245	0	0
P	0.486	0.438	0.016	0.519	0.422	0.635	0.030	0	0.198	0.138	0.597	0.051	0.020	0.188	0.656	1	0.494	0.229	0.072	0
Q	0.510	0.186	0	0.744	0.607	0.324	0	0.046	0.463	0.250	0.372	0	0.064	0.751	0.211	0.494	1	0.044	0	0
R	0	0.216	0.086	0.024	0.019	0.215	0	0.497	0.044	0	0.090	0.093	0.074	0	0.245	0.229	0.044	1	0.043	0.077
S	0	0	0.415	0	0	0	0.343	0.362	0	0	0	0.235	0.189	0	0	0.072	0	0.043	1	0.260
T	0	0	0.515	0	0	0	0.479	0.358	0	0	0	0.345	0.277	0	0	0	0	0	0.077	1

Now, the clusters of the research papers existing in the database are formed using the similarity matrix obtained and are saved for future use. On the basis of the similarity matrix three clusters are formed and are shown in table IV, V and VI

TABLE IV
RESEARCH PAPERS IN CLUSTER I

S.No	Paper Title
A	Web Mining Research: A Survey
D	Page Ranking Algorithms for Web Mining
E	A Survey- Link Algorithm for Web Mining
I	Application of Page Ranking Algorithm in Web Mining
J	Weighted Page Rank Algorithm Based on Number of Visits of Links of Web Page
N	Comparative study of Page Ranking Algorithms for Web Mining
Q	Analysis of Various Web Page Ranking Algorithms in Web Structure Mining

TABLE V
RESEARCH PAPERS IN CLUSTER II

S.No	Paper Title
B	Web Crawler Architecture
F	How search engines work and a web crawler application
K	A Crawler-based Study of Spyware on the Web
O	Mercator: A scalable, extensible Web crawler
P	Web Crawler: Extracting the Web Data
R	Design and Implementation of a High-Performance Distributed Web Crawler*

TABLE VI
RESEARCH PAPERS IN CLUSTER III

S.No	Paper Title
C	Network Security: History, Importance, and Future
G	Network Security: it's time to take it seriously
H	Network Security Attacks Solution and Analysis
L	Network Security Using Cryptographic Techniques
M	Cybercrime: A threat to Network Security
S	A Review of types of Security Attacks and Malicious Software in Network Security
T	Significances and Issues of Network Security

After the retrieval of clusters, they are saved in the cluster database. Set of keywords are attached to each cluster that are listed in table VII.

TABLE VII
KEYWORDS ATTACHED TO EACH CLUSTER

Cluster	Keywords
Cluster 1	web, mining, rank, algorithms, page, ranking, structure, link, categories, content, weighted, algorithm
Cluster 2	Web, crawler, architecture, application, crawling, historical, background, foundation, key, future, directions, search
Cluster 3	network, security, IPv4, IPv6, history, architecture, developments, attacks, internet, analysis, current, protocol

Now, static ranking mechanism is performed for computing the weight for each paper within a cluster. The static ranking considers number of downloads, paper posted time and initial page rank of the paper. The number of download of each

paper is assumed in this example which is utilized to compute an average download score is for every paper.

Consider the citation graph of papers in the database shown in figure 5, where each node represents a paper stored in the database and edge from one node to other represents citation of the paper.

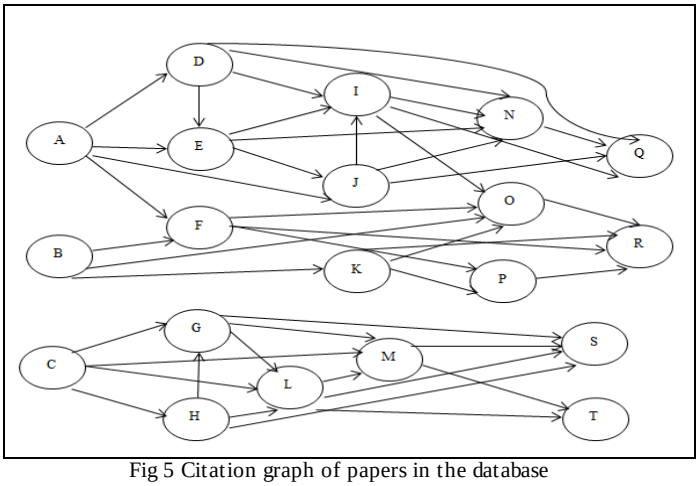

Fig 5 Citation graph of papers in the database

The paper posted time is calculated by utilizing Time Dependent Citation Count Algorithm (TDCC) [2] on the citation graph in figure 5 and the data regarding year of paper publish in table VIII.

TABLE VIII
DATA RETRIEVED FROM CITATION GRAPH

Paper S.No	Publication Year
A	2000
B	2000
C	2000
D	2011
E	2011
F	2011
G	2011
H	2011
I	2012
J	2012
K	2012
L	2012
M	2012
N	2013
O	2013
P	2013
Q	2014
R	2014
S	2014
T	2014

The score of each paper in cluster 1 is obtained by using equations [2] (11), (12), (13), (14), (15), (16) and (17) and are shown below. The value of time delay factor is assumed to be 6 years.

$$W_A = 0 \quad (11)$$
$$W_D = e^{-w(15)} \quad (12)$$
$$W_E = e^{-w(15)} + e^{-w(4)} \quad (13)$$
$$W_I = e^{-w(15)} + e^{-w(4)} + e^{-w(4)} + e^{-w(3)} \quad (14)$$
$$W_J = e^{-w(15)} + e^{-w(4)} \quad (15)$$
$$W_N = e^{-w(4)} + e^{-w(4)} + e^{-w(3)} + e^{-w(3)} \quad (16)$$
$$W_Q = e^{-w(4)} + e^{-w(3)} + e^{-w(3)} \quad (17)$$

After solving above equations the TDCC obtained for each paper in cluster 1 is listed in table IX below.

TABLE IX
TDCC OF PAPERS IN CLUSTER 1

Paper S.No	TDCC
A	0
D	0.0371
E	0.0470
I	0.0644
J	0.0470
N	0.0347
Q	0.0247

The initial page rank [3][4] of the papers in the cluster 1 is computed by applying PageRank algorithm on the citation graph shown in fig 5. The PageRank of the papers is calculated using the equations [3][4] (18), (19), (20), (21), (22), (23) and (24) shown below and is shown in table X,

$$PR(A) = (1 - 0.85) + 0.85\,[0] \quad (18)$$
$$PR(D) = (1 - 0.85) + 0.85\left[\frac{PR(A)}{4}\right] \quad (19)$$
$$PR(E) = (1 - 0.85) + 0.85\left[\frac{PR(A)}{4} + \frac{PR(D)}{4}\right] \quad (20)$$
$$PR(I) = (1 - 0.85) + 0.85\left[\frac{PR(D)}{4} + \frac{PR(E)}{3} + \frac{PR(J)}{3}\right] \quad (21)$$
$$PR(J) = (1 - 0.85) + 0.85\left[\frac{PR(A)}{4} + \frac{PR(E)}{3}\right] \quad (22)$$
$$PR(N) = (1 - 0.85) + 0.85\left[\frac{PR(D)}{4} + \frac{PR(I)}{3} + \frac{PR(E)}{3} + \frac{PR(J)}{3}\right] \quad (23)$$
$$PR(Q) = (1 - 0.85) + 0.85\left[\frac{PR(D)}{4} + \frac{PR(J)}{3} + \frac{PR(I)}{3} + PR(N)\right] \quad (24)$$

TABLE X
PAGERANK OF PAPERS IN CLUSTER 1

Paper S.No	PageRank
A	0.15
D	0.18
E	0.22
I	0.31
J	0.24
N	0.40
Q	0.68

The computed paper weight after static ranking for cluster I, II and III are listed in table XI, XII and XIII respectively.

TABLE XI
STATIC RANKING WITHIN CLUSTER I

Paper S.No	Download Score	Page Rank	TDCC	Paper Weight
A	0.8	0.15	0	0.95
D	0.9	0.18	0.0371	1.1171
E	0.7	0.22	0.0470	0.967
I	0.8	0.31	0.0644	1.1744
J	0.7	0.24	0.0470	0.987
N	1	0.40	0.0347	1.4347
Q	0.9	0.68	0.0247	1.6047

TABLE XII
STATIC RANKING WITHIN CLUSTER II

Paper S.No	Download Score	Page Rank	TDCC	Paper Weight
B	0.9	0.15	0	1.05
F	0.8	0.22	0.0743	1.0943
K	0.7	0.19	0.0470	0.937
O	0.8	0.30	0.0619	1.1619
P	0.9	0.26	0.0173	1.1773
R	0.7	0.74	0.0173	1.4573

TABLE XIII
STATIC RANKING WITHIN CLUSTER III

Paper S.No	Download Score	Page Rank	TDCC	Paper Weight
C	0.9	0.15	0	1.05
G	0.8	0.23	0.0470	1.077
H	0.7	0.18	0.0371	0.9171
L	0.8	0.29	0.0099	1.0999
M	0.9	0.32	0.0619	1.2819
S	0.7	0.48	0.1346	1.3146
T	0.8	0.36	0.0014	1.1614

The total weight of each paper within a cluster is obtained by adding all the three parameters. Then, each cluster is rearranged according to the computed weight of the papers. Hence, the sequence of the research papers stored in the clusters formed is,

Cluster I: Q, N, I, D, J, E, A
Cluster II: P, O, R, F, B, K
Cluster III: S, M, T, L, G, C, H

Let the user query be Q, which the user submits to the search engine through query interface for retrieving the relevant documents.

Query Q: Concept of page ranking algorithms in web mining. The query keyword extractor extracts the keywords from the user's query which are listed below,

Query Keywords: Concept, page, ranking, algorithms, web, mining.

Now, the query processing engine will match the above listed keywords with the cluster keywords mentioned in table 7 so as to select the appropriate cluster for serving the user's query. The similarity score between the query and the cluster computed using the equation (6) is show in table XIV.

TABLE XIV
SIMILARITY SCORE BETWEEN THE CLUSTER AND QUERY

Cluster	Similarity Score	Selected Cluster (Yes/ No)
Cluster 1	0.566	Yes
Cluster 2	0.213	No
Cluster 3	0	No

Clearly, it can be seen from the table XIV that the cluster 1 is the suitable cluster for forming the result set of the query fired. Now, the papers in the matched cluster will be rearranged according to the final rank computed by dynamic ranking mechanism. The final rank of each paper in cluster 1 is shown in table XV.

TABLE XV
FINAL RANK OF EACH PAPER IN CLUSTER 1

Paper S.No	Paper Weight	Similarity Score	Final Rank
A	0.95	0.549	1.499
D	1.1171	0.752	1.8691
E	0.967	0.657	1.624
I	1.1744	0.482	1.6564
J	0.987	0.223	1.21
N	1.4347	0.748	2.1827
Q	1.6047	0.720	2.3247

.

Now, the papers in the cluster 1 will be re-ordered according to the final rank computed and will be displayed to the user as search result set. The final result set provided to the user is shown in table XVI.

TABLE XVI
FINAL RESULT SET AGAINST THE USER'S QUERY

S.No	Paper Title
Q	Analysis of Various Web Page Ranking Algorithms in Web Structure Mining
N	Comparative study of Page Ranking Algorithms for Web Mining
D	Page Ranking Algorithms for Web Mining
I	Application of Page Ranking Algorithm in Web Mining
E	A Survey- Link Algorithm for Web Mining
A	Web Mining Research: A Survey
J	Weighted Page Rank Algorithm Based on Number of Visits of Links of Web Page

IV. RESULT ANALYSIS

This section presents the comparison between the results of the proposed algorithm with the existing ranking approaches namely, PageRank and Time Dependent Citation Count (TDCC). The papers in the cluster I are used for the

comparison purpose. The rank of each paper in cluster I obtained by applying the three algorithms are shown in table XVII .

TABLE XVII
RESULT OF THREE RANKING APPROACHES

Paper S.No	PageRank	TDCC	Proposed Algorithm
A	0.15	0	1.499
D	0.18	0.0371	1.8691
E	0.22	0.047	1.624
I	0.31	0.0644	1.6564
N	0.4	0.047	1.21
Q	0.68	0.0347	2.1827
R	0.74	0.0247	2.3247

V. EXPERIMENTAL RESULTS

A database of some research papers are taken into consideration for carrying out proposed ranking mechanism. The below results shows search result list against the user query utilising the proposed algorithm.

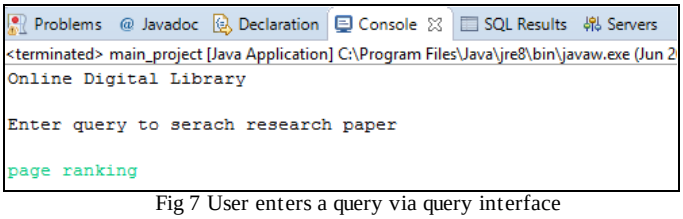

Fig 7 User enters a query via query interface

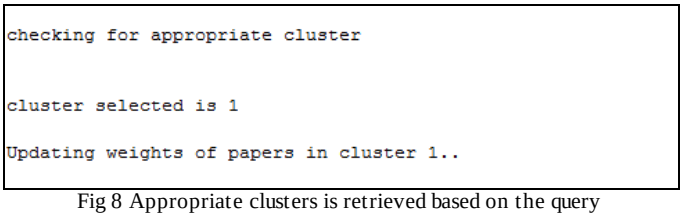

Fig 8 Appropriate clusters is retrieved based on the query

It can be clearly seen that results obtained from the three ranking approaches are different from each other. Results analysis is presented through a graph shown in figure 6.

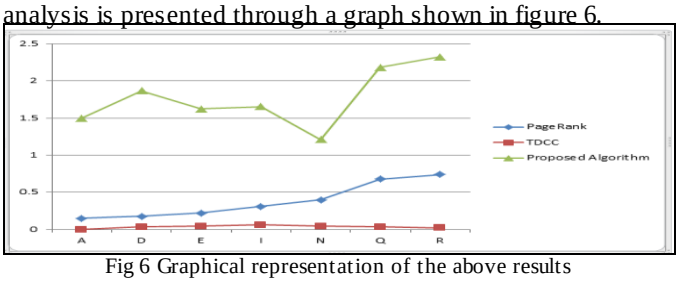

Fig 6 Graphical representation of the above results

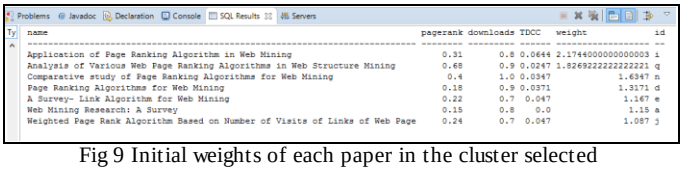

Fig 9 Initial weights of each paper in the cluster selected

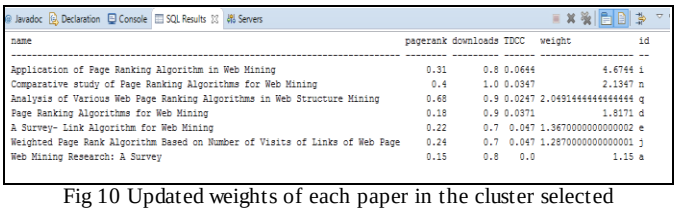

Fig 10 Updated weights of each paper in the cluster selected

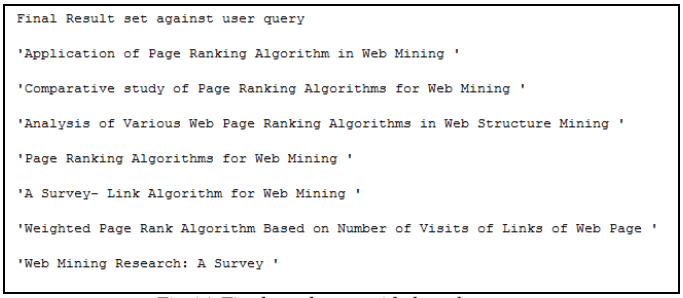

Fig 11 Final results provided to the user

VI. COMPARISON

This section presents the comparison between the proposed algorithm and some of the existing ranking approaches namely, PageRank, TDCC and SimRank [2]. The comparison among the ranking algorithms are carried out on the basis of various parameters like technique used, input parameter, complexity etc. The complete comparison is shown in table XVIII.

TABLE XVIII
COMPARISON OF PAGERANK, TDCC, SIMRANK AND PROPOSED ALGORITHM

Algorithms → / Measures ↓	PageRank	TDCC	SimRank	Proposed Algorithm
Main Technique used	Web structure mining	Web structure mining,	Web Content Mining	Web Structure Mining, web content mining, web usage mining

Description	Papers are sorted according to the link structure of the papers and citations to the paper.	Papers are ranked based on the age of the citations i.e publication time (year) of the paper.	Papers are ordered according to the content similarity rather than the link structure of the papers.	Papers are ranked by considering the link structure as well as content similarity among the papers. It also involves clustering of papers for enhancing the results.
Input Parameters	Backlinks	Incoming links, Paper posted time	Paper's and query's contents.	Bookmarks, query's content, paper posted time, number of downloads
Complexity	O(logN)	O(N^2)	O(N^2)	O(N)
Relevancy	Less	More(more than PageRank but less than Citation Count)	Results gained are relevant and better than the traditional PageRank and various extensions of PageRank.	Results are enhanced and relevant than SimRank and other ranking approaches
Quality of results	Medium	Higher than PageRank	Improved efficiency and correctness in ranking of papers	Enhanced search result list with higher accuracy and relevant ranking results
Importance	Traditional method that focuses on the link structure to determine relevance.	This method considers the citations of a paper and distinguishes old and new citations.	Effectively examine papers or documents with few textual contents i.e annotations	Enhance the ranking approach by forming paper clusters and involving static and dynamic ranking.
Limitations	Results obtained at the time of indexing and not at the query time.	It only considers the time of the citation but not the relevancy and significance of each citation.	Its efficiency gets affected by the capabilities of the web crawler being utilized.	It does not involve user's browsing pattern and recommendations.

VII. CONCLUSION

The existing ranking approaches possess few limitations due to which they sometimes fail to produce effective results against the user's query. Since, the researchers depend on the digital libraries for retrieving the needful information content, therefore it is necessary to overcome these shortcomings. The paper presents an optimized ranking approach that enhances the ranking mechanism and provides better and relevant results than the existing algorithms. The already proposed algorithms are either based on content similarity or link structure. But, this proposed approach takes into account both the parameter mentioned as well as bookmarks of the paper. The final rank of each paper is computed by combining the result of static and the dynamic ranking. Thus it enhances the search results and relevancy of the paper.

REFERENCES

[1] Shikha Singla, Neelam Duhan, Usha Kalkal, "A Novel Approach for Document Ranking in Digital Libraries using Extractive Summarization", International Journal of Computer Applications (0975 – 8887), vol. 74, no.18, pp. 25-31, 2013.

[2] Sumita Gupta, Neelam Duhan, Poonam Bansal "*A comparative study of page ranking algorithms for online digital library*", International Journal of Scientific & Engineering Research, vol. 4, no. 4, 2013.

[3] T.Munibalaji, C.Balamurugan, "Analysis of Link Algorithms for Web Mining", International Journal of Engineering and Innovative Technology (IJEIT), Vol. 1, Issue. 2, pp. 81-86, 2012.

[4] N. Duhan, A.K. Sharma, K.K. Bhatia, "Page Ranking Algorithms: A Survey", in Proceedings of the IEEE International Conference on Advance Computing, pp. 2811-2818, 2009.

[5] S. Aggarwal, "Comparative study of Page Ranking Algorithms for Web Mining", in International Journal of Computer Trends and Technology, vol. 4, no. 4, pp. 898-902, 2013.

[6] R. Jain, Dr. GN. Purohit, "Page Ranking Algorithm for Web Mining", in International Journal of Computer

Applications (0975 – 8887), Vol. 13, Issue. 5, pp. 22-25, 2011

[7] Y. Sun, C. L. Giles, "Popularity Weighted Ranking for Academic Digital Libraries", Advances in Information Retrieval Lecture Notes in Computer Science, vol. 4425, pp 605-612, 2007.

[8] A.K. Sharma, N. Dhuan, GKumar, "A Novel Page Ranking Method based on Link-Visits of Web Pages", in Int. J. of Recent Trends in Engineering and Technology, vol. 4, no. 1, pp. 58-63, 2010.

[9] N. Dhuan, A.K. Sharma, "A Novel Approach for Organizing Web Search Results using Ranking and Clustering", in International Journal of Computer Applications (0975 – 8887), vol. 5, no.10, 2010.

[10] S. Qiaot, T. Li, H. Li, Y. Zhu, J. Pengt and J. Qiu, "SimRank: A Page Rank Approach based on Similarity Measure", IEEE, 2010.

[11] D.K.Sharma and A.K. Sharma (2010), "A Comparative Analysis of Web Page Ranking Algorithms", in International Journal on Computer Science and Engineering, Vol. 2, Issue. 8, pp. 2670-2676, 2010.

[12] A.K. Singh, R. Kumar, "A Comparative Study of Page Ranking Algorithms for Information Retrieval", in International Journal of Electrical and Computer Engineering, pp. 469-480, 2009.

[13] Neelam Duhan, A. K. Sharma, "A Novel Approach for Organizing Web Search Results using Ranking and Clustering", International Journal of Computer Applications (0975 – 8887), vol. 5, no.10 , pp. 1-9, 2010.