JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 2, 2022, p. 2839 - 2854
https://publishoa.com
ISSN: 1309-3452

Jaccard Snowball Robust Regression Based Page Ranking With Big Data In Dynamic Web Environments

Mr. P. Sujai, Dr. V. Sangeetha., M.Sc., Mphil., Ph.D, Mrs. H. M. Manjula

Asst.Professor,Department of computer Applications, IZEE College of Management and Information science,Bangalore. spsujai2@gmail.com.
Asst.professor, Department of computer science, Govt.arts College, pappireddipatti,
Tamil Nadu. sangee759@gmail.com.
Asst.Professor, Department of computer Applications ,IZEE College of Management and Information science, Bangalore. manjulahm2007@gmail.com.

ABSTRACT

World Wide Web (WWW) comprises a large volume of information and provides access to the users at anyplace and anytime. By increase of resources, users have several difficulties for detecting valuable data. But, web information methods are focused on relevant data but it still not concentrated on pages interesting to the users. In order to find the interesting web pages of the user, a novel machine learning technique called Jaccard Snowball Preprocessed Tanimoto Robust Regressive Page Ranking (JSPTRRPR) method is introduced. JSPTRRPR method was used for enhancing accuracy of web page ranking with minimum time consumption. In order to achieve this contribution, the JSPTRRPR method includes two major processes namely preprocessing and page ranking. WWW comprises a number of information. The JSPTRRPR method considers the number of user query as an input. Jaccard Snowball Stem Preprocessing Model is applied for eliminating stop as well as stem words from the input user query. After that, Tanimoto Robust Regressive Page Ranking Model is carried out to rank the significant pages at the top based on the user queries. Robust regression analysis used to find the top-ranked web pages by measuring theTanimoto similarity between the user query and webpage contents. The web pages with higher similarity value get ranked at the top using the Borda count fractional ranking method. Finally, a series of experimental results has signified and confirmed that the JSPTRRPR method achieves prominent performance in terms of higher ranking accuracy and minimum false positive rate as well as ranking time by number of user query.

Keywords: WWW, Big Data, Jaccard index, Snowball Stem Preprocessing, Tanimoto Robust Regression, Borda count fractional ranking method

1. INTRODUCTION

Web mining is a process of mining useful or important information on query. Page ranking is a major role in the web mining process. Web page ranking is a significant aspect for attaining relevant pages on user.

ACVPR was designed in [1] for web page ranking to personalized requirements based on search query using logistic regression. But the performance of web page ranking time was not minimized. A ranking approach was introduced in [2] with the visual similarities between the web pages. But, designed ranking method failed to improve the accuracy with a large number of pages on Web.

A Document Ranking method was developed in [3] to estimate and reorganizing the web pages. However, method failed to perform web information mining with higher accuracy. Multi-Layer Perceptron Neural Network was developed in [4] for web page rank estimation. Though the approach improves accuracy, the time consumption was not minimized.

A modified Page Rank algorithm was introduced in [5] based on the clustering concept and minimax similarity measure. The algorithm failed to use the efficient machine-learning algorithm to speed up the querying process. In order to improve the page ranking, a distinctive approach was introduced in [6] using search engine optimization. The designed approach has more time complexity for ranking the web pages. An incremental C-Rank algorithm was designed in [7] for effectively ranking the web pages with higher accuracy. Though the algorithm provides accurate as well as response for the dynamic web environment, the query preprocessing was not performed.

An entity ranking algorithm called NERank+ was developed in [8] using the graph model. But the accurate ranking was not carried out in a dynamic web environment. The problem of PageRank deviations was addressed in [9] using crawled web graphs. The machine learning techniques was not used to improve the page ranking performance. An effective page ranking method was introduced in [10] based on the sNorm(p). However, it was unsuccessful for reducing complexity of ranking the web pages.

1.1 contribution of the paper

The major contribution of JSPTRRPR is organized below,

- ❖ To improve page ranking accuracy in a dynamic web environment, a novel JSPTRRPR method is introduced with big data analytics. A JSPTRRPR method uses Tanimoto Robust Regressive Page Ranking method to analyze the list of words in the queries and the web page content. The proposed machine learning Regressive model easily predicts the user interested web pages using the Tanimoto similarity measure. Then the Borda count method is also applied to web pages. Thus, the user interested web pages are identified and correctly ranked. This, in turn, minimizes the false positive rate.

- ❖ The JSPTRRPR method uses the Jaccard index in the preprocessing phase to discard the stop words from the input user queries in the dynamic web environment. Followed by, the snowball stemmer is also applied for removing the stem words from the input queries. These two processes of JSPTRRPR method minimizes the web page ranking time and improves the accuracy.

1.2 structure of the paper

The article is summarized by: Section 2 describes literature review in web page ranking. Section 3 explains JSPTRRPR method using a clear diagram. Section 4 provides the simulation evaluation and dataset description with various performance metrics evaluation and experimental results are discussed. Section 5 concludes the proposed work and followed by the references are cited.

2. LITERATURE SURVEY

A Multilingual Information Search Algorithm was introduced in [11] for web page ranking based on preprocessing the queries. But the algorithm failed to accurately retrieve the more semantic equivalent document related to the query. A formal concept analysis was performed in [12] for semantic ranking of web pages but the analysis failed to use the efficient methods to compute the similarity. A Reinforcement Learning algorithm was introduced in [13] to web pages on queries. But algorithm did not perform any query preprocessing to minimize the ranking time.

A Similarity Preserving Snippet-Based representation was developed in [14] for searching the web to obtain the query results. But, the time taken to obtain the ranked query result was not reduced. The PageRank algorithm was introduced in [15] using an evolutionary algorithm. However, it was unsuccessful for reducing time complexity.

Relevance Vector (RV) Page Ranking Algorithm was designed in [16] for big data analytics. But, RV page ranking failed for enhancing performance of precision. Scalable location approach to web pages ranking was designed in [17] on query related URL click graph. The approach failed for performing query preprocessing to achieve higher accuracy.

Relevancy and keyword frequency-based approach were developed by [18] for ranking expensive as well as appropriate search outcomes. But, designed approach performs on text-based mining. GA on clustered web query was introduced in [19] for efficiently ranked the URLs. Though the algorithm performs the statistical analysis of precision it failed to analyze the error rate. TF-RFST was introduced by [20]. But the performance of ranking accuracy was not addressed.

The above-said limitations of existing methods are overcome by introducing a novel method called JSPTRRPR and a detailed description is presented in the next section.

3. Methodology

By number of data accessible on WWW, a variety of utilizations on Web is broadly considered such as web page classification and ranking, social network analysis etc. Among the variety of applications, this paper mainly focusing on ranking the web pages since the WWW comprises the billions of Web pages but these web pages did not assure user’s interest. They want not significant however expect web pages. Based on this motivation, the machine learning method called the JSPTRRPR method is applied that satisfies the user's interest of ranking used.

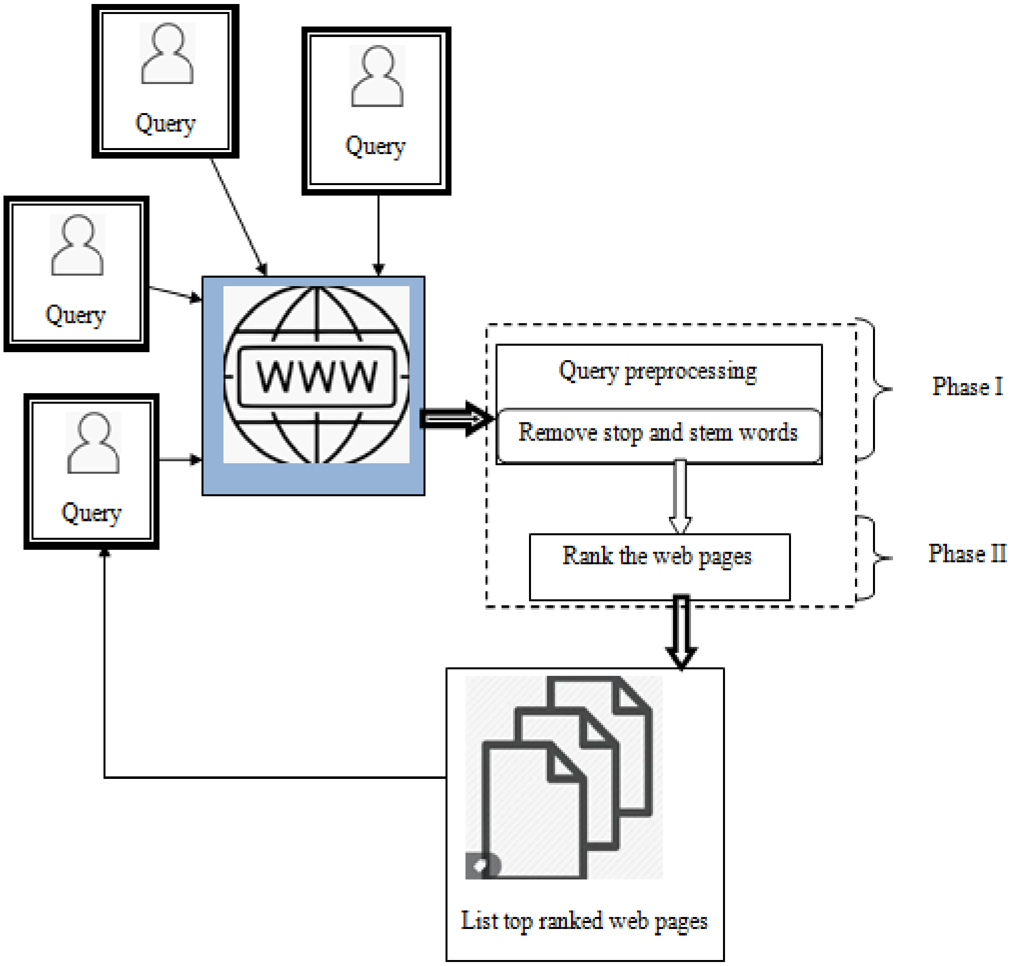


Figure 1 flow process of proposed JSPTRRPR method

Figure 1 indicates the flow process of the JSPTRRPR method to find the top-ranked web pages related to the user query with higher efficiency. The users sent their queries to WWW. The JSPTRRPR method comprises two phases namely query preprocessing and page ranking. The query preprocessing extracts the main words and remove the other words such as stop words and stem words. After preprocessing the query, the page ranking process is carried out by the means of Tanimoto Robust Regression Function. The regression-based page ranking algorithm used to display the results of user queries along with the page rank score of the web pages. These two phases of the JSPTRRPR method is described in the subsequent sections.

3.1 Phase I: Query preprocessing (minimize the ranking time)

The query preprocessing is the first phase of the JSPTRRPR method to minimize the web page ranking time. In the preprocessing phase, the stop as well as stem words were eliminated over input user. The stop word is generally a word like ‘the’, ‘a’, ‘an’, ‘in’, at, that, which, on,etc. Stem word is a part of a word used to which affixes are attached. The proposed method initially removes the stop words using the Jaccard index. Followed by the stem word removal process is carried out.

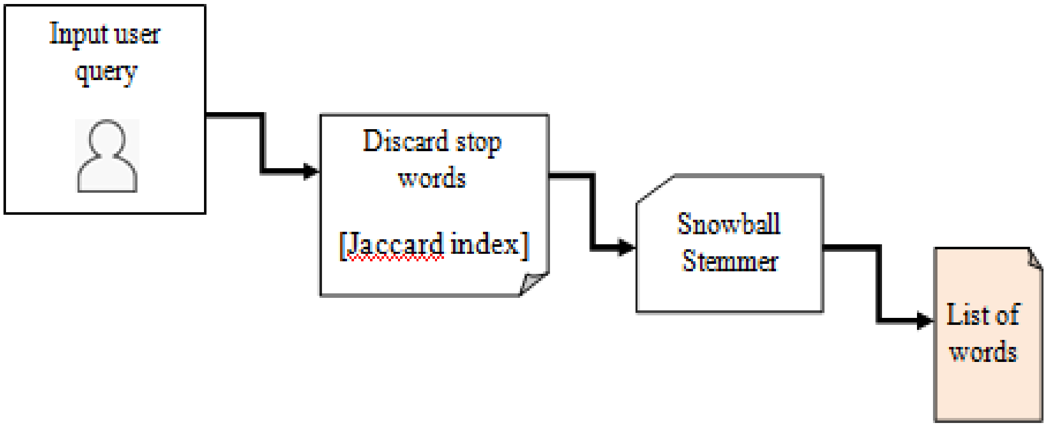


Figure 2 Query preprocessing

Figure 2 shows the flow diagram of query processing. Let us consider the number of user queries taken as input $Q_1, Q_2, Q_3, \ldots Q_n$. For each query, the stop and stem word removal is performed. The individual words $v_1, v_2, v_3, \ldots . v_m$are extracted from the query and it stored in the form of an array. In general, an array is the data structure that consists of a number of elements (i.e. words). For each word in the array is compared with the stopword directory (i.e. the list of stop words) where the collection stop words are stored. In order to compare the word in an array with the words in the directory, the Jaccard index is used to find the strop words. The Jaccard index is given below,

$$j_{(v_i,v_j)} = \frac{|v_i \cap v_j|}{|v_i \cup v_j|} \qquad (1)$$

Where, $j_{(v_i,v_j)}$ represents the Jaccard index, $v_i$indicates words stored in an array, v_j represents the words stored in the directory. In the above equation (1), the intersection symbol '⋂' indicates mutual independence between the two words which are statistically independent. The union symbol '⋃' represents the mutual dependence between the two words which are statistically dependent. The Jaccard index provides the output results in the range from 0 to 1 ($0 \leq \rho \leq 1$).

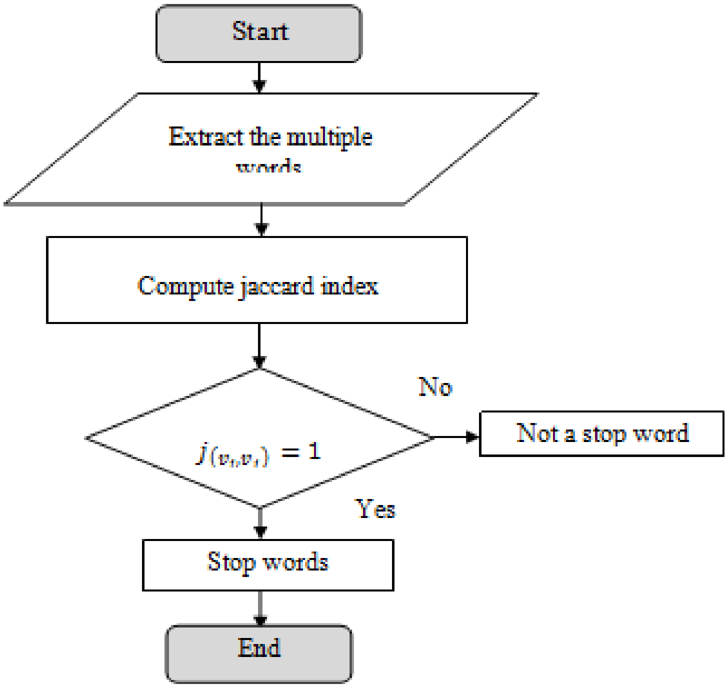


Figure 3 Flow chart of the Jaccard index based stop words identification

Figure 3 illustrates a flow process of stop word identification. The multiple words are extracted from the query. Calculate the Jaccard index for matching the words to identify the stop words. If the two words are correctly matched, then the word is said to stop words and it discards from the array. Otherwise, the word is not a stop word. After discarding the stop words, the Snowball Stemmer starts working on these words to find the stem words. In general, the stemming is a method used to minimize words from their root form, by removing original and inflectional affixes. Let us consider the words 'distributing', 'distributed', and 'distribution' given to the Snowball Stemmer.

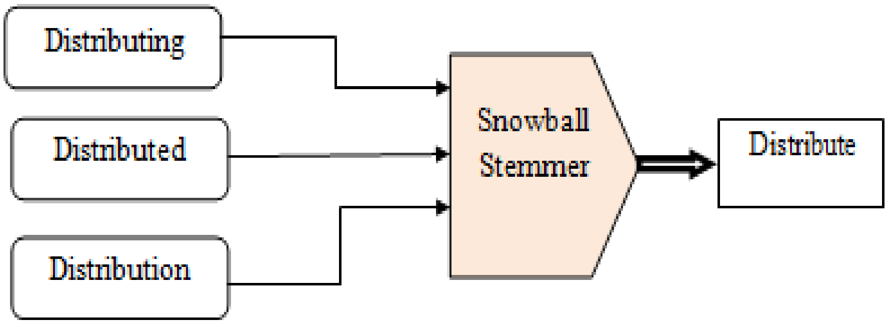
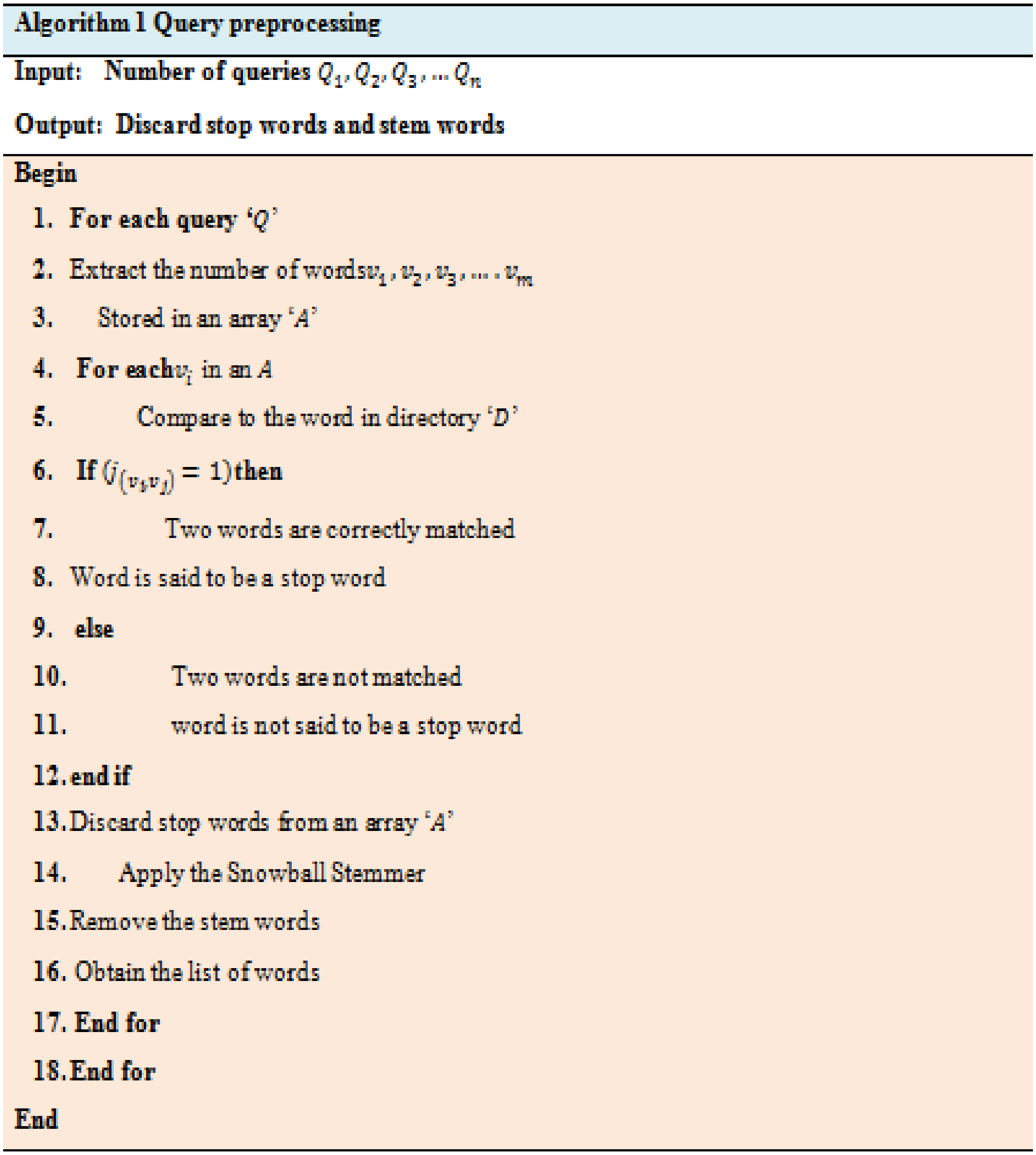


Figure 4 process of Snowball Stemmer

Figure 4 depicts the process of Snowball Stemmer to remove the suffixes i.e. 'ing', 'ion' 'ed' and obtain the root words i.e. distribute. In the big data query processing, these words cause more time to find the user interesting web pages. Therefore, the JSPTRRPR method performs the preprocessing to improve query processing with minimum time. The query processing algorithmic description is given below,



```
Algorithm 1 Query preprocessing
Input:  Number of queries $Q_1, Q_2, Q_3, \dots Q_n$
Output: Discard stop words and stem words
Begin
  For each query '$Q$'
  Extract the number of words $v_1, v_2, v_3, \dots . v_m$
     Stored in an array '$A$'
   For each $v_i$ in an $A$
         Compare to the word in directory '$D$'
   If $(J_{(v_i,v_j)} = 1)$ then
             Two words are correctly matched
  Word is said to be a stop word
   else
             Two words are not matched
             word is not said to be a stop word
  end if
  Discard stop words from an array '$A$'
        Apply the Snowball Stemmer
  Remove the stem words
  Obtain the list of words
  End for
  End for
End
```

3.2 Phase 2: Tanimoto Robust Regressive Page Ranking method (improve ranking accuracy)

After preprocessing, the page ranking is carried out to find the user interesting web pages with the list of words obtained from the preprocessing phase. The JSPTRRPR method uses the Tanimoto robust regression function to perform web page ranking. The Tanimoto robust regression is a machine learning technique used for measuring the relationships between two variables

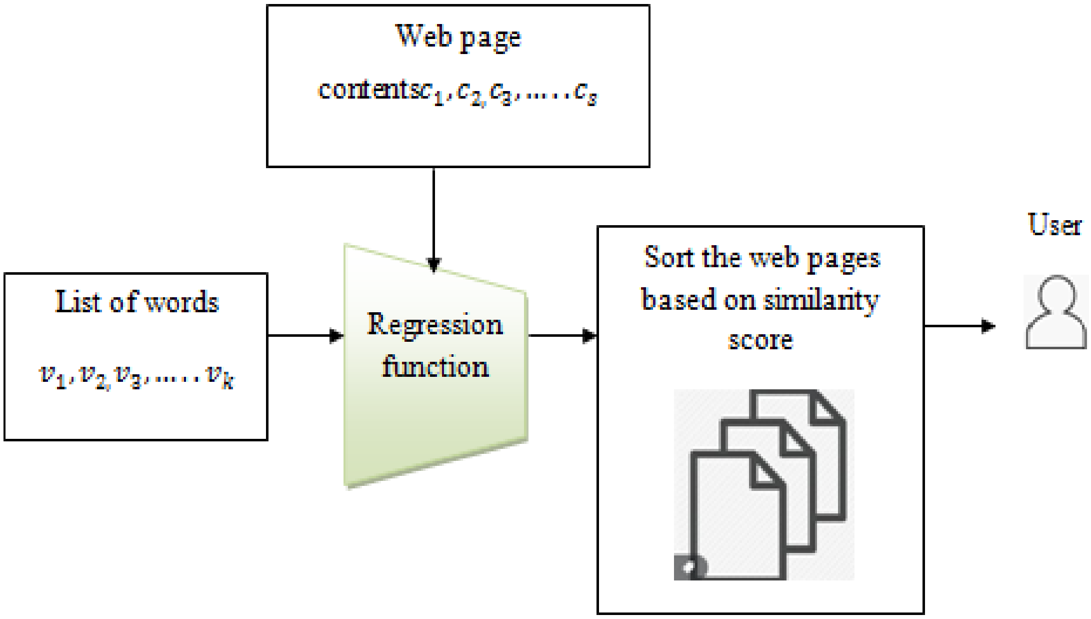

Web page
contents$c_1, c_{2,} c_3, \ldots\ldots c_s$
List of words
$v_1, v_{2,} v_3, \ldots\ldots v_k$
Regression function
Sort the web pages based on similarity score
User

Figure 5 Structure of Tanimoto Robust regression function

The Tanimoto Robust regression function is applied to identify the user interesting web pages as shown in figure 5. The regression function measures the relationship between the two variables (i.e. list of words obtained from phase-I) and the web page contents$c_1, c_{2,} c_3, \ldots\ldots c_s$. The Tanimoto similarity function is used to measure the relationship between the two variables. The regression analysis is performed using the following mathematical equations,

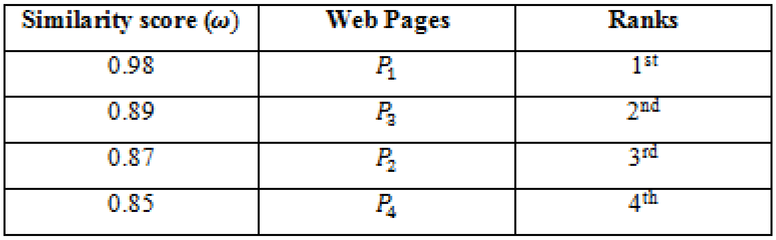
$$\omega = m * \left[\frac{\sum v_k c_s}{\sqrt{\sum {v_k}^2}+\sqrt{\sum {c_s}^2}-\sum v_k c_s}\right] \quad (2)$$

Where ω indicates Tanimoto similarity coefficient, m indicates the words appeared into a web page contents, $c_s$denotes a web page contents, $\sum v_k c_s$ indicates sum of product of paired score of v_k as well as c_s, $\sum {v_k}^2$ represents sum of squared score of v_k. $\sum {c_s}^2$ indicates sum of squared score of the c_s. Tanimoto similarity coefficient offers fractional score values of 0 to +1. Proposed technique uses the Borda count scheme for ranking web pages on similarity score value. By applying the Borda count scheme, the web pages were sorted into descending order to similarity fractional values as given below.

$$P_1 \geq P_3 \geq P_2 \geq P_4 \cdots \geq P_b \quad (3)$$

From (3), $P_1, P_2, P_3, \dots . P_b$ indicates the web pages. As a result, the Borda count scheme gives the first rank to the higher similarity value and second rank to their second most preferred similarity value, and so on. In other words, the high similarity score values are ranked top than the other web page contents. The top-ranked pages are said to be a user interesting web pages. Next, web pages were displayed to users. The example of Borda count based fractional ranking is shown in table 1.

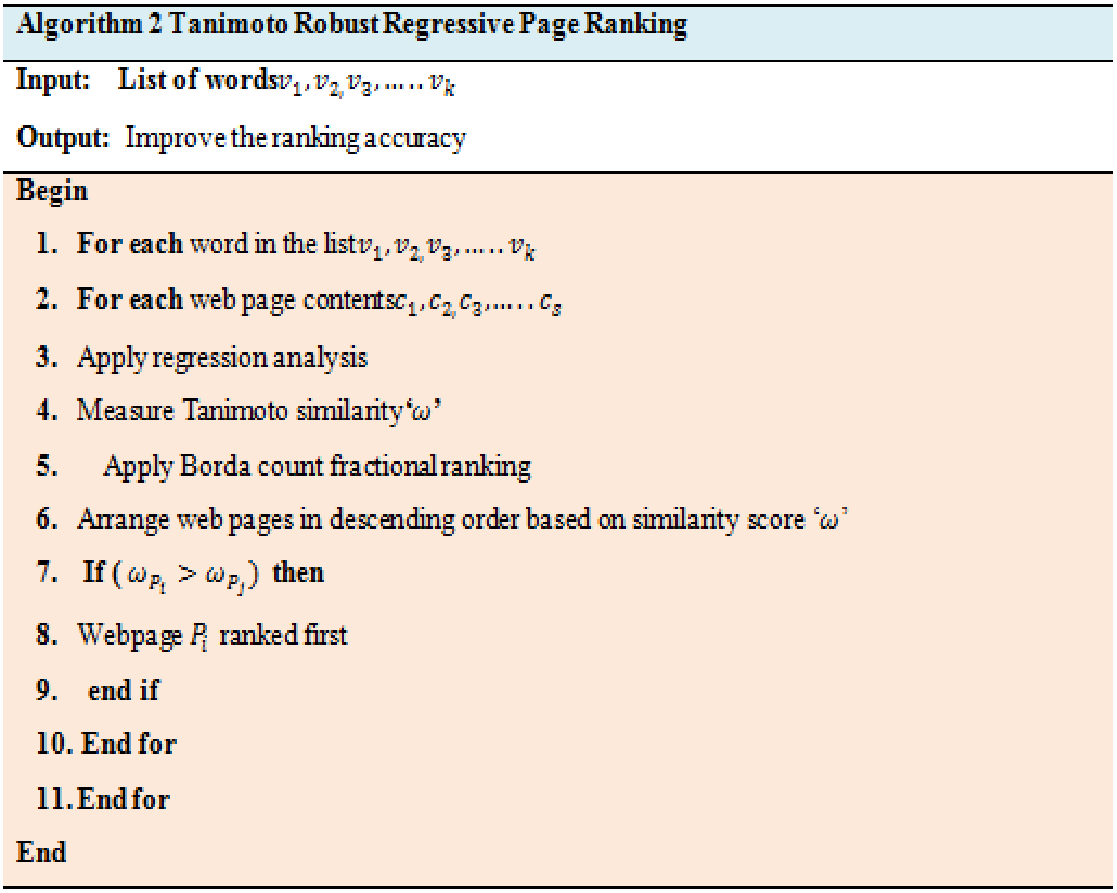
Table 1 Borda count fractional ranking



Similarity score (ω)	Web Pages	Ranks
0.98	P_1	1st
0.89	P_3	2nd
0.87	P_2	3rd
0.85	P_4	4th

Let us consider the example of Borda count fractional ranking based on the similarity score values. As shown in the above table 1, the web page 'P_1' has a higher similarity score than the other web pages and hence it ranked first, the web page $P_3$for being ranked second, the web page $P_2$for being ranked third and the web page $P_4$for being ranked last (i.e. fourth).

The algorithm 2 given below shows the step by step process of the web page ranking based Tanimoto robust regression-based Borda count fractional ranking method. The regression method measures the similarity between the list of words obtained from the preprocessing phase and the web page contents. Followed by, the pages are ranked according to their similarity score value. As shown in algorithm 2, ω_{P_i} denotes similarity of one web page and ω_{P_j} indicates similarity of another web page. In this way, proposed JSPTRRPR method estimates the rank of web pages with higher efficiency and lesser time consumption.



Algorithm 2 Tanimoto Robust Regressive Page Ranking

Input: List of words $v_1, v_2, v_3, \ldots\ldots v_k$

Output: Improve the ranking accuracy

```
Begin
  For each word in the list $v_1, v_2, v_3, \ldots\ldots v_k$
    For each web page contents $c_1, c_2, c_3, \ldots\ldots c_s$
      Apply regression analysis
      Measure Tanimoto similarity '$\omega$'
      Apply Borda count fractional ranking
      Arrange web pages in descending order based on similarity score '$\omega$'
      If ( $\omega_{P_i} > \omega_{P_j}$ ) then
        Webpage $P_i$ ranked first
      end if
    End for
  End for
End
```

4. EXPERIMENTAL SETUP

Experimental assessment of the proposed JSPTRRPR method and existing methods namely ACVPR [1] and ranking approach [2] are implemented in the java programming language. The integration of CACM dataset (http://ir.dcs.gla.ac.uk/resources/test_collections/cacm/) and Cranfield dataset http://ir.dcs.gla.ac.uk/resources/test_collections/cran/ are considered for retrieving the more relevant web pages. The input user queries are taken from these two datasets for webpage ranking.

4.1 Dataset description

CACM dataset consists of a collection of web-related documents, input queries and stops word lists in different files. Cacm.all file comprises the text of documents, the common_words file consists of stop words. Query.text file contains the 64 user queries.

Cranfield dataset includes two types of collections such as 1400 collection and 200 collections. This dataset includes (cran.all) documents of web pages, queries (cran.qry) and relevant assessments (i.e. cranqrel). The cran.qry comprises the 365 queries used as an input for ranking the web pages.

4.2 Evaluation metrics

The performance of JSPTRRPR and existing methods are analyzed with the different evaluation metrics as given below,

- Ranking accuracy
- False-positive rate
- Ranking time

Ranking accuracy: Ranking Accuracy (RA) is defined as the ratio of web pages are correctly ranked to entire number of user query. It is expressed by,

$$RA = \left(\frac{Q_n\ (P_{cr})}{Q_n}\right) * 100 \quad (4)$$

Where RA denotes a ranking accuracy, Q_n indicates number of queries, $Q_n\ (P_{cr})$ denotes the number of queries for which the web pages are correctly ranked. It is calculated by percentage (%).

False-positive rate: False positive rate (R_{FP}) was defined by proportion of web pages were wrongly ranked to entire number of queries taken as input. It is calculated by below equation,

$$R_{FP} = \left(\frac{Q_n\ (P_{Icr})}{Q_n}\right) * 100 \quad (5)$$

Where R_{FP} denotes a false positive rate, Q_n indicates number of user queries, $Q_n\ (P_{Icr})$ indicates a number of queries for which the web pages are incorrectly ranked. The false-positive rate was computed by percentage (%).

Ranking time: It is defined by the number of time consumed for web page ranking on user query. The ranking time is mathematically calculated as given below,

$$RT = Q_n * T\ (webpage\ ranking\ for\ one\ user\ query) \qquad (6)$$

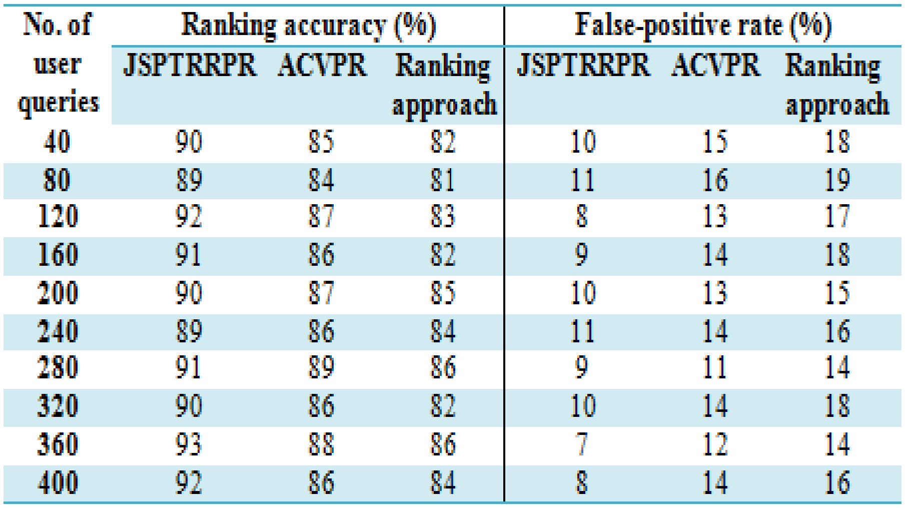
Where RT denotes a ranking time, 'Q_n' symbolizes the number of user queries, T indicates the amount of time taken to process one user query.

4.3 Comparative analyses under different metrics

The performance of proposed JSPTRRPR and existing methods ACVPR [1] and ranking approach [2] are analyzed. Comparative analysis was done based on the different evaluation metrics.

Initially, the web page ranking accuracy is measured with respect to the number of users queries taken from both CACM and Cranfield dataset. For the fair estimation, the numbers of user queries are taken in the range from 40 to 400. There are 10 different iterations are performed with respect to the number of queries.

Table 2 performance results of ranking accuracy and false-positive rate



No. of user queries	Ranking accuracy (%)			False-positive rate (%)		
	JSPTRRPR	ACVPR	Ranking approach	JSPTRRPR	ACVPR	Ranking approach
40	90	85	82	10	15	18
80	89	84	81	11	16	19
120	92	87	83	8	13	17
160	91	86	82	9	14	18
200	90	87	85	10	13	15
240	89	86	84	11	14	16
280	91	89	86	9	11	14
320	90	86	82	10	14	18
360	93	88	86	7	12	14
400	92	86	84	8	14	16

Initially, the web page ranking accuracy is measured with respect to the number of user's queries taken from both CACM and Cranfield dataset. The table values indicate the performance results of ranking accuracy and false positive rate of JSPTRRPR and two state-of-the-art methods ACVPR [1] and ranking approach [2]. The obtained results validate that the ranking accuracy is found to be improved and subsequently the false positive rate gets minimized using the JSPTRRPR method. The JSPTRRPR method uses the machine learning technique called a robust regression function to analyze the given user queries and identifies more interesting web pages using Tanimoto similarity coefficient values. Similarity was measured among number of words extracted from the queries and the web page content. The higher similarity values are ranked first than the other similarity values. Then the regression function uses the Borda count fractional ranking method to web pages on similarity values. Borda count method initially arranges the web pages in descending order according to their fractional similarity values. Then the ranks are assigned to web pages resulting it improves accuracy as well as reduces false positive rate. For each iteration, various input queries are taken as input and obtained different results. The accuracy of the JSPTRRPR method is compared with the accuracy of existing methods. Then the average ranking accuracy of the JSPTRRPR method is found to be improved by 5% as compared to ACVPR [1] and 9% compared to the ranking approach [2].

Similarly, the performances of false-positive rates are validated and the results are found to indicate the JSPTRRPR method obtains a lesser false positive rate as 31% as well as 43% compared with existing techniques [1] [2].

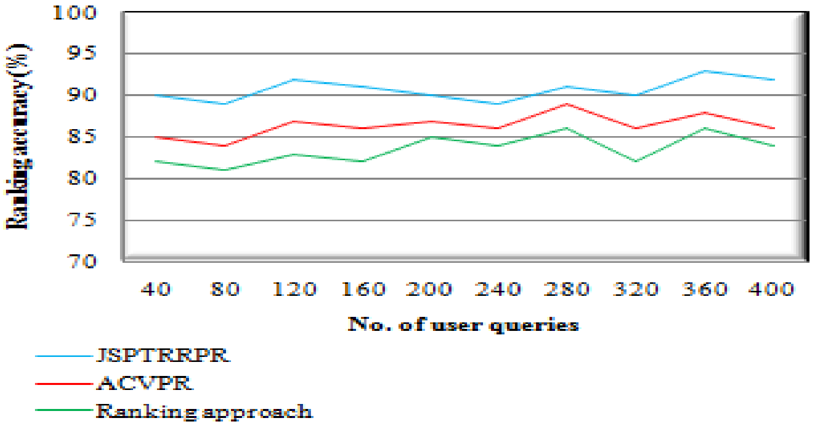


Figure 6comparative analysis of ranking accuracy

Figure 6 depicts the performance comparison of three methods, JSPTRRPR, ACVPR [1] and ranking approach [2]. The plots consist of three different lines of curves such as blue, red and green indicates the ranking accuracy of JSPTRRPR, ACVPR [1] and ranking approach [2] respectively. The number of queries is considered as input in the horizontal axis of the graph whereas the different results of accuracy obtained at the vertical axis. The graphical illustration is used to observe the positive contribution of the JSPTRRPR method when compared to the other two methods.

Figure 7 illustrates false-positive rate by number of user queries. False-positive rate performance of JSPTRRPR method was comparatively lesser than ACVPR [1] and ranking approach [2]. The reason behind that the JSPTRRPR method for reducing number of web pages is incorrectly ranked as compared to existing works.

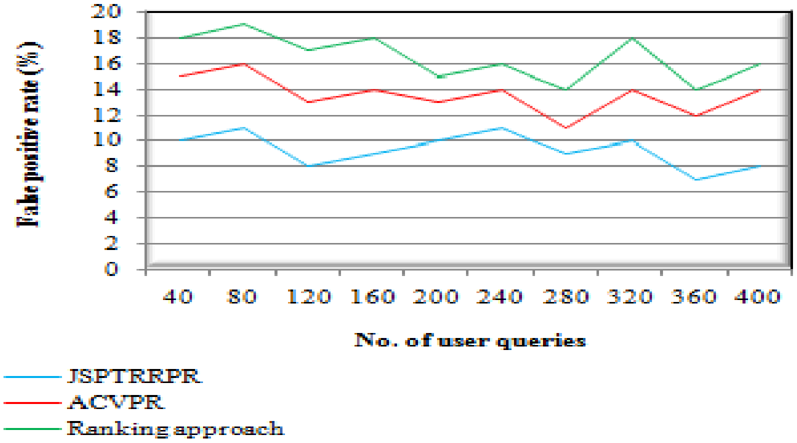


Figure 7comparative analysis of the false-positive rate

The final metric of the comparative analysis is the ranking time used to find the number of time consumed for ranking web pages to specific user queries. The comparative analysis of the ranking time of three methods is illustrated in the table.

Table 3 performance results of ranking time

No. of user queries	Ranking time (ms)		
	JSPTRRPR	ACVPR	Ranking approach
40	14	16	18
80	17	18	21
120	20	23	26
160	24	27	30
200	26	30	32
240	29	32	36
280	31	35	39
320	34	38	42
360	37	40	43
400	40	44	46

Table 3 illustrates the results of ranking time versus queries taken from 40-400. Experimental results show that the number of user queries is directionally proportional to the ranking time. In other words, while increasing the number of queries, time taken to rank the web pages also gets increased. But comparatively the JSPTRRPR method consumes lesser time. This is proved by the numerical example. With '40' queries taken as input, the JSPTRRPR method consumes '14ms' of time to web pages. Similarly, other two existing methods ACVPR [1] and ranking approach [2] consumes '16ms' and '18ms' of time with a similar count of user queries. There are ten different results of ranking time are obtained for three methods. The results are plotted in the graph as shown in figure 8.

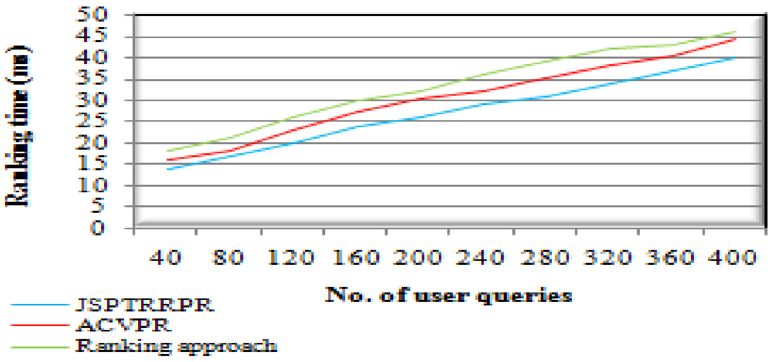


Figure 8 comparative analysis of ranking time

Figure 8 depicts the impact of ranking time by dissimilar number taken in the count from 40, 80, 120…400. From the figure, it is illustrative that the ranking time is in the increasing trend while increasing the number of queries. Besides, for experimentation, the 10 iterations are used and hence the fair comparison is said to be ensured for all the three different methods. The graphical results show that the ranking time is found to be minimized using the JSPTRRPR method when compared to [1] and [2]. This significant performance improvement is achieved by applying the query preprocessing. After receiving the input queries, the JSPTRRPR method discards the stop and stem words. The JSPTRRPR method uses the Jaccard index to compare the words in an array and words stored in the directory (i.e. a collection of stop words list). This helps to identify the stop words and it removes from an array. In addition, the snowball stemmer is also applied to remove the word stemming and obtain significant words from the query. With the obtained list of significant words, the ranking process is carried out resulting in it minimizes the time for ranking the web pages. These obtained results of the JSPTRRPR method are compared to other conventional ranking methods. As a result, the JSPTRRPR method minimizes the web page ranking time by10% and 19% when compared to the existing ACVPR [1] and ranking approach [2] respectively.

The discussed results of various metrics indicate that the JSPTRRPR method accurately performs the webpage ranking by considering the more user queries with minimum time and false-positive rate.

5. Conclusion

A novel method called JSPTRRPR is introduced for ranking the web pages by considering the dynamic nature of a query in a web environment. This JSPTRRPR method is having two different phases to handle the number of user queries in the dynamic web environment. The JSPTRRPR method first introduces the preprocessing phase before page ranking to solve more time consumption problems. With the preprocessed results, the web page is carried out for listing user interesting web pages. JSPTRRPR method uses the Tanimoto similarity coefficient which is more robust to solve the problem of accurate page ranking by the means of Borda count fractional approach. An experimental result on different metrics shows JSPTRRPR is better in terms of higher ranking accuracy as well as minimum false positive rate as well as ranking time.

REFERENCES

[1] Dheeraj Malhotra and O.P. Rishi, "IMSS-P: An intelligent approach to design & development of personalized meta search & page ranking system",Journal of King Saud University - Computer and Information Sciences, Elsevier, November 2018, Pages 1-16

[2] Ahmet Selman, Bozkir EbruAkcapinar Sezer, "Layout-based computation of web page similarity ranks",Elsevier, International Journal of Human-Computer Studies, Volume 110, Pages 95-114, February 2018

[3] Jaekwang Kim, "A Document Ranking Method with Query-Related Web Context",IEEE Access, Volume 7, 2019, Pages 150168-150174

[4] Hengameh Banaei and Ali Reza Honarvar, "Web page rank estimation in search engine based on SEO parameters using machine learning techniques", IJCSNS International Journal of Computer Science and Network Security, Volume 17, Issue 5, 2017, Pages 95-100

[5] Qidong Liu, Ruisheng Zhang, Xin Liu, Yunyun Liu, Zhili Zhao & Rongjing Hu, "A novel clustering algorithm based on PageRank and minimax similarity", Neural Computing and Applications, Springer, Volume 31, 2019, Pages 7769-7780

[6] M N A Khan & A Mahmood, "A distinctive approach to obtain higher page rank through search engine optimization, Sādhanā, Springer, Volume 43, 2018, Pages 1-12

[7]Jangwan Koo, Dong-Kyu Chae, Dong-Jin Kim, and Sang-Wook Kim, "Incremental C-Rank: An effective and efficient ranking algorithm for dynamic Web environments", Knowledge-Based Systems, Elsevier, Volume 176, 2019, Pages 147–158

[8] Chengyu Wang, Guomin Zhou, Xiaofeng He, and Aoying Zhou, "NERank+: a graph-based approach for entity ranking in document collections",Frontiers of Computer Science, Springer, Volume 12, 2018, Pages 504–517

[9] Helge Holzmann, Avishek Anand, and Megha Khosla, "Estimating PageRank deviations in crawled graphs", Applied Network Science, Springer, Volume 4, Issue 86, 2019, Pages 1-22

[10]Shubham Goel, Ravinder Kumar, Munish Kumar, Vikram Chopra, "An efficient page ranking approach based on vector norms using sNorm(p) algorithm", Information Processing and Management, Elsevier, Volume 56, 2019, Pages 1053–1066

[11]Vidya P V, Reghu Raj P C, Jayan V, "Web Page Ranking Using Multilingual Information Search Algorithm - A Novel Approach", Procedia Technology, Elsevier, Volume 24, 2016, Pages 1240 – 1247

[12]YaJun Du and YuFeng Hai, "Semantic ranking of web pages based on formal concept analysis", The Journal of Systems and Software, Elsevier, Volume 86, 2013), Pages 187-197

[13]Vali Derhami ∗, Elahe Khodadadian, Mohammad Ghasemzadeh, Ali Mohammad Zareh Bidoki, "Applying reinforcement learning for web pages ranking algorithms", Applied Soft Computing, Elsevier, Volume 13, 2013, Pages 1686–1692

[14] Erick Gomez-Nieto, Frizzi San Roman, Paulo Pagliosa, Wallace Casaca, Elias S. Helou, Maria Cristina F. de Oliveira, and Luis Gustavo Nonato"Similarity Preserving Snippet-Based Visualization of Web Search Results", IEEE Transactions on Visualization and Computer Graphics, Volume 20, Issue 3, 2014, Pages 457 – 470

[15]M. Coppola, J. Guo, E. Gill, and G. C. H. E. de Croon, "The PageRank algorithm as a method to optimize swarm behavior through local analysis", Swarm Intelligence, Springer, Volume 13, Issue 3–4, December 2019, Pages 277–319

[16]Dheeraj Malhotra and O. P. Rishi, "An intelligent approach to the design of E-Commerce metasearch and ranking system using next-generation big data analytics", Journal of King Saud University - Computer and Information Sciences, Elsevier, March 2018, Pages 1-12

[17]Yuening Hu, Changsung Kang, Jiliang Tang, Dawei Yin, and Yi Chang, "Large-scale Location Prediction for Web Pages", IEEE Transactions on Knowledge and Data Engineering, Volume 29, Issue 9, 2017, Pages 1902 - 1915

[18]Ms. Nilima V. Pardakhe, Prof. R. Keole, "Enhancement of Web Search Engine Results Using Keyword Frequency Based Ranking", International Journal of Computer Science and Mobile Computing, Volume 3, Issue.5, Pages 395-403, May- 2014

[19] Suruchi Chawla, "A novel approach of cluster-based optimal ranking of clicked URLsusing genetic algorithm for effective personalized web search", Applied Soft Computing, Elsevier, Volume 46, 2016, Pages 90-103

[20]R. Lakshmi and S. Baskar, "Novel term weighting schemes for document representation based on the ranking of terms and Fuzzy logic with semantic relationship of terms", Expert Systems with Applications, Elsevier, Volume 137, 2019, Pages 493-503