Volume 65, Issue 1, 2021
Journal of Scientific Research
Institute of Science,
Banaras Hindu University, Varanasi, India.

Webpage Recommendation for Organization Users via Collaborative Page Weight

Laxmi Rajani* and Urjita Thakar

Department of Computer Engineering, Shri Govindram Seksaria Institute of Technology and Science, Indore, India.
laxmirajani93@gmail.com*, urjita@rediffmail.com

Abstract: **Webpage Ranking is commonly used to cater the most relevant webpages to web-users for the given query. Traditional page ranking algorithms generate generic ranking for all users which may not result in most suitable ranking for different individual users. In this paper, a novel method is presented to make more personalized and effective web page recommendations. It uses Cumulative Page Weight (CPweight) by adding Google's PageRank, average Visit Count and average dwell time of organization's users. It is observed that more than 90 percent recommendations are matching to the user's requirements and effective as compare to the conventional approaches.**

Index Terms: **Clustering, Data-mining, Pagerank, Ranking Algorithm, Recommendations, Web-mining**

I. INTRODUCTION

Over the World Wide Web (WWW) enormous number of webpages exist which contain information corresponding to a variety of domains and areas. To search pages of their choice, users use search engines. To fetch the pages of end user's interest accurate Webpage recommendation is needed for search query. Webpage ranking makes recommendation process more productive by arranging the sequence of recommended URLs (webpages) in order of their importance. Web-mining is used to study and focus on uncovering the frequent data pattern from cyberspace (Kosala R, 2000; Mobasher et al., 2000) and help the web-users to reach their destination early and grab relevant knowledge.

Google's SEO use the PageRank algorithm to rank websites in search result. To estimate the importance of the webpage quantity and quality of in-links to a page are considered. Here the quality of in-links means the referrer page should also have higher PageRank value. The underlying assumption is that pages that receive a higher volume of links from quality pages, are more likely to be more important. PageRank was the original idea that got a lot of attention. Google PageRank has been evolved a lot in so many years. Google keep revising its ranking criteria and algorithms. The standard webpage ranking algorithm considers the web-structure, in-links volume and quality, to rank the webpage hits for the given query. But this ranking is very general for all the web-users all around the world.

The Weighted PageRank algorithm (WPR), an enhanced version of well-known PageRank algorithm (Xing W., 2004). A Weighted PageRank Algorithm sets greater rank values to more influential pages rather than distributing the rank value of a page equally among it's out-link pages. Weighted PageRank algorithm can find more relevant pages for the given query than the standard PageRank algorithm. Although WPR consider both in-links as well as out-links to calculate the page weight, still it is not able to personalise the rank at organization level as it does not consider the access pattern of the users of an organization.

In an organization, most users generally search for pages which are accessed by other users. Taking advantage of this observation, in this paper a method is proposed which is useful to present personalized rank to webpages to users in an organization. Apart from PageRank, visit count and time factor are used to obtain collaborative page weight. Consider an example, an institute keeps planning academic activities for students. And different students are browsing different webpages for the same topic using different query structure in their minds. Some find the interesting webpages, while others restructure the query and keep searching. The proposed method keeps track of webpage accessed by users of organization and based on their browsing history recommends the pages to other users of the institute. This will help new users to find the useful content at earliest.

* Corresponding Author

This paper is organized as follows: Section 2 briefs the related work. Section 3 presents methodology for CPweight calculation. Section 4 presents the result analysis of experiment and comparison between the classical PageRank and CPweight results. Section 5 concludes this paper.

II. RELATED WORK

In general, manuscripts may contain Title, Authors' names, Affiliation, E-mail address, Abstract, Keywords, Introduction, Literature Survey, Proposed Approach, Results and Discussion, Conclusion, Experimental Section, Acknowledgments, References and Endnotes. However, authors can organize the contents of the manuscript according to their requirements.

To filter relevant information, two widely used techniques by recommendation system are (i) Content based filtering method and (ii) Collaborative filtering method (Mobasher et al., 2000). Some researchers have also used hybrid of the two methods in their recommender systems (Nadi et al., 2011). After filtering the information for recommendation, the next task is to rank the items in filtered information and list them in descending order of their ranking. The idea is to bubble up the items with higher interest and importance at top positions. Web-structure mining techniques are exploited for search result optimization and ranking. Many research works have been done in the field of webpage recommendations and ranking. A multimedia PageRank algorithm is introduced (Chanda J, 2015) with some modification to support the searching of multimedia objects in the web. Hyperlinks are analysed for retrieving multimedia web objects, including webpages, images, and videos. The ranks are reworked by applying it in Learning Automata (LA) environment in (Bharti P.M, 2019). Learning automaton is the theoretical system, which acquire knowledge from the surroundings and acknowledge either with a bonus or a penalty. The aim is to downgrade the spam pages to improve the search engine correctness and pace, based on user behaviour evaluation which in turn saves the energy. A topic-aware Markov model is suggested in work (Zhao et al., 2018) for recommendation of topically coherent websites. This model learns users' navigation patterns and rank the pages based on both topical and temporal relevance. Furthermore, they make use of collaborative filtering architecture for customized webpage suggestions. An improvised item-based collaborative filtering recommender engine is presented in (Q. Yang et al., 2010) use the weighted k-mean clustering technique to cluster the movie's webpage. A web-usage mining in collaboration with web text mining is used in work (Smitha L, 2017) to improve the webpage access prediction. The clusters with higher similarity are identified for the active user followed by searching the most identical sessions in that clusters. Hybrid Similarity for information recommendation is presented in work (Wu & Davison, 2005). To establish relationship among objects the presented algorithm makes use of semantic similarity, which is based on whether two objects have weak or no link closeness in between. First, semantic similarity is calculated among the objects. Finally link similarity is computed to obtain the ultimate results. A crawler based dynamic system is introduced in (Yerma S., 2016) to rank the research paper using the account access time, number of citations and author details. An approach of automatic seed selection (training documents) is presented in (Sen et al., 2017) for topical and trust-based ranking. The links between the pages are reweighted for damping the effect of spam pages and a modified web-graph is prepared for rank calculation (Usha M., 2018). The paper (C. C. Yang, 2005), revises the classical PageRank algorithm using some additional elements like synonym, navigation, backlinks, and time. Two Phase Page Ranking (Forsati R., 2010), computes user interest based on content and usage of the pages. In phase one similar webpages are identified and in phase two, user interest score for a webpage is calculated using the stay time of that user on the page.

Ranking in SEO (Search Engine Optimization) refer to a position of website in the result set of search engine. There are many influencing factors that decides whether a website appears higher on the SERP (Search Engine Ranking Position). *PageRank* (Brin & Page, 1998) is the factor used by Google SEO to determine the best webpage for the given query. PageRank (PR) algorithm is coined by Google founders Larry Page and Sergey Brin, which measure the quality and quantity of links to a webpage to calculate the relative score of the webpage and rate them on 0 to 10 scale.

HITS algorithm (Kleinberg, 1999) is a link analysis algorithm invented by Kleinberg in 1997. HITS (Hyperspace Induced Topic Search) use the idea of Hubs and Authority to rate the webpages. A good hub is the page that point to many other good pages (authorities) and a good authority is the page that is referenced by many good pages (hubs). It considers in-links as well as out-links for webpage ranking and used successfully in the domain of web-structure mining. The final hub-authority scores of nodes are determined after infinite repetitions of the algorithm. And at the end these scores are normalized.

EigenRumor algorithm (Fujimura K., 2005) is used to find good blogs for a user in blogspace. Eigenvectors are used to calculate the score of new blogs entered in the system, which is based on weighting the hub and authority scores of the bloggers.

Time Rank algorithm (Hua Jiang et al., 2008) was proposed by H Jiang. To calculate the PageRank with more precision time element is used in this algorithm. The time a user stays on a webpage give estimation about the level of need to the client.

Distance Rank algorithm (Zareh Bidok, 2008) calculate the rank of the webpages using the distance between two pages. Here, distance between two webpages refers to the "average clicks between two webpages". Distance between pages is considered as punishment. The objective is to minimize punishment using the reinforcement learning.

III. PROPOSED METHOD

In this section the proposed method for calculation of new page rank is discussed. Ranking is done based on the page weight which is computed based on three factors – (i) Google PageRank, (ii) average visit count of the webpage, (iii) average dwell time of different users of a webpage. Architectural diagram of the proposed recommendation system is as given in Fig 1.

Webpage recommendations follow systematic steps to complete the task. At the very first step web logs are collected and preprocessed. After preprocessing useful patterns are identified with the help of appropriate web-mining techniques. These patterns are used for model training and query classification. And here we are ready with a knowledge base that can be used for recommendation.

At the time of live session query is submitted by the user. Using the trained model query is classified into the appropriate cluster of webpages. From there webpages relevant to the submitted query are identified. Page-Ranking algorithms are applied on the relevant pages and list of webpages is presented to the user in order of the ranking score.

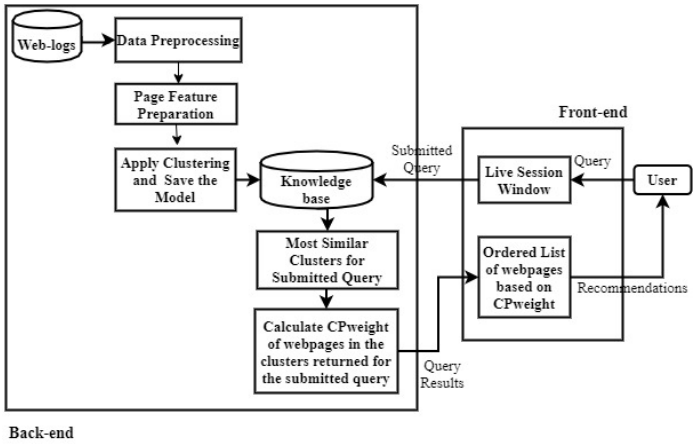

Fig. 1. Block Diagram for Proposed Webpage Recommendation System

The architecture is broadly divided into two phases; the Back-end phase and the Front-end phase. Various steps carried out in each phase are discussed next.

A. *Back-end Phase*

In back-end phase, the data is collected from various web-logs. Next, the data is pre-processed. In this step, URLs that need authentication for private accounts like facebook, gmail, github, udemy, google-drive, localhost etc. and duplicate URLs are removed from the browsing history dataset. The URLs available in the clean history files are scraped to collect the webpage content.

The raw text of webpage is cleaned to improve the feature quality by applying data preprocessing through following steps-

- all letters are converted to the specific case.
- words are reduced to its root word (lemmatization).
- stop words, punctuations, white spaces are removed.
- TfidfVectorizer is applied on the text for feature extraction (Kadhim et al., 2014).

In second step, similar webpages are clustered using k-means clustering (Blömer et al., 2016) applied on the features (keywords) collected in first step. The model is saved for recommendation. Only keywords are not useful enough. Therefore, to calculate the semantic relationship between keywords, probability theory has been used. The semantic connection between the webpages is determined.

To find the latent relation between keywords it is assumed that if two keywords are frequently and simultaneously appeared on the same webpage, then it shows semantic relevance between them. The similarity measure support can be used to find the extent of closeness of the two keywords. For given two keywords k_1 and k_2 , their support is computed as:

$$S(k_1,k_2) = \frac{N_{k_1 \cap k_2}}{N} \tag{1}$$

where, $N_{k_1 \cap k_2}$ denotes the number of pages in which $k_1$and k_2 appear together, and N denotes the total number of the webpages.

The normalized support $S(k_1,k_2)$is calculated as:

$$\overline{S}(k_1,k_2) = \frac{S(k_1,k_2) - S_{min}}{S_{max} - S_{min}} \tag{2}$$

where, S_{max} is the maximum support of all pairs of the keywords, and $S_{min}$is the minimum

Thus, the support between two keywords is computed. Next it is used to compute semantic similarity.

Consider two webpages w_1 and w_2. The keyword vectors that represent the webpages are $w_1 = \{k_{11}, k_{12}, \dots\dots, k_{1m},\}$ and $w_2 = \{k_{21}, k_{22}, \dots\dots, k_{2n}\}$ respectively. The top k extracted keywords represent the webpage to reduce the complexity. TF-IDF (Term Frequency-Inverse Document Frequency) is a popular method to find top k representative keywords of an object. Semantic similarity between pages w_1 and w_2 is calculated as:

$$sim(w_1,w_2) = \frac{\sum_{i=1}^{k} \bar{S}(k_{1i},w_2) + \sum_{j=1}^{k} \bar{S}(k_{2j},w_1)}{m+n} \tag{3}$$

where, $\overline{S}(k,w)$ is the semantic similarity between keyword k and webpage w. The $\overline{S}(k,w)$ is calculated as:

$$\overline{S}(k,w) = \max_{k_i \in w} \overline{S}(k,k_i) \tag{4}$$

where, $k_i \in w = \{k_1,k_2,\dots,k_n\}$ is the keyword vector of w.

In addition to the content-based clustering above approach is also used to cluster the semantically related webpages.

1) *Calculation of CPWeight*

Collaborative Page Weight (CPweight) distributes the page weight on different factors. Browsing history files of different users in an organization are used for weight calculation. The three

main factors of CPweight are:

1) Google's PageRank
2) Average visit count of the webpage.
3) Average Dwell time of the webpages

CPWeight is computed as follows:

$$CPweight = 0.5 * PR + 0.25 * Avc + 0.25 * DTW \quad (5)$$

where, PR ($PageRank$) is the relative weightage assigned to the webpage by google.

AVC ($Average\ Visit\ Count$) is the average of total visits to a webpage in given all browsing history profiles.

DTW ($Dwell\ Time\ Weight$) is computed as follows:

$$DTW = \begin{cases} 1, & ADT = ERT \pm \Delta \\ 0, & Otherwise \end{cases} \quad (6)$$

where, ADT ($Avg\ Dwell\ Time$) is the average of total time spent by all users on a webpage.

ERT ($Estimated\ Read\ Time$) is the ideal or approximate time the given webpage requires to read completely.

Δ is the small time period that can be adjusted to fine tune the recommendation results.

All browsers save the visit count and visit time of all URLs accessed on the system in a file. Similarly browsing history of all the users of an institute is captured at proxy server by configuring it to capture the user name, URL, visit time and visit count etc.

With the help of information obtained from browsing history of users, the average visit count and average dwell time of accessed URLs are calculated.

Average visit count is obtained by adding all the visit counts of accessed URLs by different users then dividing it with the number of users. For average dwell time, first the dwell time of URLs by different users is calculated. Dwell time of a URL is calculated by subtracting the visit time of the URL from the visit time of next accessed URL by the user. Average Dwell Time is obtained by adding the dwell time of a URL by all users and dividing it with number of users. For PageRank of a URL, open PageRank APIs available on Internet are used. Finally, the CPweight is calculated for webpages in the dataset using the computed parameters.

Next, front-end of the system is discussed.

B. Front-end Phase

In front-end a user interface (UI) is provided to the user to submit a search query. The most similar clusters for the searched query has been identified on the fly. Each webpage in the identified similar clusters is associated with a CPweight score. This CPweight signifies the importance of webpages, specifically for users of the organization whose browsing history files are used to calculate the CPweight parameter. Traditionally, the pages with higher PageRank value are recommended as a query result, which is quite general for all web-users. However, a user may find some pages with low PageRank value more interesting due to its content and correlation with current scenario in surrounding.

Some pages may have lesser PageRank value as compared to the other page in the list of relevant pages, but accessed frequently within the intranet of the organization. These are considered to be logically useful for the users of that organization. The standard PageRank algorithm cannot personalize the webpage ranking at organization level. Hence CPweight is used to filter out logically high weighted webpages specific for the organization.

It is assumed that the webpage *p* is logically more important for the query of a user if all the other users in his institute/company click the page *p* comparatively more times than the other pages and spent approximately equivalent time on the page that the page ideally required to read. Hence, browsing history of a user is a key for determining the user's interest. For personalized recommendation, the browsing history of users has been used for re-ranking the webpages.

2) Recommendations:

In recommendation phase a set of top *n* most similar pages of query are returned to the user and the results are arranged in descending order of their CPweight. These recommended webpages have captured more attention of institute's users, which is confirmed by their browsing habits.

IV. EXPERIMENTAL EVALUATION

To evaluate performance of the proposed model, the data is collected form proxy servers and browsing history of organization's systems. The feature/properties used in datasets are as given in Table I:

Table I. Properties of the Dataset

URL	Title	Visited On	Visit Count	Referrer
https://matplotlib.org/tutorials/index.html	Tutorials – Matplotlib 3.1.1 documentation	15-10-2019 10:08:35	2	https://matplotlib.org/
https://matplotlib.org/tutorials/index.html	Tutorials – Matplotlib 3.1.1 documentation	15-10-2019 10:56:35	2	https://matplotlib.org/

Using well known web-scraping technique, all URLs in datasets are scraped to enrich the dataset. These page features contribute for better and more accurate webpage clustering. The scraped dataset contains following attributes to make clusters: Title, Meta-Keywords, Meta-description, Headings, First paragraph, Alt-tag. Preprocessing of dataset was done for further processing. Wrong and duplicate URLs were dropped; stop words and punctuations were removed. Top representative keywords were extracted from the webpage.

For each URL average visit count, average dwell time and estimated reading time was calculated. Using these parameters CPweight was calculated. Finally, all webpages/URLs were

clustered using the features extracted in previous step.

Table II. Properties of the Dataset

UID	Url	Estimated Reading Time	Avg Dwell Time	Avg Visit Count	File Count	Page Rank	CPweight	Cluster ID
1	https://www.w3schools.com/PHP/php_intro.asp	946	450	2	2	7.45	4.23	37
2	https://www.w3schools.com/PHP/php_comments.asp	906	562	2	2	7.45	4.23	37
3	https://www.w3schools.com/PHP/DEfaULT.asP	949	800	2	2	7.45	4.48	37
4	https://www.tutorialspoint.com/php/index.htm	328	259	9	4	5.66	5.33	37
5	https://www.tutorialspoint.com/php/php_decision_making.htm	402	436	7	3	5.66	4.83	37
6	https://www.tutorialspoint.com/php/php_error_handling.htm	593	745	8	4	5.66	5.08	37
7	https://docs.python.org/3/tutorial/interpreter.html#the-interpreter-and-its-environment	409	536	5	3	6.33	4.67	89
8	https://docs.python.org/3/tutorial/interpreter.html#invoking-the-interpreter	409	535	3	3	6.33	4.17	89
9	https://docs.python.org/3/tutorial/	386	2546	5	3	6.33	4.42	89
10	https://www.w3schools.com/python	518	83	3	3	7.45	4.48	89
11	https://www.w3schools.com/python/python_intro.asp	501	323	2	3	7.45	4.48	89
12	https://www.w3schools.com/python/python_while_loops.asp	474	33.6	1	2	7.45	3.98	89
13	https://www.tutorialspoint.com/computer_fundamentals/computer_overview.htm	329	68	3	2	5.66	3.58	6
14	https://www.tutorialspoint.com/computer_fundamentals/computer_cpu.htm	272	323	3	2	5.66	3.83	6
15	https://www.tutorialspoint.com/compiler_design/compiler_design_types_of_parsing.htm	190	485	3	2	5.66	3.58	6
16	https://www.tutorialspoint.com/compiler_design/compiler_design_regular_expressions.htm	311	376	3	2	5.66	3.83	6
17	https://www.tutorialspoint.com/compiler_design/compiler_design_phases_of_compiler.htm	389	128	3	2	5.66	3.58	6
18	https://www.javatpoint.com/how-does-cloud-computing-work	200	203	9	4	4.35	4.68	52
19	https://www.javatpoint.com/history-of-cloud-computing	293	3385	3	2	4.35	2.93	52
20	https://www.javatpoint.com/cloud-computing-vs-grid-computing	200	329	2	2	4.35	2.93	52
21	https://www.javatpoint.com/advantages-and-disadvantages-of-cloud-computing	361	389	6	3	4.35	3.93	52

In Table II the PageRank and calculated Collaborative Page Weight of each URL is mentioned in its corresponding row. Cluster ID is the cluster assigned to the corresponding URL. All URLs with same cluster-ID represent the cluster of similar webpages. When the query is fired to the model, then by using similarity measure the most similar clusters are identified from all the available clusters. The top *n* results are displayed to the user.

In Table II, Cluster-ID, PageRank, and computed CPweight for the corresponding webpages are presented. The relative difference between the PageRank and CPweight for each URL can be analyzed from this table.

Consider the webpages of cluster 37, according to PageRank attribute, the sequence of presented URLs should be 1, 2, 3, 4, 5, 6 as the URLs 1,2 and 3 have higher PageRank (7.45) and URLs 4,5 and 6 have lesser PageRank (5.66). But it has been observed that URLs 4, 5 and 6 are accessed comparatively more often than the URLs 1, 2 and 3. Also in terms of dwell time, the average time spent by all users on webpage 4, 5 and 6 is approximately same as the estimated time required to read these pages completely. This signifies that these pages are found interesting by the users within the intranet of an organization. Therefore, according to the computed CPweight, the sequence of URLs presented should be 4, 5, 6, 1, 2, 3. Similarly while considering PageRank for cluster 89, webpage sequence should be 10, 11, 12, 7, 8, 9. And based on their CPweight, the webpage sequence should be 10, 11, 7, 9, 8, 12.

From these observations it is clear that it is more productive for an organization to re-rank the webpages within the intranet, especially when there are thousands of webpages in a cluster. Using traditional PageRank, the results are very generic for all users. But when the URL weightage is calculated based on the browsing history of an organization's users; then this page weight is personalized for the users of that organization.

The graph in Fig. 2, illustrates the proportional increase in CPweight as visit count increases with time. The y-axis shows the visit count and the x-axis shows the period of time (in this case days from 20-May-2019 to 16-Jun-2019) of the study. The dashed lines represent different URLs. Logarithm is taken to visualize the minute difference between the values.

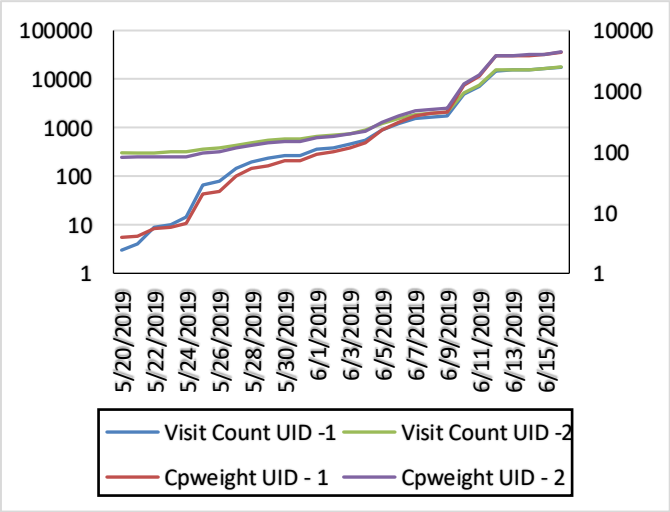

Fig 2. CPweight v/s Visit Count

The URL with URL-ID 1 start with visit count 3 on 20-May-

2019 upto 16-Jun-2019 the visit count reached to 17458, hence the CPweight also increased from very small value 3.91 to very large value 4367.66.

However, the URL with URL-ID 2 started with visit count 300 on 20-May-2019 but upto 16-Jun-2019, the visit count reached to 350 only, hence the CPweight also increased from 77.42 to 89.92 only.

Fig. 3 illustrate the User Interface for webpage recommender system. The search result presents the URLs for the given query "cloud services". At first step similar clusters are identified for the search query. Next, the webpages identified in most similar clusters are arranged in descending order of their CPweight.

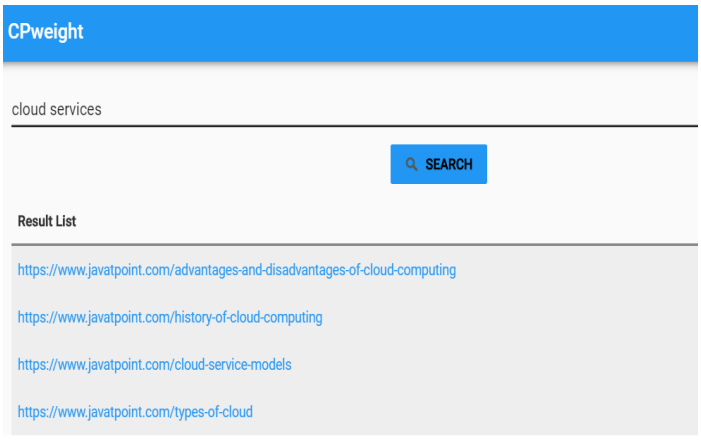

Fig 3. User Interface for webpage recommendation

CONCLUSION

In this paper, a novel method is presented to offer interesting webpages to the users of an organization. It uses Collaborative Page Weight (CPweight) by adding Google's PageRank, average Visit Count and average dwell time of organization's users. It enables generation of most suitable ranking for users of a specific organization. Earlier methods offer pages based on generic rank to all the users. It is observed that more than 90 percent recommendations match to the user's requirements and is effective as compared to the conventional approaches though it takes some computational time to rank and display the URLs to the users. Context specific recommendations is a useful feature of this system, which offers the webpages that are closely related to the user's intension. For example, in context of general English IDK means "I don't Know", but in context of computer science IDK refers to the "Integrated Development Kit", so the frequent access to the webpages in a specific context make the recommendations accurate as visit count is used to calculate the CPweight. The problem of cold start may arise, if the searched query does not match to any of the cluster available in the dataset in-hand.

In future work, auto-correction and auto-completion features can be added to the proposed model. Query can be redirected to the internet to handle the cold start problem, and in such case the presented search results will be ordered by PageRank. Also, this model can be used as add-on for any web-browser.

REFERENCES

Bharti, P. M., & Raval, T. J (2019). Improving Web Page Access Prediction using Web Usage Mining and Web Content Mining. *2019 3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. https://doi.org/10.1109/iceca.2019.8821950Qin

Blömer, J., Lammersen, C., Schmidt, M., & Sohler, C. (2016). Theoretical Analysis of the k-Means Algorithm – A Survey. *Algorithm Engineering,* 9220, 81–116. https://doi.org/10.1007/978-3-319-49487-6_3

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30(1–7), 107–117. https://doi.org/10.1016/s0169-7552(98)00110-x

Chanda, J., & Annappa, B. (2015). An Improved Web Page Recommendation System Using Partitioning and Web Usage Mining. *Proceedings of the International Conference on Intelligent Information Processing, Security and Advanced Communication*. https://doi.org/10.1145/2816839.2816910

Forsati, R., & Meybodi, M. R. (2010). Effective page recommendation algorithms based on distributed learning automata and weighted association rules. *Expert Systems with Applications,* 37(2), 1316–1330. https://doi.org/10.1016/j.eswa.2009.06.010

Fujimura, K., Inoue, T., & Sugisaki, M. (2005). The EigenRumor Algorithm for Ranking Blogs. Chiba, Japan.

Hua Jiang, Yong-Xing Ge, Dan Zuo, & Bing Han. (2008). TimeRank: A method of improving ranking scores by visited time. *2008 International Conference on Machine Learning and Cybernetics,* 1654–1657. https://doi.org/10.1109/icmlc.2008.4620671

Kadhim, A. I., Cheah, Y.-N., & Ahamed, N. H. (2014). Text Document Preprocessing and Dimension Reduction Techniques for Text Document Clustering. *2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology*, 69–73. https://doi.org/10.1109/icaiet.2014.21

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM,* 46(5), 604–632. https://doi.org/10.1145/324133.324140

Kosala, R., & Blockeel, H. (2000). Web mining research: A survey. *ACM Sigkdd Explorations Newsletter*, 2(1), 1–15. https://doi.org/10.1145/360402.360406.

Mobasher, B., Cooley, R., & Srivastava, J. (2000). Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8), 142–151. https://doi.org/10.1145/345124.345169

Nadi, S., Saraee, M. H., & Bagheri, A. (2011). A Hybrid Recommender System for Dynamic Web Users. *International Journal Multimedia and Image Processing*, 1(1/2), 3–8. https://doi.org/10.20533/ijmip.2042.4647.2011.0001

Sen, T., Chaudhary, D. K., & Choudhury, T. (2017). Modified Page Rank Algorithm: Efficient Version of Simple Page Rank with Time, Navigation and Synonym Factor. *2017 3rd International Conference on Computational Intelligence and Networks (CINE),* 879–882. https://doi.org/10.1109/cine.2017.24

Smitha, L., & Fatima, S. S. (2017). Topical and Trust Based Page Ranking Using Automatic Seed Selection. *2017 IEEE 7th International Advance Computing Conference (IACC)*, 803–806. https://doi.org/10.1109/iacc.2017.0165

Usha, M., & Nagadeepa, N. (2018). Combined two phase page ranking algorithm for sequencing the web pages. *2018 2nd International Conference on Inventive Systems and Control (ICISC),* 876–880. https://doi.org/10.1109/icisc.2018.8398925

Wu, B., & Davison, B. D. (2005). Identifying link farm spam pages. *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web - WWW '05*. https://doi.org/10.1145/1062745.1062762

Xing, W., & Ghorbani, A. (2004). Weighted PageRank algorithm. Proceedings. Second Annual Conference on Communication Networks and Services Research, 2004., 1, 305–314. https://doi.org/10.1109/dnsr.2004.1344743

Yang, C. C., & Chan, K. Y. (2005). Retrieving multimedia web objects based on PageRank algorithm. *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web - WWW '05,* 906–907. https://doi.org/10.1145/1062745.1062791

Yang, Q., Fan, J., Wang, J., & Zhou, L. (2010). Personalizing Web Page Recommendation via Collaborative Filtering and Topic-Aware Markov Model. *2010 IEEE International Conference on Data Mining*, 1145–1150. https://doi.org/10.1109/icdm.2010.28

Yerma, S., & Majhvar, A. K. (2016). Updated page rank of dynamically generated research authors' pages: A new idea. *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT).* https://doi.org/10.1109/rteict.2016.7807954

Zareh Bidoki, A. M., & Yazdani, N. (2008). DistanceRank: An intelligent ranking algorithm for web pages. *Information Processing & Management,* 44(2), 877–892. https://doi.org/10.1016/j.ipm.2007.06.004

Zhao, Q., Wang, C., Wang, P., Zhou, M. C., & Jiang, C. (2018). A Novel Method on Information Recommendation via Hybrid Similarity. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(3), 448–459. https://doi.org/10.1109/tsmc.2016.2633573

***