VESTNIK UDMURTSKOGO UNIVERSITETA. MATEMATIKA. MEKHANIKA. KOMP’UTERNYE NAUKI
COMPUTER SCIENCE 2021. Vol. 31. Issue 1. Pp. 116–131.

MSC2020: 68T10, 68U35

© *N. M. Ali, A. M. Gadallah, H. A. Hefny, B. A. Novikov*

ONLINE WEB NAVIGATION ASSISTANT

The problem of finding relevant data while searching the internet represents a big challenge for web users due to the enormous amounts of available information on the web. These difficulties are related to the well-known problem of information overload. In this work, we propose an online web assistant called OWNA. We developed a fully integrated framework for making recommendations in real-time based on web usage mining techniques. Our work starts with preparing raw data, then extracting useful information that helps build a knowledge base as well as assigns a specific weight for certain factors. The experiments show the advantages of the proposed model against alternative approaches.

Keywords: web mining, web personalization, link prediction, web usage mining, recommender systems, web log, web navigation assistant.

DOI: 10.35634/vm210109

Introduction

The Internet represents the primary source that users rely on for information searching and gathering. The volume of information available on the Internet is rapidly growing due to the ongoing expansion of the World Wide Web and tremendous amounts of applications. So, the information overload problem represents the biggest challenge facing users during web searching. This situation occurs when people are flooded with lots of information and services’ options. Therefore, obtaining more “relevant” or “interesting” information becomes difficult for users [1].

This work aims to help web users to access the relevant resource while surfing the web by providing them with a list of links that users most likely will explore [2]. The authors investigate this goal through modeling web users’ behavior [3]. Our model exploits the existence of web usage data (e. g., log file) to compute a rank of web pages that more consistent with the user’s behavior. This paper involves the evaluation of our model in comparison with other models.

Unlike our previous model [4], the new work has a guarantee to support the user in all cases, even if there is no similar historical behavior concerning the current user. On the other hand, the process of data preparation comprises the collection of numerous similar patterns. Also, it involves the computation of the relative weight for each one. This new feature is referred to the rate of repetition, which helps to reduce the time of providing users with advice.

The organization of the rest of this paper is as follows: A brief discussion of recommender systems and their categories in § 1 is followed by an overview of previous and related works in § 2. A demonstration of our approach and an illustrative case study are explained in § 3 and § 4. Finally, the conclusion summarizes the results and outlines the future work.

§ 1. Recommender Systems

Generally, web recommender systems guide users towards the more interesting or useful objects in a large area of possible options. There is a wide range of web recommender systems. Most of them are different on the base of work (e. g., web content mining, web usage mining, etc.). In this work, we are interested in those systems whose work is based on web usage mining because of the great importance of the web usage data. Link prediction is mainly relying on the navigation behaviors and the explicit feedback provided by users' ratings on different items [5–7].

Commonly, web recommender systems (RS) aim to predict the user's intentions, and the needs of data to facilitate and customize their online experience. They try to help users by providing them with a list of suggested online web resources (e. g., Uniform Resource Locator: URL) [8].

Therefore, there is a need to build recommender systems; to help users while surfing the web for quick access to web resources mostly related to the search topic rather than wasting time surfing irrelevant issues. Recently, there is an increasing interest in the development of recommender systems by applying web usage mining techniques [9–11]. The following subsections present some of such techniques.

1.1. Statistical Techniques

The most common mining technique for a sequential pattern used for web recommendation is Markov Model (MM) [12]. Generally, considering consecutive and sequential accessed pages could achieve good prediction accuracy. Commonly, the number of deemed pages in the weblog file entries determines the order of the model. It varies from Lower-Order to Higher-Order. Regularly, the lower-order Markov model provides high coverage, but with low accuracy.

On the contrary, the higher-order Markov model gives little coverage but high efficiency with more time complexity [13]. In the first order's Markov model, each state represents a single web page, and a state transition is matched by a pair of visited pages. So, there exist two artificial states; a single web page represents "start and final" each one. On the other hand, in the second-order Markov model, each state corresponds to a sequence of two visited web pages, and so on [14].

1.2. Data-Mining Techniques

Commonly, data-mining techniques work on extracting implicit and potentially useful navigational patterns. Using some data mining techniques such as association rules mining and clustering represent good alternatives. Such techniques discover the user's preferences from their implicit feedbacks (e. g., web pages already visited).

Several approaches, like clustering and collaborative filtering, can be integrated with weight-based methods for web pages recommendation. Commonly, web page weights may be binary or non-binary. Binary weight is used for computing efficiency [14], it represents in the page view the existence or nonexistence of action in the transaction like product-purchase, or document access.

On the other hand, non-binary weight is represented as a function of the duration of the associated page view in the user's session [15]. Moreover, using association rule (AR) mining could achieve an improvement in the recommendation's accuracy. Also, it has the advantage of the facility in scaling too large data sets. Incorporate page weight into AR models has not been explored yet in previous studies. Weighted association rule (WAR) mining represents an improvement of the classical AR model, which allows the assignment of different weights to different evaluation criteria [16, 17].

§ 2. Background

Generally, many research works attempt to study and model the behaviors of web users that have emerged in recent times [18]. However, most of these works are hindered by some limitations [19, 20]. Many researchers proposed different combinations of mining approaches toward web access recommendation. Commonly, some of these approaches are based on the Markov model, which is considered the widest one used to model user's web navigation.

Bhushan and Nath (2013), introduced a new model based on learning from weblogs. It provides users with a list of recommended web pages more relevant to their intentions, considering the user's historical behavior. Subsequently, an evaluation operation takes place to rank the list of search results. The failure of dealing with pages that have zero visits and the behavior of the new users remain the most critical defects in this model. This work solves a similar problem, but our work differs in the mechanism of ranking web pages. Regarding current user behavior; we use historically similar sessions to ranking pages, while they provide an absolute rank to pages ignoring existing behavior.

Based on the Markov model, another model for web access prediction was introduced by [22]. However, considering all-access sequences throughout the prediction process increases the complexity that is regarded as the major drawback of this model.

Another approach introduced in [23] respects the information content semantics and interests of the user; it proposes a recommendation approach that works at the content level, and recommendations are made across different categories. Linked data used as an underlying structure for the content's semantics, it considered as the source of finding relevant results. After measuring the similarity and relevance between retrieved data, it is then grouped to form a cluster. Finally, the most exciting content is recommended to general users based on the content consumption trends monitored by user groups. These contents are characterized as the most prominent, proactive, and often consumed. Defectives of this work consisting in the high calculations used are not commensurate with achieved accuracy.

In 2006 [24] the authors proposed a new model using an integration between clustering and sequential pattern mining. This model uses rough sets clustering and all kth order Markov model. Low prediction accuracy is the major drawback of this model due to approximation while forming clusters. Prediction accuracy is affected by cluster tightness, which is due to the relative relation between object and cluster.

In Khalil, F. et al. (2006), an integration between the Markov model and association rules was introduced for predicting web page access. The lower order of the Markov model is the base of such a combination. On the other hand, to resolve ambiguous predictions, a sub-sequence set of association rules is used to complement the Markov model using long history data.

Using the base of the majority intelligence technique [26], the authors introduced a heuristic approach which is able to adapt to changes of the navigational patterns easily. It provides users with recommendations at a low cost during web surfing. In an unidentified environment, the proposed technique tries to mimic human behavior. This approach works perfectly in real-time with reasonable accuracy in the case of web surfing by several users in parallel. This paper solves a similar problem, but our work differs in working at a single page level, which seems more suitable for users instead of page category level.

On the other hand, a new approach to predict user browsing behavior was introduced in [27]. It works at two levels to meet the nature of the navigation. The category stage is the first level, and the second level is concerned with the web page stage; the early-stage is concerned with predicting the category under interest. Consequently, unnecessary categories were excluded. The scope of calculation is massively reduced. Next, pruned Markov models with higher-order are used in the second level to predict the users browsing pages.

Another new approach has been introduced by [28], for the next page access prediction. The authors proposed a hybrid approach, which combines the integration of the Markov model with clustering-based pairwise nearest neighbor technique. The application of the proposed model has improved the quality of the resulting patterns and reduced the size of the data used in the sequential mining process significantly. Ignoring the loosely connected sequences of web access in the mining process is the major drawback of this work.

Additionally, [29] proposed a novel web recommendation system called WebPUM. It predicts the near future intentions for web users online based on classifying user's navigation patterns. To accomplish this task, they developed a new system using the application of web usage mining techniques. They use the new graph-partitioning algorithm in the first phase for modeling of user's navigation patterns. Moreover, classifying current user activities is based on using the longest common subsequence algorithm to predict the next user's action. This work was impeded by its moderate accuracy in spite of highly complex computations.

An improved web page recommendation algorithm using profile aggregation based on the clustering of transactions (IPACT) was proposed by [30]. This algorithm is a two phases recommendation algorithm (offline and online) based on the clustering technique. Offline phase is responsible for data pre-processing and session clustering using previous log data of profiles. The input for the online stage is the output of the offline phase. By using offline & online session clustering, the online phase will generate the session classification. After that, the recommendation engine will make the proper recommendation based on session clustering & classification. The main task of both phases is to generate the navigation patterns profiles to get a recommendation. This paper solves a similar problem, but our work differs in using a dynamic pruned search technique while searching for a similar session, which ensures finding a result.

A Novel web prediction approach was introduced by [31]. It predicts the next movement for web users considering current sequential information and content information that exists in web navigation patterns. Such a model adopts the application of soft clusters through the clustering process to enhance captured multiple user's concerns. It applies a rough set-based similarity upper approximation using both sequential similarity and content similarity. Singular value decomposition (SVD) has been used to generate recommendations for users. This paper solves a similar problem, but our work differs in the use of weighted session rather than rough set clustering. That may lead to imprecision while forming clusters, these sessions are used later to compute other factors used to rank web pages.

§ 3. Our Approach

This paper introduces efficient online navigation assistance for web users. The proposed model presents a page ranking scheme for rating relevant web pages that convenient for the user request. Our work is based on the application of web usage mining techniques respecting the historical navigation behaviors existed in the weblog. Fig. 1 demonstrates the working mechanism of our model in real-time responding to user requests. It starts with receiving a request from the user agent; the server response involves analyzing the current user's behavior carefully. Finally, it provides a client with a list of recommended web pages sent from server to user agent.

On the other hand, this work incorporates four main phases, as mentioned in Fig. 2. The first one is the preprocessing of available web data. In consequence, knowledge extraction represents

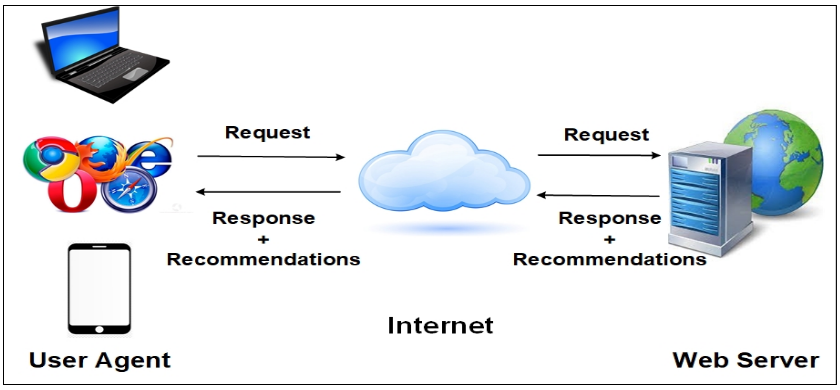

Fig. 1. Real-time response to users' requests

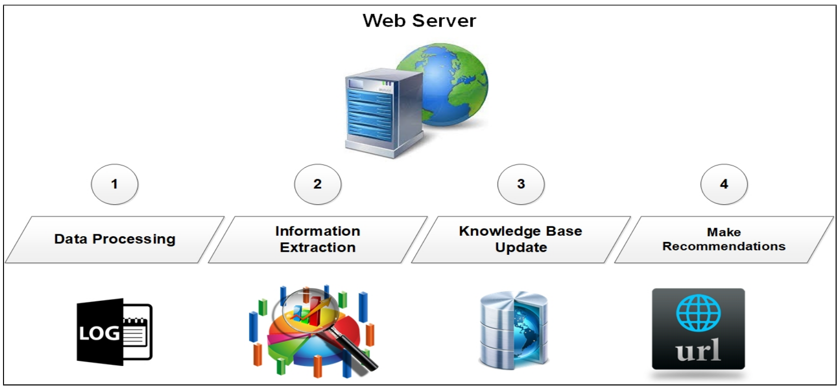

Fig. 2. Server-side processing phases

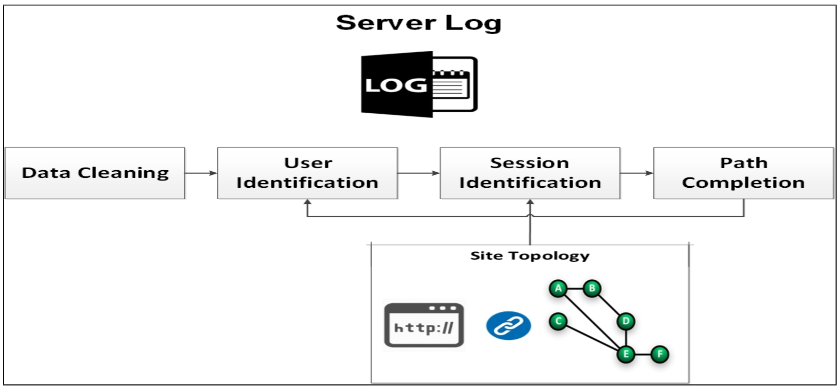

Fig. 3. Web data preprocessing steps

the second phase, which making and providing by updating the knowledge base. Finally, making and providing the recommendations is the last step.

3.1. Data Processing

This phase involves performing the preparation of a web access log file as a primary source of web usage data. The process starts with importing selected parsed access logs into a relational database. Thus, the preparation process takes place on the imported data, as shown in Fig. 3. The phase incorporates the following steps:

Data Cleaning. Cleaning and filtering of irrelevant, redundant, and noisy data that is not suitable for the mining process [32,33].

User Identification. Identify unique users by considering each IP address that represents a single user. Furthermore, if more than one log has the same IP address with different User-Agent, such an IP address represents a different user. Next, for each user, the access log is used in conjunction with the referrer logs and site topology to build browsing paths. If the requested page is not reachable directly from a hyperlink from any of the pages visited by the user, then there is another user with the same IP address.

Session Identification. Session Identification is the next step that incorporates one of the significant actions in the data preparation phase. Creating the user sessions involves the division of user's paths, which were formed in the previous step to individual sessions. It's according to a time threshold *(in our experiment 30 minutes)*. If the time between two pages request exceeds this threshold, then it is assumed that the user starts a new session [34–37].

Path Completion. Complete user's paths using site topology and link structure for the web site to fill missing references pages that are not recorded in the access log.

3.2. Information Extraction

The main goal of this phase is to visualize and summarize the data. It involves the extraction of statistical data to get more insights and obtain an overview of data characteristics (e. g., repeat page request, average browsing time, etc.). Furthermore, this information is used in the subsequent phase; also, it could be reported to the web server administrator, advertising agencies, etc. As follow, explanation of this phase's tasks in detailed.

Rating Individual Pages. This step involves the computation of the following factors. 1. Repeat Page Request (RPR) represents the number of times the current page was visited. 2. Average Browsing Time (ABT) indicates the mean stay time by the user while exploring the specified page. Accordingly, such indicators can be used to identify the pages with high and low requests. This information can help web server admin to rearrange the web site to reach the more requested pages easily and quickly.

3.3. Knowledge Base Update

Generally, the integration of the recently summarized data with previously extracted knowledge becomes crucial. So, we will have up-to-date information that helps to improve system performance. Additionally, maintained processed information improves the real-time interaction through the reduction of the processing time and delay as well.

Data Conversion. This step concerns with the user-session representation [38]. Not all sessions and web pages are involved. That is, the sessions that consist of less than two visited pages and web pages with a request repetition less than a specified threshold are excluded. Consequently, certain thresholds are set to reduce search space and increase recommendation accuracy.

Data Aggregation. Concerning the previously existed patterns, a weight will be assigned that represents the number of times it appear identically; this technique reduces the size of data included in search space by over 70 %. On the other hand, regarding the new patterns, it will be inserted as a new entry.

3.4. Make Recommendation

This phase involves the application of the proposed algorithm on the knowledge base in conjunction with current user requests. As follows, a demonstration of the algorithm in more detail with clear and specific steps.

Input. The system receives the input parameters, which consist of the current user's browsing history involving the most recent requests *"input pattern"*.

Processing:

1. Search in the knowledge base for the sessions, which contain one or more pages from the input pattern.

2. Exclude input pattern from results and create a list of individually unique pages.

3. Compute the repetition for every page in the list, then divide the result by the total number of pages repetition in search result "Repetition Ratio".

4. Compute the Input Pattern Repetition "IPR" by counting the number of times input pattern was repeated in the search result. Zero results (IPR = 0) indicate no matched result; then use a dynamic pruned search until (IPR > 0) by eliminating the earliest page from the input pattern ($L = L - 1$), where L refers to the length of the pattern.

5. Compute the Page Support (PS) for each web page in the list using Eq. (3.1), then compute the Page Coverage (PC) by dividing results by IPR from step No. 4.

$$PS_{cp_i} = \sum_{\substack{ip_i \in vp_i \\ cp_i \in vp_i}} w(vp_i), \tag{3.1}$$

where PS_{cp_i} represents page support for candidate page p_i, and $w(vp_i)$ is the occurrence of the visited pattern vp_i that includes the input pattern page ip, and includes the candidate page cp_i.

6. Calculate the Candidate Page Rate (CPR) for every page in the list using Eq. (3.2). At this step, the derived variables "page coverage, repetition ratio, and average browsing time", are multiplied by weights specified to each one. Then, aggregate results and divide by the sum of weights. (Normalized average browsing time was done using Eq. (3.3) to bring them into proportion with other variables).

$$CPR_{cp_i} = \frac{W_1 * PC_{cp_i} + W_2 * RR_{cp_i} + W_3 * ABT_{cp_i}}{\sum_{j=1}^{n} W_j}, \tag{3.2}$$

where CPR_{cp_i} represents page rate for candidate page cp_i, and w_i represents weights for a specific variable.

$$ABT_{cp_i,0to1} = \frac{ABT_{cp_i} - ABT_{cp_{MIN}}}{ABT_{cp_{MAX}} - ABT_{cp_{MIN}}}, \tag{3.3}$$

where ABT_{cp_i} represents average browsing time of candidate page cp_i, $ABT_{cp_{MIN}}$ represents the minimum value among all candidate pages browsing time, $ABT_{cp_{MAX}}$ represents the maximum value among all candidate pages browsing time, and $ABT_{cp_i,0to1}$ refers to that normalized values which are between 0 and 1.

Output. Provide the web user with the recommendation list, that contains the candidate pages sorted by CPR. In the case of the existence of two or more pages having an identical rate, RPR is referenced to re-rank pages; Select the top page and then recommend it to the user.

§ 4. Experimental Evaluation

In order to help the reproduction of our work, we evaluated our approach on a publicly available dataset. Subsequent, we describe the data and the experimental process in detail. Additionally, we present a brief discussion of our experimental results.

Dataset. We performed experiments on the log file of the central web server of DePaul University CTI (http://www.cs.depaul.edu) during April 2002. The original file size was 252 MB that involves 1051105 records of request to the webserver with the original log format used by IIS [39]. The dataset includes only references to the content files and scripts that generate content pages. There are no other cleaning operations, such as removing erroneous references or spider navigational references performed by owners. Generally, any record in the access log file includes the following fields: user IP address, user name, date and time of the request, request method, requested page, web site name, network web server name, server IP, port number, user search query, time taken, hostname, protocol version, status code, referred page, and user agent.

Implementation. We implemented our approach, OWNA, in C#. We used a command-line utility Microsoft log parser to import the log file into the Database. Also, we used SQL Server to store and manage data.

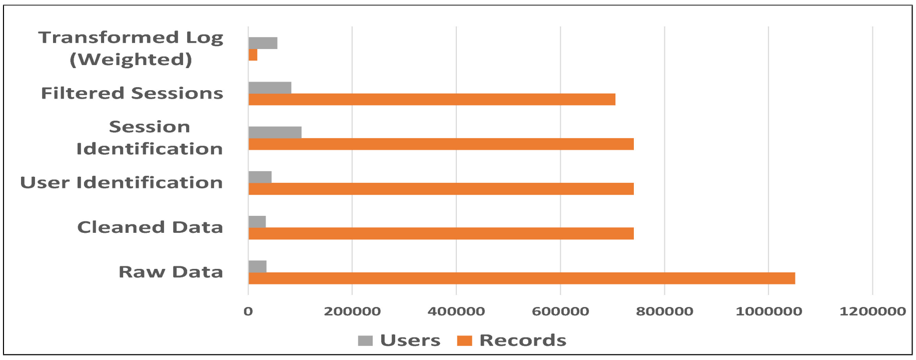

Fig. 4. Data size through processing phases

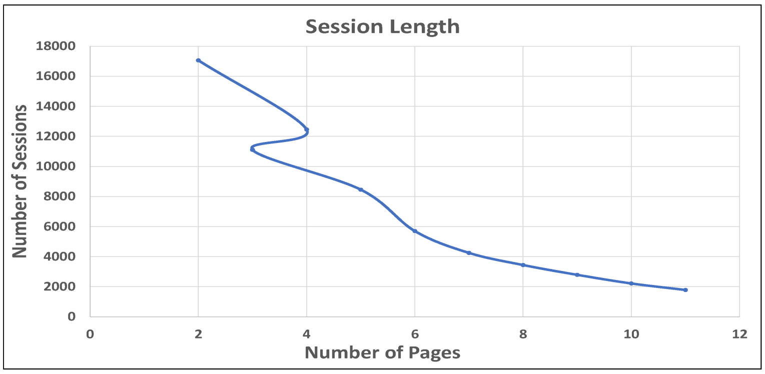

Fig. 5. The most frequent sessions' lengths

Data Manipulation. Raw data have information not relevant to our experiment's purpose (e.g., time taken, port number, etc.). Therefore, essential preparation must be done that involve removing such fields. Next, creating a timestamp for each request that serves as a multi-purpose field. Regarding the removal of erroneous and invalid requests, we only preserve records with status code from 200 to 299, that refers to the successful claims; almost 75.39 % of raw data.

Similarly, we only keep records with a request method "GET"; that represents over 95 % of raw data. Additionally, we delete all requests that contain graphical extensions or any other content pages. On the other hand, to ensure data validity and quality of results, non-human behavior like spiders, crawlers and automatic web bots are deleted, it represents almost 2.16 % of raw data.

Consequently, the next step is the identification of all unique users to study the individuals' behavior. The applied method incorporates the division of user's transactions into sessions respecting to a 30-minute time threshold to consider a later request as a new session beginning. We preserve only repeated web pages and sessions with accepted length. Changes of data size across adjacent processing phases are shown in Fig. 4. Also, Fig. 5 shows the most common sessions lengths and their frequencies. The length of kept sessions' ranges from 2 to 10 pages. We excluded single-page sessions and too long sessions as well, to sustain recommendation accuracy.

The aggregated sessions represent the primary component in our knowledge base. We introduce a representation form that includes the session's pattern, average browsing time, repetition. Such a method reduces the size of the dataset by over 70 %. Therefore, it helps in the search process by reducing the search space leading to speed the search process, and thus reduce the time of making the recommendation.

Testing Process. We randomly chose a sample of the dataset, then removed the last page from every session. Next, we inputted data to the algorithm. We compared the outputs of our model with actually visited pages to compute the efficiency of our work. We performed many trials with

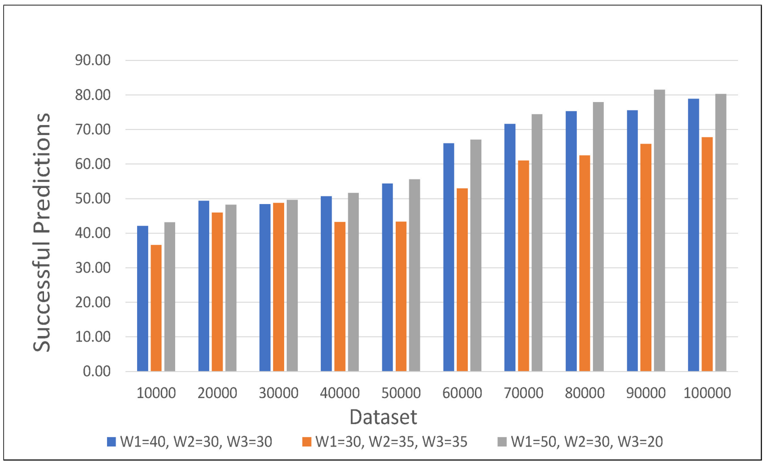

Fig. 6. Number of correct predictions across different combinations of weights

Table 1. Comparison between current and previous models.

Data Set (Records)	Approach I	Approach II
10000	40.00	43.10
20000	41.15	48.20
30000	49.06	49.61
40000	51.86	51.62
50000	56.82	55.60
60000	66.90	67.02
70000	76.30	74.42
80000	79.49	77.93
90000	79.85	81.49
100000	79.89	80.25

different samples. Many combinations of weights were introduced, as shown in Fig. 6. Among too many trials, the best combination that assigns 50:30:20 to page coverage, repetition ratio, and average browsing time, respectively. Candidate pages were sorted by CPR, but in the case of the existence of two or more pages with an identical rate, RPR referenced to re-rank pages. The selection of the top page represents the last task, then this list is recommended to the user.

Evaluation Metrics. We evaluated our approach, OWNA, in terms of an accuracy metric. Accordingly, accuracy has been defined as the ratio of the number of correct recommendations to the number of total recommendations.

Baselines. We compared our approach, OWNA, against similar models, and our previous version as well for evaluation purpose in terms of prediction accuracy, and to demonstrate the extent of our model's ability to give convenient recommendations to clients.

Based on the results shown in Table 1, which presents a comparison between the OWNA model and our previous version, it's clear that OWNA provides better performance with a smaller number of inputs and little improvements with a bigger number of inputs. This behavior seems to be more beneficial in the real-world due to the highly frequent changes in web pages that don't leave sufficient time for a significant number of views for the same source. Thus, the new approach offers better improvements in the coverage and accuracy of the recommendation systems in comparison with the previous work.

For the following experiment conducted between the OWNA model and the WebPUM mo-

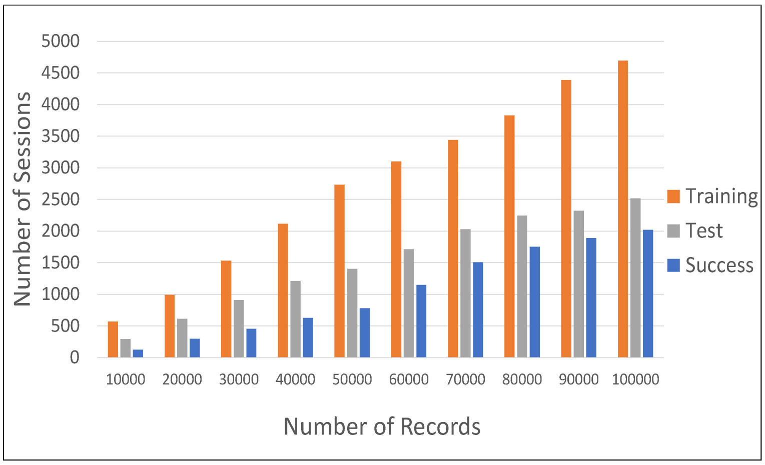

Fig. 7. The number of success predictions of the OWNA model regarding the sample dataset distribution

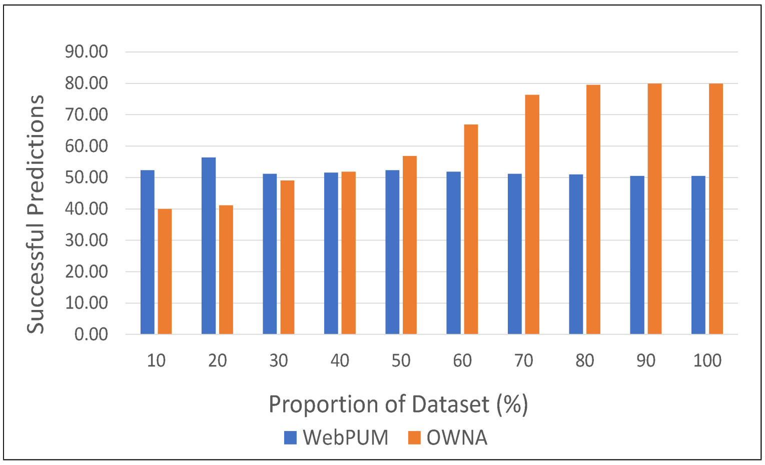

Fig. 8. The number of correct predictions of the OWNA mode and the WebPUM model

del [29], we randomly selected a sample. We then divided it into ten parts so that each piece represents a proportion of the sample starting from 10 % to 100 % of the sample. Fig. 7 demonstrates and summarizes the data distribution; the training set ratio to the test set is 70 to 30, respectively. The session's average length is almost 13-page views.

Based on the test results shown in Fig. 8, we can observe that the OWNA model starts with achieving a moderate prediction accuracy and goes to reaching a noticeable improvement by a high number of successful predictions with growing in the sample size. On the other hand, the WebPUM model starts with a simple preference, but it remains stable. It does not change by the variations of the proportion of the dataset, with a slight decrease in the number of successful predictions.

Next, we conducted a comparison between the OWNA model and the model introduced by [31], in terms of prediction accuracy. Here, we chose a random set that consists of a 5000 user-sessions as a training set. Similarly, we randomly selected ten different groups of size 2000 user-sessions from the dataset as test sets. Table 2 shows the accuracy of the results regarding the OWNA model across various samples of the CTI dataset.

We have compared test results of our model with the results of Mishra's model using the user-defined parameter M (*number of clusters chosen for constructing the response matrix* A), where $M = 32$. Fig. 9 shows the results' accuracy of our model compared with the first prediction, which is greater than Mishra's model. In contrast, the success of the second and third predictions

Table 2. Test result accuracy of OWNA model across number of predictions

Test Set	No. of Predictions		
	1	2	3
Sample 1	42.00	43.25	46.08
Sample 2	36.00	39.45	42.85
Sample 3	37.00	40.65	43.95
Sample 4	39.25	45.45	47.15
Sample 5	35.49	37.19	40.05
Average	37.95	41.20	44.02

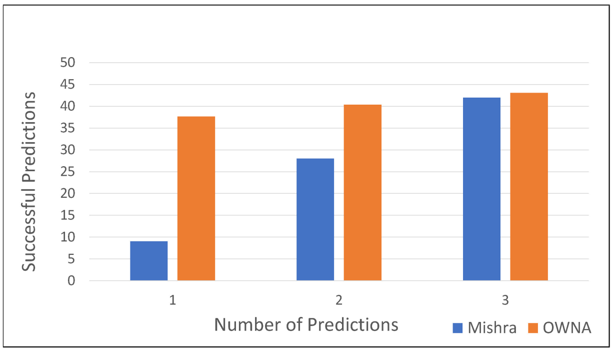

Fig. 9. Number of correct predictions of the OWNA model and Mishra's model

has slight effects on improving results. However, in Mishra's model, the number of predictions has a significant impact on improving the accuracy of the results.

Finally, we performed the last comparison between the OWNA model and the IPACT model [30]. Recall that the experiment was designed to evaluate the ability of both models on predicting the next web page visit based on the current navigation session for testing the prediction accuracy of our algorithm. Page coverage has been used in our experiments instead of recommendation scores and applies a recommendation threshold to the set of recommended pages. Fig. 10 shows test results for both models. The results of the two models are close together. Still, in the case of our model, the more restrictive of the recommendation score has a more significant effect on improving the quality of predictions than the comparative model.

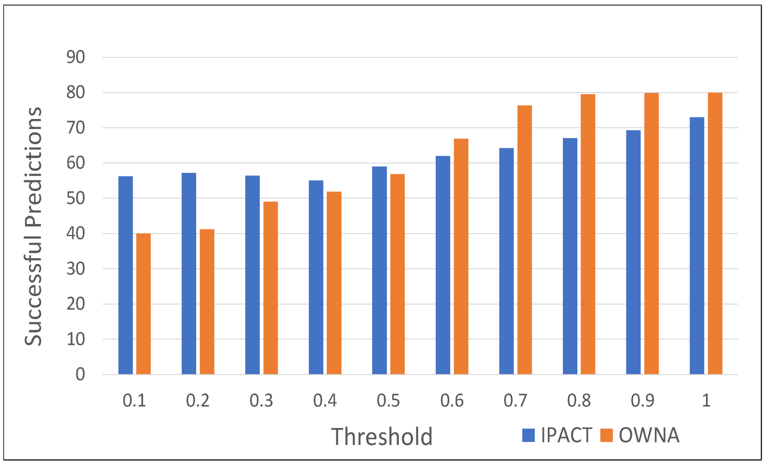

Fig. 10. The number of correct predictions of the OWNA model and the IPACT model

§ 5. Conclusions

The problem of information overloading attracted a reasonable interest from researchers to help web users to access intended resources while surfing the web. Many approaches were introduced in this manner, but they are still insufficient concerning the velocity of producing data available on the internet. Consequently, it's clear that getting insights and employing the advantages available through web usage resources has growing importance. Web systems record user-server interaction; therefore, studying, analyzing, and modeling historical web navigation behaviors become essential to improve the quality of web services offered to web users. There are many sources to obtain recorded behavior; still, weblog files are the most common resources.

In this paper we introduced an online web navigation assistant model (OWNA) to help users in real-time to access the most related web pages to search topic. We aim to facilitate web surfing and reduce wasted time while surfing irrelevant resources. The mechanism of our work relies on the comparison of current user surfing behavior with processed historical data and selects the most similar navigation patterns. Next, we combined these results with other factors that exist in our knowledge base obtained during the processing of historical data. The results of this procedure are a list of candidate web pages. Then, we applied the weighting scheme to that list; the most rated web pages will be recommended to the user. Accordingly, there is a guarantee to help the user in the case of a loss of similar historical patterns.

We have experimented with this work on real data and compared it with other models. Results presented in the illustrative case study demonstrate that our approach is more efficient and reliable than the compared models. In the future, there is a plan to design an incremental algorithm, and introduce another scheme to measure the similarities between web pages, which exploits the existence of web page semantics. Also, we plan to make improvements in the weighting scheme to include additional factors.

REFERENCES

1. Cooley R., Mobasher B., Srivastava J. Web mining: information and pattern discovery on the World Wide Web, *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*, IEEE, 1997, pp. 558–567. https://doi.org/10.1109/tai.1997.632303
2. Resnick P., Varian H. R. Recommender systems, *Communications of the ACM*, 1997, vol. 40, no. 3, pp. 56–58. https://doi.org/10.1145/245108.245121
3. Burke R. Hybrid recommender systems: survey and experiments, *User Modeling and User-Adapted Interaction*, 2002, vol. 12, no. 4, pp. 331–370. https://doi.org/10.1023/A:1021240730564
4. Al-Yazeed N. M. A., Gadallah A. M., Hefny H. A. A hybrid recommendation model for web navigation, *2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS)*, IEEE, 2015, pp. 552–560. https://doi.org/10.1109/IntelCIS.2015.7397276
5. Herlocker J. L., Konstan J. A., Borchers A., Ried J. An algorithmic framework for performing collaborative filtering, *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '99)*, ACM, 1999, pp. 230–237. https://doi.org/10.1145/312624.312682
6. Deshpande M., Karypis G. Item-based top-N recommendation algorithms, *ACM Transactions on Information Systems (TOIS)*, 2004, vol. 22, no. 1, pp. 143–177. https://doi.org/10.1145/963770.963776
7. Jafari M., Sabzchi F. S., Irani A. J. Applying web usage mining techniques to design effective web recommendation systems: a case study, *Advances in Computer Science: International Journal (ACSIJ)*, 2014, vol. 3, no. 2, pp. 78–90. http://www.acsij.org/acsij/article/view/216
8. Sarria M. D. D., Guzman E. L. A recommendation-based web usage mining model for a university community, *2012 Eighth Latin American Web Congress*, IEEE, 2012, pp. 71–78. https://doi.org/10.1109/la-web.2012.23

9. Fu X., Budzik J., Hammond K. J. Mining navigation history for recommendation, *Proceedings of the 5th International Conference on Intelligent User Interfaces*, ACM, 2000, pp. 106–112. https://doi.org/10.1145/325737.325796
10. Wu Y.-H., Chen Y.-C., Chen A. L. P. Enabling personalized recommendation on the Web based on user interests and behaviors, *Proceedings Eleventh International Workshop on Research Issues in Data Engineering. Document Management for Data Intensive Business and Scientific Applications. RIDE 2001*, IEEE, 2001, pp. 17–24. https://doi.org/10.1109/ride.2001.916487
11. Singh M. P. *The practical handbook of internet computing*, New York: Chapman and Hall/CRC, 2004. https://doi.org/10.1201/9780203507223
12. Anitha A., Nallaperumal K. A web usage mining based recommendation model for learning management systems, *2010 IEEE International Conference on Computational Intelligence and Computing Research*, IEEE, 2010, pp. 1–4. https://doi.org/10.1109/iccic.2010.5705888
13. Nigam B., Tokekar S., Jain S. Evaluation of models for predicting user's next request in web usage mining, *International Journal on Cybernetics and Informatics (IJCI)*, 2015, vol. 4, no. 1, pp. 1–13. https://doi.org/10.5121/ijci.2015.4101
14. Forsati R., Meybodi M. R. Effective page recommendation algorithms based on distributed learning automata and weighted association rules, *Expert Systems with Applications*, 2010, vol. 37, no. 2, pp. 1316–1330. https://doi.org/10.1016/j.eswa.2009.06.010
15. Mobasher B., Dai H., Luo T., Nakagawa M. Effective personalization based on association rule discovery from web usage data, *Proceedings of the 3rd International Workshop on Web Information and Data Management (WIDM '01)*, ACM, 2001, pp. 9–15. https://doi.org/10.1145/502932.502935
16. Lin W., Alvarez S. A., Ruiz C. Collaborative recommendation via adaptive association rule mining, *Proceedings of the International Workshop on Web Mining for E-Commerce- Challenges and Opportunities (WebKDD '2000)*, Citeseer, 2000. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.7811
17. Langhnoja S. G., Barot M. P., Mehta D. B. Web usage mining using association rule mining on clustered data for pattern discovery, *International Journal of Data Mining Techniques and Applications*, 2013, vol. 2, no. 1, pp. 141–150. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.678.4780&rep=rep1&type=pdf
18. Lakshminarayan C., Kosuru R., Hsu M. Modeling complex clickstream data by stochastic models: theory and methods, *Proceedings of the 25th International Conference Companion on World Wide Web*, International World Wide Web Conferences Steering Committee, 2016, pp. 879–884. https://doi.org/10.1145/2872518.2891070
19. Vellingiri J., Pandian S. C. A survey on web usage mining, *Global Journal of Computer Science and Technology*, 2011, vol. 11, no. 4, pp. 66–72. https://computerresearch.org/index.php/computer/article/view/710
20. Géry M., Haddad H. Evaluation of web usage mining approaches for user's next request prediction, *Proceedings of the 5th ACM International Workshop on Web Information and Data Management (WIDM '03)*, ACM, 2003, pp. 74–81. https://doi.org/10.1145/956699.956716
21. Bhushan R., Nath R. Recommendation of optimized web pages to users using Web Log mining techniques, *2013 3rd IEEE International Advance Computing Conference (IACC)*, IEEE, 2013, pp. 1030–1033. https://doi.org/10.1109/IAdCC.2013.6514368
22. Dhyani D., Bhowmick S. S., Ng W.-K. Modelling and predicting Web page accesses using Markov processes, *Proceedings of the 14th International Workshop on Database and Expert Systems Applications*, IEEE, 2003, pp. 332–336. https://doi.org/10.1109/dexa.2003.1232044
23. Ko H.-G., Kim E., Ko I.-Y., Chang D. Semantically-based recommendation by using semantic clusters of users viewing history, *Proceedings of the International Conference on Big Data and Smart Computing (BIGCOMP)*, 2014, pp. 83–87. https://doi.org/10.1109/bigcomp.2014.6741412
24. Chimphlee S., Salim N., Ngadiman M. S., Chimphlee W., Srinoy S. Rough sets clustering and Markov model for web access prediction, *Proceedings of the Postgraduate Annual Research Seminar*, 2006, pp. 470–475. http://eprints.utm.my/id/eprint/3370/

25. Khalil F., Li J., Wang H. A framework of combining Markov model with association rules for predicting web page accesses, *Proceedings of the 5th Australasian Conference on Data Mining and Analystics (AusDM '06)*, Australian Computer Society, 2006, pp. 177–184. https://dl.acm.org/doi/10.5555/1273808.1273832
26. Maratea A., Petrosino A. An heuristic approach to page recommendation in web usage mining, *Proceedings of the 9th International Conference on Intelligent Systems Design and Applications (ISDA)*, 2009, pp. 1043–1048. https://doi.org/10.1109/isda.2009.252
27. Rao V. V. R. M., Kumari V. V. An efficient hybrid successive Markov model for predicting web user usage behavior using web usage mining, *International Journal of Data Engineering (IJDE)*, 2010, vol. 1, no. 5, pp. 43–62. http://www.cscjournals.org/library/manuscriptinfo.php?mc=IJDE-25
28. Anitha A. A new web usage mining approach for next page access prediction, *International Journal of Computer Applications (IJCA)*, 2010, vol. 8, no. 11, pp. 7–10. https://www.ijcaonline.org/archives/volume8/number11/1252-1700
29. Jalali M., Mustapha N., Sulaiman M. N., Mamat A. WebPUM: A Web-based recommendation system to predict user future movements, *Expert Systems with Applications*, 2010, vol. 37, no. 9, pp. 6201–6212. https://doi.org/10.1016/j.eswa.2010.02.105
30. AlMurtadha Y., Sulaiman M. N. B., Mustapha N., Udzir N. I. IPACT: Improved web page recommendation system using profile aggregation based on clustering of transactions, *American Journal of Applied Sciences*, 2011, vol. 8, no. 3, pp. 277–283. https://doi.org/10.3844/ajassp.2011.277.283
31. Mishra R., Kumar P., Bhasker B. A web recommendation system considering sequential information, *Decision Support Systems*, 2015, vol. 75, no. 1, pp. 1–10. https://doi.org/10.1016/j.dss.2015.04.004
32. Ali N. M., Gadallah A. M., Hefny H. A., Novikov B. An integrated framework for web data preprocessing towards modeling user behavior, *Proceedings of the 2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, IEEE, 2020, pp. 1–8. https://doi.org/10.1109/FarEastCon50210.2020.9271467
33. Iliou C., Kostoulas T., Tsikrika T., Katos V., Vrochidis S., Kompatsiaris Y. Towards a framework for detecting advanced web bots, *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ACM, 2019, no. 18. https://doi.org/10.1145/3339252.3339267
34. Patel P., Parmar M. Improve heuristics for user session identification through web server log in web usage mining, *International Journal of Computer Science and Information Technologies*, 2014, vol. 5, no. 3, pp. 3562–3565. http://ijcsit.com/docs/Volume5/vol5issue03/ijcsit20140503201.pdf
35. Ganibardi A., Ali C. A. Weblog data structuration: A stream-centric approach for improving session reconstruction quality, *Proceedings of the 20th International Conference on Information Integration and Web-based Applications and Services*, ACM, 2018, pp. 263–271. https://doi.org/10.1145/3282373.3282379
36. Leoni M. D., Dündar S. Event-log abstraction using batch session identification and clustering, *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, ACM, 2020, pp. 36–44. https://doi.org/10.1145/3341105.3373861
37. Chitraa V., Thanamani A. S. A novel technique for sessions identification in web usage mining preprocessing, *International Journal of Computer Applications*, 2011, vol. 34, no. 9, pp. 23–27. https://www.ijcaonline.org/archives/volume34/number9/4127-5958
38. Markov Z., Larose D. T. *Data mining the web: uncovering patterns in web content, structure, and usage*, John Wiley and Sons, 2007. https://doi.org/10.1002/0470108096
39. Microsoft Docs: The Modern Documentation Service for Microsoft, *IIS Logging*, 2018, Available: https://docs.microsoft.com/en-us/windows/win32/http/iis-logging, last access (01 Feb., 2020).

Received 08.07.2020

Ali No'aman Muhammad, Master of Science, Department of Computer Science, Cairo University, Giza, Egypt;
Ph. D. Student at the Department of Computer Science, Faculty of Mathematics and Mechanics, Saint Petersburg State University, Saint Petersburg, Russia;
Assistant Lecturer, Port Said University, Port Said, Egypt.
ORCID: https://orcid.org/0000-0002-3922-7136
E-mail: no3man_mohamed@himc.psu.edu.eg

Gadallah Ahmed Mohamed, Ph.D. in Computer Science, Associate Professor, Department of Computer Science, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt.
ORCID: http://orcid.org/0000-0001-8177-7433
E-mail: ahmgad10@yahoo.com

Hefny Hesham Ahmed, Ph.D. in Computer Science, Professor, Department of Computer Science, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt.
ORCID: http://orcid.org/0000-0001-5862-675X
E-mail: hehefny@hotmail.com

Novikov Boris Asenovich, Doctor of Science in Mathematics and Physics, Professor, Department of Informatics at National Research University Higher School of Economics, Saint Petersburg, Russia.
ORCID: https://orcid.org/0000-0003-4657-0757
E-mail: borisnov@acm.org

Citation: N. M. Ali, A. M. Gadallah, H. A. Hefny, B. A. Novikov. Online Web Navigation Assistant, *Vestnik Udmurtskogo Universiteta. Matematika. Mekhanika. Komp'yuternye Nauki*, 2021, vol. 31, issue 1, pp. 116–131.

Н. М. Али, А. М. Гадалла, Х. А. Хефни, Б. А. Новиков

Онлайн-помощник для навигации в веб

Ключевые слова: веб-майнинг, веб-персонализация, прогнозирование ссылок, анализ использования веб, рекомендательные системы, веб-журнал, помощник веб-навигации.

УДК 004.048, 004.622, 004.657

DOI: 10.35634/vm210109

Задачи поиска релевантной информации создают значительные трудности для пользователей из-за огромного объема данных, доступных в интернет. Эти трудности связаны с известной проблемой информационной перегрузки. В этой работе мы предлагаем онлайн-веб-помощник под названием OWNA. Мы разработали полностью интегрированную платформу для выработки рекомендаций в режиме реального времени, основанную на методах анализа журналов использования веб. Наша работа начинается с подготовки исходных данных, а затем извлечения полезной информации, которая помогает построить базу знаний, а также присваивает определенный вес определенным факторам. Эксперименты показывают преимущества предложенной модели по сравнению с альтернативными подходами.

Поступила в редакцию 08.07.2020

Али Ноаман Мухаммад, магистрант кафедры компьютерных наук Каирского университета, Гиза, Египет;
аспирант кафедры компьютерных наук математико-механического факультета Санкт-Петербургского государственного университета, Санкт-Петербург, Россия;
ассистент, Университет Порт-Саида, Порт-Саид, Египет.
ORCID: https://orcid.org/0000-0002-3922-7136
E-mail: no3man_mohamed@himc.psu.edu.eg

Гадалла Ахмед Мухаммад, кандидат компьютерных наук, доцент кафедры компьютерных наук факультета аспирантуры по статистическим исследованиям Каирского университета, Гиза, Египет.
ORCID: http://orcid.org/0000-0001-8177-7433
E-mail: ahmgad10@yahoo.com

Хефни Хешам Ахмед, доктор компьютерных наук, профессор кафедры компьютерных наук факультета аспирантуры по статистическим исследованиям Каирского университета, Гиза, Египет.
ORCID: http://orcid.org/0000-0001-5862-675X
E-mail: hehefny@hotmail.com

Новиков Борис Асенович, доктор физико-математических наук, профессор департамента информатики Высшей школы экономики в Санкт-Петербурге, Санкт-Петербург, Россия.
ORCID: https://orcid.org/0000-0003-4657-0757
E-mail: borisnov@acm.org

Цитирование: Н. М. Али, А. М. Гадалла, Х. А. Хефни, Б. А. Новиков. Онлайн-помощник для навигации в веб // Вестник Удмуртского университета. Математика. Механика. Компьютерные науки. 2021. Т. 31. Вып. 1. С. 116–131.