Web Page Access Prediction with Popularity and Similarity-based Page Rank Algorithm

Phyu Thwe[#1], Ei Ei Chaw[#2]

[#]*Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City)*
Pyin Oo Lwin, Myanmar

[1]`pthwe19@gmail.com`
[2]`eieichaw1981@gmail.com`

***Abstract*— Web Usage Mining is an important type of Web Mining, which handles the extraction of interesting knowledge from the web log files. The web is a large source of information that can be turned into knowledge. That part of knowledge is valuable to website and organization of websites and concerns the usage of web itself. The problem of predicting web page access on a web site has fundamental significance due to the rapid growth of the World Wide Web. Predicting the next page to be accessed by the users has attracted a large amount of research. In this paper, a new web usage mining approach is proposed to predict next page access. At first, similar web sessions are clustered from preprocessing results of web log by applying K-mean clustering algorithm. Markov model is used for prediction of next page accesses. Popularity and Similarity-based Page Rank algorithm is applied to predict next page accesses if the ambiguous results are found.**

***Keywords*— Markov model, next page access prediction, page rank algorithm, web log mining, web usage mining**

I. Introduction

The Internet provides a rich environment for users to retrieve information. At the same time, it also makes easy for a user to get lost in the millions of information. One way to help the users with their relevant need is to predict a user's future request and use the prediction for pre-fetching, caching and recommendation. The purpose of this paper is to explore ways to use the information from web logs for predicting users' next page access on the particular web site.

In this paper, a Page Rank-like algorithm is proposed to manage more accurate next page access prediction. Page Rank [1] is a Web page popularity measure used in the Google search engine. Popularity is equivalent to the likelihood of future access of the page in a set of Web pages; therefore the Page Rank value of a Web page can be used to measure the likelihood of the page being accessed. The popularity and similarity of web page is used so that the pages that are more important to users can be found.

The system focuses on the improvements of predicting web page access. Data preprocessing is the process to convert the raw data into the data abstraction necessary for the further applying the data mining algorithm. Web page sessions are grouped by using the k-means clustering algorithm. To predict the next page access, kth order Markov model is used. And if ambiguous results are found, Page Rank-like algorithm is used for deciding the correct answer.

The objectives of the proposed system are:

- To improve pattern discovery steps in web usage mining that would reveal new opportunities to the data analyst
- To predict web user's behaviour and their next movement
- To improve the web page access prediction accuracy
- To decrease state space complexity

The rest of the paper is organized as follows: In Section 2, the related work is overviewed. In Section 3, the required theoretical background concerning the Markov Models and the Page Rank Algorithm is presented. In Section 4, the proposed system is provided in which PSPR can be applied. Section 5 provides the experimental results. Finally, Section 6 has the conclusion and future work.

II. Related Work

A number of researchers attempted to improve the web page access prediction accuracy or coverage. Khalil et al. [2] have been proposed the combination of Association rules and Markov model. They used lower order all k-th Markov models to predict the next page to be accessed. Khalil et al. [3] introduce the Integration Prediction Model (IPM) by combining Markov model, Association rules and clustering algorithm together. Then, the prediction is performed on the cluster sets rather than the actual sessions. A recommender system used longest common subsequence (LCS) algorithm to classify current user activities to predict user next movement in the particular web sites in [4].

The most widely approach is Web usage mining that involves many algorithm like Markov models, Association rules and clustering [5]. However, there are some challenges with the current state of the art solutions when, it comes to accuracy, coverage and performance. A Markov model is a popular approach to predict what pages are likely to be accessed next.

The Page Rank algorithm [6] uses the link structure of pages for finding the most important pages with respect to the search result. The algorithm states that if the in-links (pages that pointed to the page) of a page are important, then out-links (pages that pointed by the page) of the page also become important. Therefore, the page rank algorithm distributes the rank value of itself through the pages it points to. The Page Rank algorithm [1] is the most popular algorithm proposed for ranking the results of a Web search engine. Many variations have been proposed in this context. There are models that bias Page Rank algorithm with other type of web usage data, structural data or web contents. In [7], Usage Based Page Rank algorithm is introduced as the rank distribution of pages depending on the frequency value of transitions and pages. They model a localized version of ranking directed graph. In [8], they modify Page Rank algorithm with considering the time spent by the user on the related page. However, in their work, the effect of size value of pages is not considered. In [9], Duration based Page Rank and Popularity based Page Rank are introduced. They are based on duration and popularity of page and they did not consider the page's similarity.

III. Theory Background

A. K-means Clustering

It is the simplest and most commonly used clustering algorithm, especially with large data sets. The process is as follow:

Define a set of sessions (n-by-p data matrix) to be clustered.

Define a chosen number of clusters (k).

Randomly assign a number of sessions to each cluster.

The k-means clustering repeatedly performs the following:

Calculate the mean vector for all items in each cluster.

Reassign the sessions to the cluster whose center is closest to the session.

Until there is no change for all cluster centers.

B. Markov Model

Markov model is a commonly used method for modelling stochastic sequences with an underlying finite-state structure and was shown to be well-suited for modelling and predicting a user's browsing behaviour on a web site [10]. The goal is to build the user behavioural models that can be used to predict the web page that the user will most likely access next. The input for this problem is the sequence of web pages that were accessed by a user and it is assumed that it has the Markov property. In such a process, the past is irrelevant for predicting the future given knowledge of the present. Let, P = {P_1, P_2,... ,P_m} be a set of pages in a Web site. Let, W be a user session including a sequence of pages visited by the user in a visit. Assuming that the user has visited i pages then P (p_i|W) is the probability that the user visits page p_i next. The conditional probabilities are commonly estimated by assuming that the process generating sequences of the web pages visited by users follows a Markov process. That is, the probability of visiting a web page p_i does not depend on all the pages in the web session. Using the Markov process assumption, the web page p_{i+1} will be generated next is given by

$$P_{l+1} = \arg\max_{p \in P} \{ P(P_{l+1} = p \mid p_l, p_{l+1}, \ldots, p_{l-(k-1)}) \} \quad (1)$$

where, k denotes the number of the preceding pages and it identifies the order of Markov model. The resulting model of this equation is called the kth-order Markov model.

C. Page Rank Algorithm

Page Rank, which was developed at Stanford University by Larry Page and Sergey Brin [1], is the most popular link analysis algorithm used to rank the results returned by a search engine after a user query. Page Rank is a static ranking of web pages in the sense that a Page Rank value is computed for each page off-line and it does not depend on search queries [11]. The Web is treated as a directed graph G = (V, E), where V is the set of vertices or nodes, i.e., the set of all pages, and E is the set of directed edges in the graph, i.e., hyperlinks. In page rank calculation, especially for larger systems, iterative calculation method is used. In this method, the calculation is implemented with cycles. In the first cycle all rank values may be assigned to a constant value such as 1, and with each iteration of calculation, the rank value become normalized within approximately 50 iterations under ε = 0.85 [13].

IV. Proposed System

The processing steps of the system have three main phases. Preprocessing is performed in the first phase. The second phase is clustering web sessions using K-means clustering. In the final phase, Markov model is used to predict next page access based on resulting web sessions. The popularity and similarity-based page rank algorithm is used to decide the most relevant answer if the ambiguous result is found in Markov model prediction.

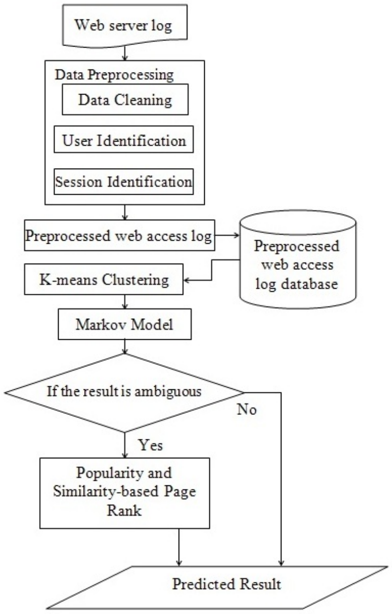

Figure 1. Proposed System Architecture

The input of the proposed system is a web log file. A web log is a file to which the web server writes information each time a user requests a resource from that particular site. Web server logs are plain text (ASCII) files [12].

A. Data Preprocessing

Web log data pre-processing step is a complex process. It can take up to 80% of the total KDD time [12]. The aim of data pre-processing is to select essential features, clean data from irrelevant records and finally transform raw data into sessions. The data preprocessing step involves data cleaning, user identification and session identification. In the data cleaning phase, it eliminates the irrelevant information from the original web log file. Usually, this process removes requests concerning non-analyzed resources such as images, multimedia files and page style files. For example, requests for graphical page content (*.jpg & *.gif images) and requests for any other file which might be included into a web page. By filtering out useless data, the system can reduce the log file size to use less storage space and to facilitate upcoming tasks. In the user identification phase, some rules are used to identify unique users: If there is new IP address, then there is a new user. In the session identification phase, the system uses the time out mechanism to identify the access time of the user for a respective web page. The time out mechanism defines a time limit for the access of a particular page and this limit is usually 30 minutes. Therefore, if the user has accessed the web page for more than 30 minutes, this session will be divided into more than one session.

TABLE I. EXAMPLE: SESSION TABLE

Session ID	Transitions
S1	P3, P2, P1
S2	P3, P5, P2, P1, P4
S3	P4, P5, P2, P1, P5, P4
S4	P3, P4, P5, P2, P3
S5	P1, P4, P2, P5, P4

B. *Clustering*

This paper introduces a new method that integrates clustering, Markov model and page rank algorithm in order to improve the Web page prediction accuracy. The problem of this paper is the grouping of such sessions into k number of clusters in order to improve the Markov model prediction accuracy. Because of the increased number of clustering methods, clustering tasks can be tedious and complex. Clustering has several methods such as distance-based or model-based, hierarchical or non-hierarchical, and supervised or unsupervised. For the purpose of this paper, the system use a straightforward implementation of the k-means clustering algorithm based on user sessions.

C. *Prediction*

Markov models are commonly used in the identification of the next page to be accessed by the user based on the previously accessed pages. Once the transition probability matrix is built by using Markov model, making prediction for web sessions is straightforward. For example, consider a user that has accessed pages P2→ P5→ ?. If the system wants to predict the page that will be accessed by the user next, using a Markov model, the system will first identify the state {P2, P5} and look up the transition probability matrix to find the page p_i that has the highest probability and predict it. In the case of the example, the prediction would be page P4. However, there is an ambiguous result will be found to predict P2→ P1→? because the pages have same probability to predict for probability of pages P2=P4=P5=1/3. When the ambiguous result is found, the popularity and similarity-based page rank algorithm (PSPR) is used to make the decision for correct answer.

Popularity and Similarity-based Page Rank (PSPR) calculation simply depends on the duration values of pages and transitions, their web page file size and similarity of web page. The popularity value of page rank was discussed in [9]. Popularity defines in two dimensions. They are page dimension and transition dimension. For both dimensions, popularity defines in terms of time user spends on page, size of page and visit frequency of page. Page popularity is needed for calculating random surfer jumping behaviour of the user and transition popularity is needed for calculating the normal navigating behaviour of the user.

Similarity of web page is important to predict next page access because million of users generally access the similar web page in a particular Web site. The calculation of the similarity is based on web page URL. The content of pages is not considered and this calculation does not need for making a tree structure of the Web site. For example, suppose "/shuttle/missions/sts-73/mission-sts-73.html" and "/shuttle/missions/sts-71/mission-sts-71.html" are two requested pages in web log. The value of the similarity of the two web pages ($SURL_{j\to i}$) can be calculated as follow. These two URLs are stored in string array by dividing "/" character. And then, the system compute the length of the two arrays and give weight to the longer array: the last room of the array is given weight 1, the second to the last room of the array is given weight 2, the third to given weight 3 and so on and so forth, until the first room of the array is given higher length of the array. The similarity between two strings is defined as the sum of the weight of those matching substrings divided by the sum of the total weights. For this example, the similarity of the two requested web pages is SURL = (4 + 3)/(4+3+2+1) = 0.7.

In equation 2, the popularity and similarity based-page rank value for each page is calculated. In which, ε is a damping factor and usually ε = 0.85. In(p_i) is the set that keeps the in-links of that page. Out(p_j) is the set of pages that point to p_j. $w_{j\to i}$ is the number of times pages j and i appear consecutively in all user sessions. $d_{j\to i}$ is the duration of the transition and s_i is the size of the transition's result page. WS is the web session. $SURL_{j\to i}$ is the similarity of web page j to page i.

$$PSPR_i = \varepsilon \times \sum_{P_j \in In(P_i)} \left[PSPR_j \times \frac{w_{j\to i}}{\sum_{P_k \in Out(P_j)} w_{j\to k}} \times \frac{(d_{j\to i}/s_i)}{\max(d_{m\to n}/s_n)} \times \frac{SURL_{j\to i}}{\sum_{P_k \in Out(P_j)} SURL_{j\to k}} \right] + (1-\varepsilon) \times \frac{w_i}{\sum_{P_j \in WS} w_j} \times \frac{(d_i/s_i)}{\max(d_m/s_m)} \tag{2}$$

Where, $\frac{w_{j\to i}}{\sum_{P_k \in Out(P_j)} w_{j\to k}} \times \frac{(d_{j\to i}/s_i)}{\max(d_{m\to n}/s_n)}$ is the transition popularity based on transition frequency and duration. $\frac{SURL_{j\to i}}{\sum_{P_k \in Out(P_j)} SURL_{j\to k}}$ is the similarity calculation between web pages. $\frac{w_i}{\sum_{P_j \in WS} w_j}$ is the frequency calculation for page i. $\frac{(d_i/s_i)}{\max(d_m/s_m)}$ is the average duration calculation for page i. The popularity of page is calculated based on page frequency and average duration of page.

By using this equation, the popularity and similarity-based page rank (PSPR) can be calculated for every page. In order to make rank calculations faster, the system record required steps of the calculations to database. The step values related to rank calculations are average duration value of pages, average duration values of transitions, page size, frequency value of pages, frequency value of transitions and the similarity value of pages. The result can be used for ambiguous result found in Markov model to make the correct decision.

D. PSPR Calculation

This section presents how the given equation is used in the proposed algorithms on a sample case. PSPR calculation is presented in using frequency, time, page size and similarity of web page. In Table 2, page, frequency values of pages, duration and page size for the sample case are listed.

TABLE II. PAGE PROPERTIES FOR SAMPLE SECTION

Page	Frequency (w_i)	Duration (d_i)	Page Size (byte) (s_i)
P1	3	297000	7543
P2	5	231000	4179
P3	4	197000	4085
P4	6	105000	3985
P5	5	187000	6245

TABLE III. FREQUENCY AND DURATION OF EACH PAGE

Page	Frequency of page $\frac{w_i}{\sum_{P_j \in WS} w_j}$	Duration of page $\frac{(d_i/s_i)}{\max(d_m/s_m)}$
P1	0.13	0.71
P2	0.21	0.98
P3	0.17	0.87
P4	0.26	0.47
P5	0.21	0.54

TABLE IV. FREQUENCY, DURATION AND SIMILARITY OF EACH PAGE TRANSITION

Transition	Frequency for transition	Duration for transition	Similarity of web page
	$\frac{w_{j\to i}}{\sum_{P_k \in Out(P_j)} w_{j\to k}}$	$\frac{(d_{j\to i}/s_i)}{\max(d_{m\to n}/s_n)}$	$\frac{SURL_{j\to i}}{\sum_{P_k \in Out(P_j)} SURL_{j\to k}}$
P3→ P2	0.33	0.85	0.78
P2→ P1	0.6	0.82	0.87
P3→ P5	0.33	0.67	0.88
P5→ P2	0.6	0.78	0.67
P1→ P4	0.67	0.58	0.89
…	…	…	…

In Table 3, frequency of page and duration of page for the sample case are given. They are easily calculated by using the above data. This is a synthetic data that is produced for illustration purpose.

In table 4, frequency, duration and similarity of each page transition are given.

By using this data, the system can easily calculate the popularity and similarity-based page rank (PSPR). From this PSPR result, the page to be next accessed with highest rank value can be determined. In order to make rank calculations faster, the system record intermediate steps of the calculations to database. Intermediate step values related to rank calculations are duration value of pages, duration values of transitions, page size, frequency value of pages, frequency value of transitions and similarity of web pages.

V. EXPERIMENTAL EVALUATION

In this section, the experiment is presented to be performed for evaluating the impact of the proposed technique on the prediction process. Overall the experiment has verified the effectiveness of the proposed techniques in web page access prediction based on particular website.

A. Experimental Data Set

For our experiments, the system used NASA web server data sets. The system obtained the web logs in August, 1995 and used the web logs from 01/Aug/1995 to 15/Aug/1995 as the training data set, and the web logs from 16/Aug/1995 to 17/Aug/1995 as the testing data set. The system filtered the records (such as *.jpg, *.gif, *.jpeg) and only reserved the hits requesting web pages. When identifying user sessions, the system set the session timeout to 30 minutes, with a minimum of 10 pageviews per session. After filtering out the web session data by preprocessing, the training data set contained 94307 records and 5574 sessions, while the testing data set contained 1271 records and 74 sessions.

B. Evaluation Methods

In comparing the predictions with the real page visits, the system use two similarity algorithms that are commonly preferred for finding similarities of two sets. The first one is called Osim [13] algorithm, which calculates the similarity of two sets without considering the ordering of the elements in the set. It focuses on the number of common elements of two sets with a limit value. The limit value can be seen as the top-n next page prediction for a visited page. The equation of

Osim algorithm is defined in Equation 3, where A and B are the sets to be compared, having the same length and n is the top-n value of comparison. The similarity value range is [0-1] and 1 denotes maximum similarity.

$$Osim\ (A, B) = \frac{A \cap B}{n} \qquad (3)$$

As the second similarity metric the system use Ksim similarity algorithm, which concerns Kendall Tau Distance [13, 9] for measuring the similarity of next page prediction set produced by training data set and real page visit set on the test data. Kendall Tau Distance is the number of pairwise incompatibility between two sets. In this similarity metric, as the distance increases, similarity decreases. Ksim similarity calculation is given in Equation 4. Sometimes the compared sets may have different lengths. The lengths of the sets are equalized, by utilizing the union set, as shown in Equation 4.

$\delta_1 = A \cup B - A$ and $\delta_2 = A \cup B - B$

$A' = A$ followed by δ_1 and $B' = B$ followed by δ_2 then,

$$Ksim\ (A, B) = 1 - \frac{\tau dis\ \tan\ ce\ (\delta_1', \delta_2')}{|A \cup B| \times (|A \cup B| - 1)} \qquad (4)$$

τ distance comes from the Kendall Tau distance algorithm mentioned before. In the experiment setup, the system make experiment with top-3 comparison that are measured by Ksim and Osim methods.

C. *Experimental Results*

The results of the experiment for the next page prediction accuracy for popularity and similarity-based page ranking algorithm under Ksim and Osim similarity metrics are given in Figure 2.

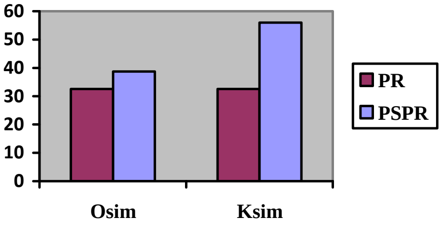

Figure 2. Average Osim, Ksim values for top-3 prediction

As depicted in Figure 2, PSPR outperforms PR significantly in all Osim and Ksim values. In the top-3 prediction, PSPR outperforms 38.68% and 55.92% for Osim and Ksim respectively, while PR outperforms 32.5% [14]. Therefore, this paper can confirm that popularity and similarity-based page rank can improve the accuracy of Web page prediction.

VI. Conclusion And Future Work

In this paper, the system used Markov model and popularity and similarity-based page rank algorithm in the context of web page access prediction. The experimental result shows that this approach can produce more accurate web page prediction than the methods that use only page rank algorithm. However in this experimental setup, the system only used the 1st order Markov model. In the future work, the system will take into account using higher order Markov model to improve the prediction accuracy.

Acknowledgement

I would like to express my deep appreciation to my supervisor, Dr. Ei Ei Chaw, Lecturer, University of Technology (Yatanarpon Cyber City) for her close supervision, helpful advice, encouragement and numerous invaluable suggestions.

References

[1] S. Brin, L. Page, 1998. The anatomy of a large-scale hypertextual Web search engine, Computer Networks, 30(1-7): 107-117, Proc. of WWW7 Conference.

[2] Khalil, F., J. Li and H. Wang, 2006. 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, Inc., pp: 177-184.

[3] Khalil, F., J. Li and H. Wang, 2007. Integrating markov model with clustering for predicting web page accesses. Proceedings of the 13th Australasian World Wide Web Conference (AusWeb 2007), June 30-July 4, Coffs Harbor, Australia, pp: 1-26.

[4] M. Jalali, N. Mustapha, A. Mamat, Md. N. B Sulaiman, 2009. A Recommender System for Online Personalization in the WUM Applications, WCECS 2009, October 20-22, 2009, San Francisco, USA.

[5] M. Eirinaki, M. Vazirgiannis, and D. Kapogiannis. 2005. Web path recommendations based on page ranking and markov models. In Proceedings of the 7th annual ACM international workshop on Web information and data management, WIDM '05, pages 2-9, New York, NY, USA, ACM.

[6] R. Jain, Dr. G. N. Purohit. Jan 2011. Page ranking algorithms for Web Mining. In International Journal of Computer Applications(0975-8887), 2011. Volume 13-No.5, pages 22-25.

[7] M. Eirinaki and M. Vazirgiannis. Nov. 2005. Usage-based pagerank for web personalization. In Data Mining, Fifth IEEE International Conference on, page 8 pp.

[8] Y. Z. Guo, K. Ramamohanarao, and L. Park. Nov. 2007. Personalized pagerank for web page prediction based on access time-length and frequency. In Web Intelligence, IEEE/WIC/ACM International Conference on, pages 687-690.

[9] B. D. Gunel, P. Senkul, 2011. Investigating the Effect of Duration, Page Size and Frequency on Next Page Recommendation with Page Rank Algorithm, ACM.

[10] M. Deshpande and G. Karypis. May 2004. Selective markov models for predicting web page accesses. ACM Trans. Internet Technol., 4:163-184.

[11] Bing Liu, 2007. Web Data Mining Exploring Hyperlinks, Contents, and Usage Data, Springer-Verlag Berlin Heidelberg.

[12] Z. PABARŠKAITĖ, Enhancements of Pre-processing, Analysis and Presentation Techniques in Web Log Mining, Doctoral dissertation was prepared at the Institute of Mathematics and Informatics in 2003–2009.

[13] T. Haveliwala. Topic-sensitive pagerank: a context-sensitive ranking algorithm for web search. Knowledge and Data Engineering, IEEE Transactions on, 15(4):784-796, July-Aug. 2003

[14] C. E. Dinuca, D. Ciobanu. Improving the prediction of next page request by a web user using Page Rank algorithm. Recent Research in Tourism and Economic Development, ISBN: 978-1-61804-043-5, 520-524