Evaluation of Models for Predicting User's Next Request in Web Usage Mining

Bhawna Nigam
Department of Information Technology ,Institute of Engineering & Technology, DAVV Indore, M.P., India
Dr. Sanjiv Tokekar
Department of Electronics & Telecommunication Engineering Institute of Engineering & Technology, DAVV Indore, M.P., India
Dr. Suresh Jain
Department of Computer Engg. Prestige Institute of Engineering Management & Research, Indore, INDIA

Abstract

Prediction of web user behavior is the demand of today competitive edge of World Wide Web. Predicting the next web page is not sufficient, evaluation of prediction models is important because every model have its own pros and cons. Prediction results will be helpful if high prediction accuracy is achieved with minimum complexity, which are depended on the prediction model. Various models and their variations are proposed for predicting the next web page accessed by the web user. Markov model and their variations are found suitable for web prediction. In this research we have evaluated and compared various models for predicting next web page accessed by the web user. Experiments are conducted on three different real datasets.

Keywords

Web usage mining, Markov Model, User Navigation Session, web log and web prediction

1.Introduction

Web prediction is a field of web usage mining in which next accessed web page by web user is predicted. Prediction results can be used for personalization of web, reducing the server response time with proper prefetching and caching strategies [1]. It can provide guidelines for improving the design of web applications, e-commerce to handle business specific issues like customer attraction, customer retention, cross sales and customer departure.

Prediction of next access web page can be achieved through modelling the web log with the help of model. The logging information is stored in a file known as web log file which resides on web server, proxy server or client cite. The web log file is the text file which contains lots of information such as IP address, date, time, request type etc., so it is preprocessed before modelling. From the preprocessed web log information the user navigation session prepared. The user navigation session is finally modeled through a model. Once the user navigation model is ready, the mining task can be performed to predict next accessed web page. In prediction model log file is divided into two parts training file and testing file, training file is used to build the model and testing file is used to test the model. Various models have been proposed to

DOI: 10.5121/ijci.2015.4101

accomplish this task such as Markov Model, Semantic Model, Dynamic Nested Markov Model, association rule mining and many more[3,4].

Predicting the next web page is not sufficient, it is also very important to evaluate the prediction because every model has their advantages and limitations [9]. Lower order Markov model is less complex with low accuracy and high order Markov model has high accuracy with high complexity and low coverage. So before uploading any prediction model on the server, it is very necessary to find limitations of the model. Various parameters are there to evaluate the prediction such as how much time required by model to predict next web page, prediction accuracy, generation time, coverage and many more. In this work we have evaluated and compared different models for mining the web log to predict next accessed web page.

2.Related Work

Several authors have proposed models for modelling the user navigation session to predict next accessed web page. Markov model is widely used for modelling the web navigation sessions. Markov model is based on a well established theory. F. Khalil et al. [5], have proposed a new framework for predicting the next web page access. In this they have study the Markov model for prediction. If the Markov model is not able to predict the next web page then the association rule are used for predicting the next web page. They have also proposed that if there will be ambiguity in the prediction, it will be resolve by association rule. J. Borges et al. [6, 7, 8], have proposed the Higher-order Markov model with clustering technique to improve the effectiveness of Markov Model. The K-mean clustering technique has been used to reduce the state space complexity. F. Khalil et al. [10] have proposed the integrated approach for predicting the next access web page where they have tried to achieve the high level of predictive accuracy with low state space complexity. Siriporn Chimphlee et al. [11], used association rule for next access prediction. Nizar R. [12] proposed semantic rich markov model for web prefetching. B. Nigam et al. [13] used the concept of dynamic nested markov model to predict next accessed web page whose analysis is done on different schemes of prefetching and caching [14]. M.T. Hassan et al. [17] presented Bayesian Models for two things like learning and predicting key Web navigation patterns. Instead of modelling the general problem of Web navigation they focus on key navigation patterns that have practical value. Mamoun A. Awad et al. [18], analyzed and studied Markov Model with all-Kth Markov Model for web prediction. They proposed a new modified Markov Model to alleviate the issue of scalability in the number of paths. Poornalatha G et al. [19] presented a paper to solve the problem of predicting the next web page to be accessed by the user based on the mining of web server logs that maintains the information of users who access the web site. Section 2, describes the Markov Model and Dynamic Nested Markov Model, Section 3, describes the experimental results and finally section 4 describes the future work and conclusion.

3.Prediction Models

Markov Model is compact, simple, expressive and based on a well-established theory. Markov Model is widely used to model user navigation sessions. In first-order Markov Model, each state corresponds to a web page and each pair of viewed web page corresponds to state transition. Two artificial state i.e. start and final, are incorporated in the model. In second-order Markov Model, each state corresponds to sequence of two viewed web pages and so on.

A) Generation Second-Order Markov Model

Markov Model is widely used to model user navigation sessions. Two artificial states i.e. start and final are incorporated in the model. In first-order Markov Model, each state corresponds to a web page and each pair of viewed page corresponds to state transition. In second-order Markov Model, each state corresponds to sequence of two viewed web pages and so on. Figure 5.2 shows the corresponding transition diagram of Hypertext Probabilistic Grammar. Hypertext Probabilistic Grammar (HPG) is a four-tuple $<V, \Sigma, S, P>$, $V=\{A_1, A_2, A_3\ldots.\}$ is set of non-terminals , $\Sigma=\{a_1, a_2, a_3\ldots.\}$ is set of terminal symbol, S is start symbol , P is set of production rule.

$$p(S \rightarrow a_i A_i) = \alpha \frac{|A_i|}{\sum_{j=1}^{|V|}|A_j|} + (1-\alpha)\frac{|SA_i|}{\sum_{j=1}^{|V|}|SA_j|}$$

$$p(A_i \rightarrow a_j A_j) = \frac{|A_i A_j|}{|A_i|}$$

$$p(A_i \rightarrow \mathrm{F}) = \frac{|A_i F|}{|A_i|}$$

$$p(F \rightarrow \varepsilon) = 1$$

if $\alpha = 1$ the probability of state being in a start production is proportional to the total number of times the state was visited in the collection of navigation sessions. Therefore, when $\alpha = 1$ the probability of a start production is proportional to the number of times the corresponding state was visited, implying that the destination node of a production with higher probability corresponds to a state that was visited more often. The parameter can take any value between 0 and 1, providing a balance between the two scenarios described above. As such, α gives the analyst the ability to tune the model for the search of different types of patterns in the user navigation. Finally, the probability of a transitive production is assigned in such a way that it is proportional to the frequency with which the corresponding link was traversed.

Table 1 shows the example of collection of training and testing user navigation sessions. T1 to T5 are the transaction ID. There are five web pages P_1 to P_5.

Table 1: Collection of User Navigation Sessions.

Transaction ID	Training Sessions
T1	P_2, P_3, P_1, P_5
T2	P_2, P_1, P_3, P_4, P_5
T3	P_1, P_2, P_5
T4	P_1, P_5, P_4
T5	P_1, P_2, P_4

Transaction ID	Testing Sessions
T1	P_3, P_1, P_4
T2	P_1, P_5, P_4
T3	P_4, P_5

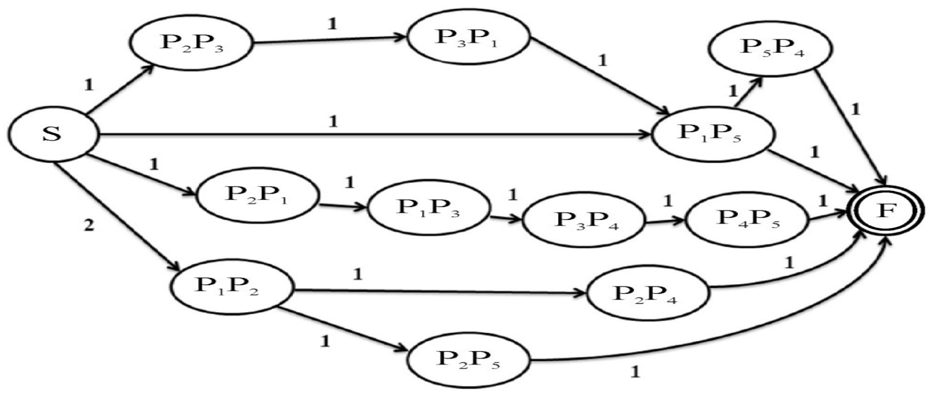

Fig. 1: Second-order Markov Model corresponds to training file of table 1.

Fig. 1 shows the second-order Markov Model corresponds to the training file of table 1. The model is represented with the hypertext weighted Matrix. Here states are the sequence of two viewed web page, S is the start state and F is final state.

Table 2 shows Hypertext weighted Matrix for second-order Markov Model. if the state exists then the weight shows the count of number of times the sequence occurs in the training file otherwise it will be 0.

Table 2: Hypertext Weighted Matrix.

2^{nd} Order Markov Model	P_1	P_2	P_3	P_4	P_5
{P_1, P_2}	0	0	0	1	1
{P_1, P_3}	0	0	0	1	0
{P_1, P_5}	0	0	0	1	0
{P_2, P_1}	0	0	1	0	0
{P_2, P_3}	1	0	0	0	0
{P_2, P_4}	0	0	0	0	0
{P_2, P_5}	0	0	0	0	0

{P_3, P_1}	0	0	0	0	1
{P_3, P_4}	0	0	0	0	1
{P_4, P_5}	0	0	0	0	0
{P_5, P_4}	0	0	0	0	0

Table 3: Prediction results of second-order Markov Model.

Last web page	Original web page	Predicted web page	Correct web page
{P_3, P_1}	P_4	P_5	X
{P_1, P_5}	P_4	P_4	√
{P_4}	P_5	Cannot be predicted	No Result

Table 3 shows the prediction results of second-order Markov Model. The prediction accuracy is 33 %. Test session P4 cannot be predicted with the help of second-order markov model. The coverage of the model becomes 50% because it cannot predict the single state.

B) Generation of Dynamic Nested Markov mode

In Dynamic Nested Markov Model the higher-order Markov Model is nested inside the lower-order Markov Model [13, 14]. DNMM uses the link list structure for storing the information of web page. DNMM is same as Markov Model with some changes so that the efficiency of model can be enhanced. This model is dynamic in nature means the addition and deletion of state can be done easily. This model uses the node structure to store the web page. All the information of a particular web page is stored in a node of that web page. In this model, only one node per web page is created. Node is a dynamic data structure rather than just name of the web page. Each node contains name of web page and an in-link-list. The inlink list is a link list in which each node contain name of a previous web page from which the current web page is traversed, count that shows number of times current web page is traversed from previous web page and an outlink list that keep track of all the corresponding to that previous web page. Outlink list is a linked list whose each node contains name of next web page and its count. Now this data structure keeps track of all the previous web pages and all the next web pages corresponding to each previous web page of the current node. In third order model every node contain data upto third order and in fourth order each node contain data up to fourth order, but number of nodes are always constant. In this way, we can model user navigation sessions in highly structured and efficient way.

In DNMM each web page is represented by unique node. All the information regarding a particular web page is stored inside that node up to the n-order model. Figure 2 shows the node structure of web page Wx for second-order DNMM. W_1, W_2…. are the second-order inlinks to the web page Wx and from Wx the corresponding second-order outlinks are shown in figure 2.

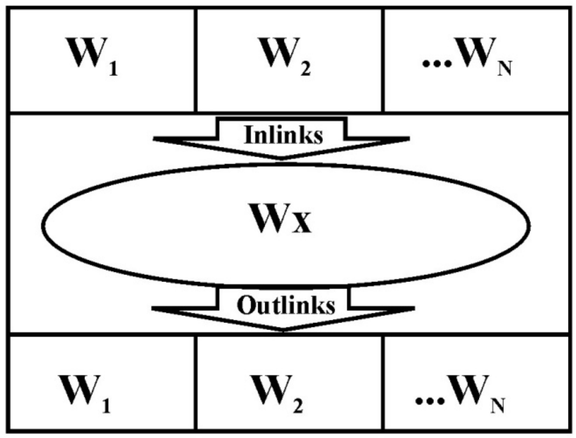

Fig. 2: Node structure of Second-Order Dynamic Nested Markov Model.

Figure 3 shows the first-order DNMM corresponds to the table 1. S and E i.e. start and end is the two artificial states incorporated in model. First-order DNMM construction starts with traversing the first user navigation sessions of web log. If the traversed web page does not exists then the node of that web page is created and corresponding f_inlink count and f_outlink list which has next web page name and count will be created. Firstly, P_2, P_3, P_1, P_5 will be traversed because it is the first sessions and there is no node existing previously. Node of P_2 web page will be created, f_inlink count set to 1 and f_outlink list which contains P_3 as next web page name and count is 1. Next P_3 web page is created, f_inlink count set to 1, and f_outlink list will have P_1 as next web page name and count is set to 1. Now P_1 web page will created, f_inlink count is set to 1 and f_outlink has P_5 as next web page and count is 1. This way the first-order DNMM is created. The first-order Model DNMM is almost like the traditional first-order Markov Model.

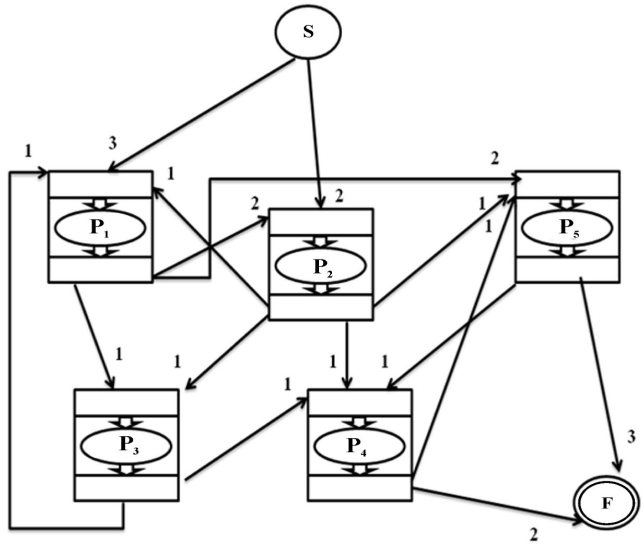

Fig. 3: First-Order Dynamic Nested Markov Model corresponds to the training file of table 1.

Figure 4 shows second-order Dynamic Nested Markov Model corresponds to the table 1. In the second-order DNMM history of previous web page is stored inside the node of first-order model. The model construction starts with the first user navigation sessions i.e. P_2, P_3, P_1 and P_5. As it is the first user navigation sessions of the training file and no node exists till now, so the P_2 node will be created. inlink of P_2 node is created and named as S because P_2 is starting web page and S is considered to be its previous web page. Now the outlink list of inlink S will be updated as P_3 web page and its count set to 1. Now P_3 is traversed which is also not exists in the model so it is created. P_2 is created as inlink and count set to 1 because P_2 is traversed one time from P_3. Now the outlink P_1 for inlink P_2 is created and its count set to 1. Similarly node P_1 will be created. When P_5 node is created, its inlink P_1 is created which has E as outlink. This way the first user navigation sessions has been modelled. Second user navigation sessions is P_2, P_1, P_3, P_4, P_5. Its first web page is P_2 which is already exist in the model. It has S as its inlink so count will be incremented by 1 and will become 2. Outlink P_1 will be checked in inlink of S which does not exist, so it is created and count set to 1. Web page P_1 is traversed and node P_1 is present in model. Node P_1 has an inlink P_3. Another inlink P_2 is created and its outlink P_3 is created. Similarly, web page P_3 is traversed and its node is updated. When web page P_4 is traversed its node does not exist, so it will be created and updated similarly. Remaining user navigation sessions are modelled in same way.

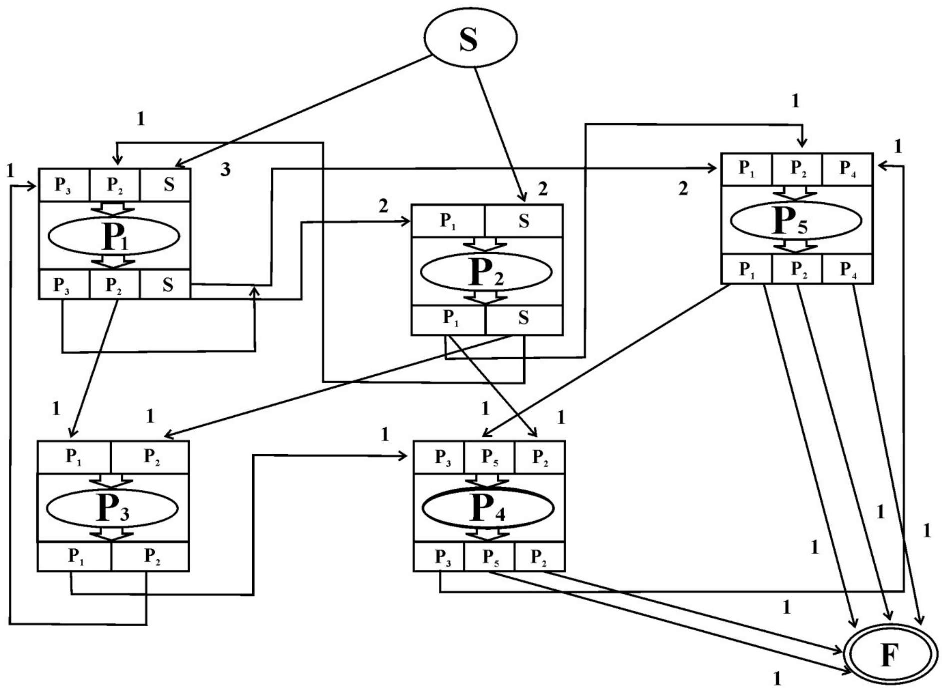

Fig. 4: Second-order Dynamic Nested Markov Model corresponds to example of table 1.

Table 4: Prediction results of second-order DNMM.

Last web page	Original web page	Predicted web page	Correct web page
$\{P_3, P_1\}$	P_4	P_5	X
$\{P_1, P_5\}$	P_4	P_4	√
$\{P_4\}$	P_5	P_5	√

Table 4 shows the Prediction results of second-order DNMM. There are three test sessions out of which two were predicted right. The prediction accuracy is 66% because it can predict the single state also. That is why its coverage is 100%.

4. Experiment Result:

Data Sets:

The experimental data set were obtained from three different data sources. First experimental weblog data is collected from Cuboid Pvt. Ltd., Indore. Second weblog data is MSNBC, collected from UCI repository and can be downloaded from https://archive.ics.uci.edu/ml/datasets/MSNBC.com+Anonymous+Web+Data. Third experimental weblog data were obtained from the authors Jose Borges and Mark Levene and downloaded from http://www.cs. washington.edu/homes/map/adaptive/download.html. Two months web log data are obtained from this website for experiments. The web site is http://machines.hyperreal.org. It is given that web site receives approximately 10000 requests per day from around 1200 users.

Table 5 summarizes the characteristics of the web log data sets. The training data set and testing data set characteristic are given below.

Table 5: Summary of Web Log Datasets (A) Training Data Set (B)Testing Data Set.

Training set Data Set	Web pages	Session	Requests
Cuboid	29	3798	9566
MsnBc	92	3234	12378
HyperReal	36	2567	8821

(A)

Testing set Data Set	Web pages	Session	Requests
Cuboid	29	3471	7894
MsnBc	92	2931	9087
HyperReal	36	2130	7845

(B)

Evaluation Parameters:

Various orders of Markov Model and Dynamic Nested Markov Model on three different datasets have been evaluated. Evaluation parameters are Model Generation Time, Prediction Time, Prediction Accuracy and Coverage.

(i)Model Generation Time

Generation time is defined as the time required by prediction model for modelling the training file. Fig. 5 shows the model generation time of various order of Markov Model. Model generation time is measured in millisecond. It is also observed that generation time depends on the number of states generated by model. Second-order markov model generated more number of states as compare to first order markov model so it take more time to generate.

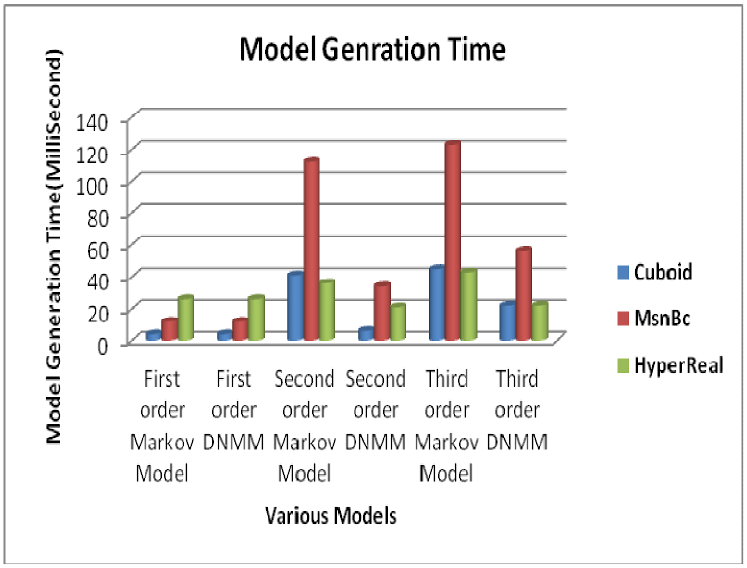

Fig. 5: The time taken in millisecond for generation various order Model

(ii)Prediction Time

Fig. 6 shows prediction time taken by various order of model which is measured on the testing file. The prediction time is depends on the other factor like the network traffic, server etc. But the major time is of model which is observed in millisecond. The prediction time will be affected by the number of web pages in the web log file also. The result shows that the prediction time will increase with respect to number of web pages. The first-order Markov Model and DNMM take same time to predict the next accessed web page. As move towards higher-order of Markov and DNMM models, DNMM takes less prediction time as compared to Markov Model.

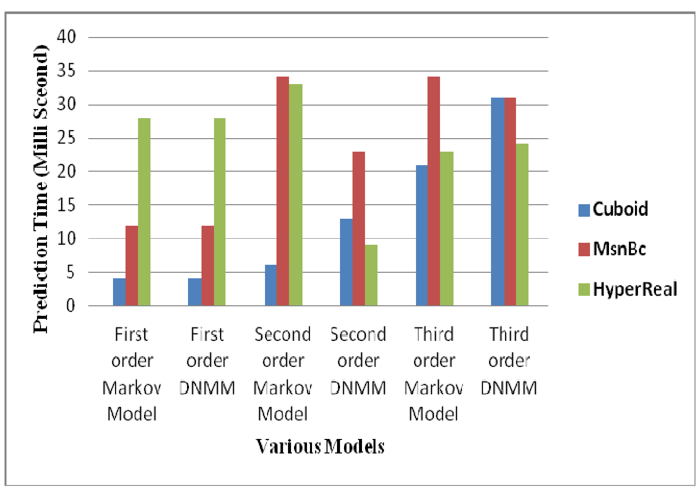

Fig. 6: The prediction time taken in millisecond for various order of model.

(iii)Prediction Accuracy

Prediction accuracy is very important parameter for the prediction model. It measures the accuracy of the prediction model applied for testing file and calculated as:

Prediction Accuracy = (Number of correct prediction) / (Number of test sessions)

Where number of correct prediction are the number of test user navigation sessions which are correctly predicted and number of test sessions are the total number of test sessions on which prediction is performed. Correct predictions are find out by comparing original and predicted web pages, those predicted web pages which are equal to original web pages are consider as correct prediction.

As shown in figure 7, first-order Markov Model and DNMM gives same prediction accuracy. As move towards higher-order of Markov and DNMM models, DNMM gives high prediction accuracy as compared to Markov Model.

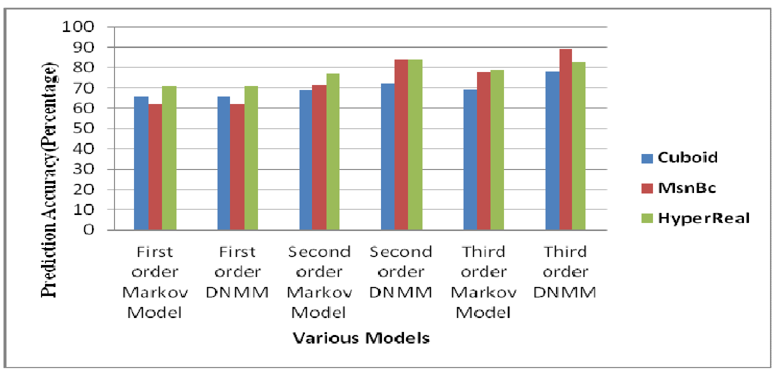

Fig. 7: The prediction accuracy for various order models.

(iv)Coverage

The coverage of the model is defined as the ratio of number of times model is able to predict to number of requests in test set. In case of first-order Markov Model the coverage of state is 100% and in second-order Markov Model the coverage of the state is 50% and as we move for the higher order Markov Model the coverage will be less. For example in second-order Markov Model where each state is a set of two web pages, if we want to predict that after web page W1which will be the next web page accessed then model will fail to predict because it is not having single state web page. In the DNMM in each order of the model, coverage will be 100%.

Conclusion

Various models have been analyzed in this work on the basis of Generation Time, Prediction Time Prediction Accuracy and coverage. DNMM gives better prediction accuracy as compare to the Markov Model. The coverage of the DNMM is better than the Markov Model.

References

[1] Federico Michele Facca, Pier Luca Lanzi, Mining interesting knowledge from weblogs: a survey, Data & Knowledge Engineering, 53 (2005) 225–241, doi:10.1016/j.datak.2004.08.001

[2] S. Zhang and Z. Shi “Research and Application in Web Usage Mining of the Incremental Mining Technique for Association Rule” IFIP International Federation for Information Processing, Springer Boston, 2005.

[3] R. Popa, T. Levendovszky “Marcov Models for Web Access Prediction” 8th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, Nov 2007.

[4] V.Valli Mayil, “Web Navigation Path Pattern Prediction using First Order Markov Model and Depth first Evaluation” International Journal of Computer Applications (0975 – 8887) Volume 45– No.16, May 2012

[5] F. Khalil, J. Li and H. Wang “A Framework of Combining Markov Model with Association Rules for Predicting Web Page Accesses”, Proc. Fifth Australasian Data Mining Conference (AusDM2006), vol. 61, pp 177-184, 2006.

[6] J. Borges, “A Data Mining Model to Capture User Web Navigation. PhD Thesis”, University College London, London University, 2000.

[7] J. Borges and M. Levene, “A Clustering-Based Approach for Modelling User Navigation with Increased Accuracy”, Proc.Second Int’l Workshop Knowledge Discovery from Data Streams, pp. 77-86, Oct. 2005.

[8] Borges, J. and Levene, M. 2005. Generating Dynamic Higher-Order Markov Models in Web Usage Mining. Proc. Ninth European Conf. Principles and Practice of Knowledge Discovery in Databases (PKDD). A. Jorge, L. Torgo, P. Brazdil, R. Camacho, and J. Gama, Eds. (Oct. 2005), 34-45.

[9] Mathias G´ery, Hatem Haddad, “Evaluation of Web Usage Mining Approaches for User’s Next Request Prediction” November 2003, New Orleans, Louisiana, USA. Copyright 2003 ACM.

[10] Khalil, F., Li, J. and Wang, H. Web Path Recommendations Based on Page Ranking and Markov Models. Proc. Seventh Ann. ACM Int’l Workshop Web Information and Data Management (WIDM ’05). (2005), 2-9. 2006.

[11] Siriporn Chimphlee, Naomie Salim, Mohd Salihin Bin Ngadiman and Witcha Chimphlee, Using Association Rules and Markov Model for Predit Next Access on Web Usage Mining” 2006, Advances in Systems, Computing Sciences and Software Engineering, Pages 371-376

[12] Nizar R. Mabroukeh, and C. I. Ezeife, “ Semantic-rich Markov Models for Web Prefetching, in the proceedings of the 2009 IEEE International Conference on Data Mining (ICDM) Workshops (Workshop on Semantic Aspects in Data Mining (SADM'09)), Miami Florida, December 6-9, 2009, pp. 465–470.

[13] B. Nigam, S. Jain, "Generating a New Model for Predicting the Next Accessed Web Page in Web Usage Mining", in Proceedings of the 3rd International Conference on Emerging Trends in Engineering and Technology (ICETET ’10), Washington, DC, USA, pp.485-490, 2010.

[14] Bhawna Nigam and Suresh Jain “Analysis of Markov Model on Different Web Prefetching and Caching Schemes” ICCIC, 28-29 Dec—2010 (2010 IEEE International Conference on Computational

Intelligence and Computing Research (ICCIC)), Tamilnadu College of Engineering Coimbatore, ISBN: 978-1-4244-5965-0. Digital Object Identifier: 10.1109/ICCIC.2010.5705732

[15] http://archive.ics.uci.edu/ml/datasets/MSNBC.com+Anonymous+Web+Data

[16] J. Borges and M. Levene, Data Mining of User Navigation Patterns. Web Usage Analysis and User Profiling, B. Masand and M. Spiliopoulou, Eds. LNAI 1836, 2000, 92-111. Springer.

[17] M.T. Hassan, K.N. Junejo, and A. Karim, "Learning and predicting key Web navigation patterns using Bayesian models," in Proceedings of Int. Conf. Comput. Sci. Appl. II, Seoul, Korea, pp. 877–887, 2009.

18] Mamoun A. Awad and Issa Khalil, "Prediction of User's Web-Browsing Behavior: Application of Markov Model," IEEE Trans. Syst., Man, Cybern. A., Syst., Humans, Volume 42, No. 4, pp., Aug. 2012.

[19] Poornalatha G, Prakash S Raghavendra, "Web Page Prediction by Clustering and Integrate Distance Measures" IEEE/ ACM Trans. Syst., Man,Cybern. A, Syst., Humans, Volume 44, No. 2, pp., Sep. 2012.