IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 4, Issue 4, Aug - Sept, 2016
ISSN: 2320 – 8791 (Impact Factor: 2.317)
www.ijreat.org

A Hybrid Technique Based On Web Usage And Content Features For Web Page Recommendation

Nandita Tiwari[1], Dr. Amit Shrivastava[2]
Research Scholar, Professor
Computer Science and Engineering, SIRTS Bhopal, India

Abstract— By the increase of digital media in today's era, internet surfers are also increasing. This attracts most of the computer science researcher to develop highly responsive system. So web mining come in existence to provide solutions in this field. Out of various research area of web mining this paper works on recommendation of next page of the web user as this reduce server load, increase fast user response, etc. Here two features of the web mining was used first is weblogs which help in finding the rank of the pages as per the user pattern. Web content is second feature of the work which helps in building the ontology of the pages keywords and then understanding the user keyword for surfing. So use of these feature combination improve the prediction accuracy of the work. Experiment was performed on real dataset. Result shows that proposed combination of features and techniques increases the values of different parameters as compare to previous existing methods.

Index Terms — Text categorization, Ontology, Information Extraction, Text Analysis, feature extraction, clustering.

I. INTRODUCTION

These days, the web is an important source of information retrieval, and also the users accessing the web are from different backgrounds. Analyzing web log files to extract useful patterns is called web usage mining. The usage information about users are recorded in web logs. Web usage mining approaches include association rule mining, clustering, sequential pattern mining etc. Web recommendation model is needed, to facilitate web page access by users. Thus, the interest has been increasing rapidly, in the analysis of user's behaviour on the Web. This increase stems from the realization that added value for Web site visitors are gained through easier access to the required information at the right time and in the most suitable form; not by merely through larger quantities of data on a site. Estimates of Web usage expect the number of users to climb up to 945million by 2004 [12]. It is very difficult to keep up with the rapid development of the computer technologies as, the majority of these users sre non – expert; but they recognize that the web is an invaluable source of information for their everyday life. The pace, at which information becomes available online, is also accelerated by the increasing usage of the Web. In various surveys of the Web, e.g. [13], it is estimated that over 600 GB of pages change per month and roughly one million new pages are added every day. A new Web server, providing Web pages, is emerging every two hours. More than three billion Web pages are available online now-a-days, making it, almost one page for every two people on the earth [14]. In the above, one notices the emergence of a spiral easing number of users causing an increase in the quantity of online information, a tracing even more users, and so on. This pattern is responsible for the 'explosion' of the Web, which results in the frustrating phenomenon known as 'information overload' to Web users.

Web usage mining is valuable in many applications like E-businesses, online marketing, etc. The important information can be gathered from the customers visiting the site by the use of this type of web mining. By this, an in-depth log to complete analysis of a company's productivity flow can be achieved. To direct the company to the most effective Web server for the promotion of their product or service, E-businesses depend on this information .

A large number of users access web sites all over the world, with the growing popularity of the World Wide Web. When user accesses a websites, large volumes of data such as addresses of users or requested URLs are automatically gathered by Web servers. It is very important to collect it in access log because many times user repeatedly accesses the same type of web pages and the record is maintained in log files. Web access pattern, are the series of accessed web pages, which is helpful to find out the user behaviour. Through this behaviour information, we can find out the accurate user next request prediction that can save the time of the user and decrease the server load by reducing the browsing time of web page. There has been a lot of research work done in the field of web usage mining, "Future request prediction", in recent years,. The main motivation of this study is to know that what research has been done on Web usage mining in future request prediction.

In Web prediction, main challenges are in both pre-processing and prediction. Pre-processing challenges include choosing optimum sliding window size, handling large amount of data that cannot fit in the computer memory, and extracting/seeking domain knowledge, identifying sessions. Prediction challenges include long training/prediction time, low prediction accuracy, and memory limitation.

II. RELATED WORK

Weighted Links Rank Algorithm

A modification of the standard page rank algorithm is given by Ricardo Baeza-Yates and Emilio Davis [13] named as weighted links rank (WLRank). This algorithm provides weight value to the link based on three parameters i.e. tag in which the link is contained, length of the anchor text and relative position in the page. Simulation results show that the results of the search engine are improved using weighted links. The best attributes in this algorithm be the length of anchor text. Relative position is not so result oriented, which reveal that physical position does not always in synchronism with logical position. Future work in this algorithm includes, weight factor tuning of every term for further evolution.

Eigen Rumor Algorithm

There is a challenge for service provider to provide good blogs to the users because the number of blogging sites is increasing day. Page rank and HITS are very promising in providing the rank value to the blogs but they have a few limitations if we apply them directly to the blogs. The rank scores of blog entries is often very low, as decided by the page rank algorithm, so it cannot allow blog entries to be provided by rank score according to their importance. To resolve these limitations, a Eigen Rumor algorithm [14] is proposed for ranking the blogs. This algorithm allots a rank score to every blog by weighting the scores of the hub and authority of the bloggers; which depends on the calculation of eigen vector.

Distance Rank Algorithm

Ali Mohammad Zareh Bidoki and Nasser Yazdani [15] proposed an intelligent ranking algorithm known as distance rank. It is based on reinforcement learning algorithm. Here, the distance between pages is known as a punishment factor. And the ranking is done on the basis of the shortest logarithmic distance between two pages. Finding pages with high quality and more quickly with the use of distance based solution is the advantage of this algorithm. The Limitation of this algorithm is that, if new page is inserted between the two pages, the crawler should perform a large calculation to calculate the distance vector.

Time Rank Algorithm

An algorithm named as Time Rank, is proposed by H Jiang et al.[16], for improving the rank score by using the visit time of the web page. To know about the degree of importance to the users, authors have measured the visit time of the page after applying improved and original methods of web page rank

algorithm. This algorithm utilizes the time factor to increase the ranking accuracy of the web page. Due to the methodology used in this algorithm, it can be assumed to be a combination of link structure and content. The results of this algorithm are very satisfactory and in agreement with the applied theory of developing the algorithm.

TagRank Algorithm

An algorithm named as TagRank [17] for the social annotations based web page ranking on is proposed by Sun Rong-Shuang, Chen Chen, Shen Jie, Zhang Hui, He Kun and Zhu Yan. This algorithm calculates the heat of the tags by using time factor of the new data source tag and the annotations behaviour of the web users. This algorithm provides a better authentication method for ranking the web pages. The results of this algorithm are very accurate and this algorithm index new information resources in a better way. Future work in this direction can be, to utilize co-occurrence factor of the tag to determine weight of the tag; and this algorithm can also be improved by using semantic relationship among the co-occurrence tags.

Relation Based Algorithm

Andrea Sanna,Fabrizio Lamberti and Claudio Demartini [18] proposed a relation based algorithm for the ranking the web page for semantic web search engine. Various search engines are presented, by using relations of the semantic web, for better information extraction. This algorithm proposes a relation based page rank algorithm for semantic web search engine that depends on information extracted from the queries of the users and annotated resources. Results are very encouraging on the parameter of time complexity and accuracy. Further improvement in this algorithm can be the increased use of scalability into future semantic web repositories.

Query Dependent Ranking Algorithm

Lian- Wang Lee, Jung- Yi Jiang, ChunDer Wu and Shie-Jue Lee [19] have presented a ranking algorithm for search engine which is query dependent. In this approach, the similarities between the queries, are measured by a simple similarity measure algorithm. For every training query, a single model for ranking is made, with corresponding document. Documents are extracted and ranked depending on the rank scores, whenever a query arises, calculated by the ranking model. The combination of various models of the similar training queries is the ranking model in this algorithm. Results of experiment show that ranking algorithm dependent on query is better than other algorithms.

Ranking and Suggestive Algorithm

M Vojnovic et al. [20] have proposed a ranking and suggestive algorithm, based on user feedback, for popular items. User feedback is measured by using a set of suggested items. Depending on the preferences of the user, items are selected. The aim of this technique is to measure the correct ranking of items based on the actual and unbiased popularity. Proposed algorithm can suggest the search query by various techniques. This algorithm can also be used in social tagging system for providing tag suggestion. In this algorithm various techniques are studied for ranking and suggesting popular items and their results are provided based on their performance.

III. PROPOSED WORK

In order to prepare the model for the page prediction different feature of the web mining are used such as structure and logs. Different steps which are required for the prediction are:

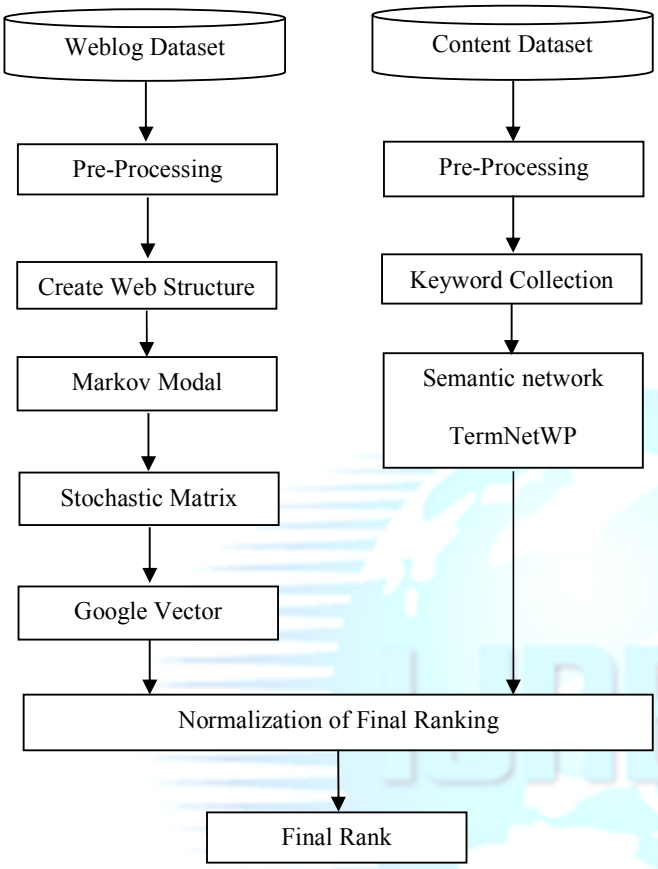

Fig.1: Working modal of Markov Based Page Prediction

Dataset

Web log feature is used for the construction of markov modal. In order to utilize this, one has to pre-process this data as well. The Web user dataset is a collection of the time, sequence of pages, date, etc. Now in order to work on this data, pre-processing is required for taking the required data.

This can be understood as user session is required for training the Markov Modal, and other information is not required. For this divide whole dataset in same pattern and collect it in the form of matrix where each entry is in a separate row.

2006-02-01 00:08:43 1.2.3.4 – GET/ classes/cs589/papers.html – 200 9221 HTTP/1.1 naya.cs.depaul.edu

$$X[n] \leftarrow D$$

4.2.1 Pre-Processing

Here H is a matrix with n rows and D is the rough dataset. Now find each word before '/' and consider it as a separate page. For example get/ classes / cs589 then pages are get, classes, cs589. In this manner each row of the X matrix is restore with new values that are sequence of pages and represent the session.

$$X[n] \leftarrow \text{pages}\ (\ X[n]\ )$$

Where pages is a function that perform all these activities.

Assign Page Number

Now As the purpose of using this dataset is to train the Markov modal, which is a mathematical modal therefore it need to give specific number to each page of the whole website. This is done by putting unique word in a separate matrix L from the X matrix then assign number to each after this replace each word sequence of the matrix by a number.

Let L = {get, set, met, classes, cs589, cs585, cs584, cs582 papers, result}

$$X[n] = \text{Session}(L[m], X[n])$$

Here session is a function that compares and replaces the row pages with the corresponding number. In this manner finally data is in the X vector again but in the number form representing the pages.

Create Web Structure

In this step web site structure feature is develop for the generation of stochastic matrix. Here by the use of the website web log sequence web structure is prepared. This can be understand as the let sequence be {P1, P3, P4}, {P1, P2, P3, P6} then from page P1 direct link is present towards P3 and P2. In similar fashion page P3 is connect with P4 and P6.

Markov Modal

Here in this step Frequent Web Access Pattern are generate with the web log feature obtained from the website. As markov modal generate different order for various predictions. Out of different markov order this work generate third order can be understand as let the three pages are present in the web log in consecutive manner then this act as the pattern in web log. So counting of those patterns is set as the frequency value. This is shown in table 1. Let P1 and P2 are two pages then sequence for P2→P1 in the web log will give the markov order for it.

P1,P2,P5,P6,P9
P8,P4,P5,P3,P2
P6,P9,P5,P6
P1,P2,P5
P5,P6,P9
P5,P6,P1,P2,P5,P9
P1,P9

Now P1, P2, P5 = 3

In this way or in the similar fashion one can operate the same function for the different sequence of the most frequent page view after the particular or group of page.

From the above calculation it can be predict that chance of occurrence of the page P5 is highest as compare to the other pages after the page P2. In similar fashion other values can be calculate.

Stochastic Matrix

Here the dataset is again pre-process for the web log portion in order to get the logs that are use for testing the built model. Pre-Processing steps are similar as done in step 3 of model preparation. The only difference here is that pre-processed logs are break such that each sequence first few pages are in the testing part and the next page after that sequence is store for the evaluation of result.

For testing from above table each web log is divide into two part first two page sequence of the log is consider as the testing phase while other act as the next page, pages after the next page is discard as that is not required for the testing. So the resultant dataset for testing is shown in below table. Where testing act as the input to the model while Next page is to evaluate the output.

Stochastic Matrix

In this step input from web structure as well as markov modal is done. Here one weight matrix is use for the calculation which is taken randomly where each element in the vector is between 0-1. One more parameter is evaluate in this step that is transition matrix P where its value is calculate by finding the ratio between the page Pij where link present between page I and j is divide by the total number of links produce from the page i. So web structure feature plays important role in generating the P matrix. Here d matrix contains 1 value in those page positions where link is not present.

Stochastic Matrix= P +w* d^T+ markov---(1)

In above equation calculation is done for all possible page combinations.

Google Matrix

Here damping factor μ is calculate by

$$\mu = \frac{j}{j+1} \text{----------(2)}$$

Where j = 1……k.

K is total size of path it may be any integer value.

$$\xi = \mu_k .....\mu_2 \mu_1 \text{ ----------(3)}$$

$$\rho_{k-j+1} = \frac{\xi_{k-j+1}}{\xi_{k-j}} \text{ ----------(4)}$$

The damping factors in *Mk* can be computed by means of the forward recurrence.

$$\mu_j = 1 - \frac{1}{1 + \frac{\xi_{k-j+1}}{1-\mu_{j-1}}}, j = 1.....k \text{ ----------(5)}$$

$$\varsigma(\beta) = \sum_{j=0}^{k} \frac{1}{j+\beta} where \beta > 1$$

$$P = \mu \times S \text{ ----------(6, 7)}$$

$$G = \sum_{j=0}^{k} \left((\xi \times S) + P) \times \left(\frac{1}{\varsigma(\beta)} \left(\frac{1}{(1+j)^{\beta}} S^j \right) \right) \right) \text{--(8)}$$

Above equation generate random walk with markov modal. As S is calculate by including the markov modal.

Web Content In this step first Collect the heading of all the Web-pages from the website this then store it in the text file which will be read from that file and utilize the words present in the page for the effective prediction as this represent the user interest.

Pre-Processing: Now keywords are generate from the text file content depend on the frequency of the words present in the file. So some pre-processing is required for the file that is to remove the unwanted words or the stop words from the text file, this can be done by reading a file then put all the words in a matrix H[i, n] where n is the number of total words present in the matrix and i is the page number. Now read the stop words file then put it in the W[m] matrix where m is the number of stop words present. Here one has to compare each word of the H matrix to the W matrix and the matched words are removed from the H matrix file this is like a subtraction of the matrix H by W.

Semantic Network

In this step the semantic network of the keywords are prepare this is term as – TermNetWP. So in order to prepare the semantic network (TermNetWP). H matrix need to utilize as this can be seen as the graph where nodes act as the keywords and page number while edge act as the link between the nodes. Here one has H matrix that contain both the page number as well as the keyword, it is possible that one keyword is present in more than one page. So the keyword is shared by both the pages and the link is made between those pages; one more information maintained is the number of occurrence of the keyword.

Normalization of Final Ranking

In this step rank generate from the google matrix is selected for identifying the pages which are next as per current user path. Here all pages which are next to the current user sequence are use for the semantic network analysis.

Loop 1:4

N←Intersect(G, User_path)

EndLoop

Now all N number of pages keywords are compare with the User_path keywords. So page matching large number of User_path keywords are consider as the next page or prediction page.

IV. EXPERIMENT AND RESULT

In order to predict new web user session page the Data Sets and Pre-processing

This work considered data sets, namely, the unimi data set, from the http://law.dsi.unimi.it/datasets.php one generates artificial by the website creates for this work. In addition to many other items, the pre-processing of a data set includes the following: grouping of sessions, identifying the beginning and the end of each session, assigning a unique session ID for each session, and filtering irrelevant records detail of the dataset is mention in table 2. In this experiments, the cleaning steps and the session identification techniques is done in two modules of clustering and keyword feature vector creation.

Table I *Dataset Summary*

Features	Unimi
Total Session	20,000
Number of Pages	1,382,908
Number of Links	16,917,053
DataSet Time	2004

Table 1. Different properties of DataSet

Accuracy

In this evaluation parameter let us consider a Web_log = {L1, L2, L3……..Ln}. Here L is the particular web page sequences such L1 = (P1, P2, P4, P5, P6, P9). To find the prediction, the part of the log is passed in the system such as (P1, P2, P4) then for this pass correct prediction is P5; if the system generates the P5 value then consider it as the correct prediction otherwise consider it as the incorrect one.

So Accuracy = Rc/ R

Where Rc is the number of correct prediction

R is the total number of steps.

Results

Evaluation Parameter	**Dataset Size**		
	D1	**D2**	**D3**
Precision	0.998073	0.9977	1
Recall	0.998073	0.9322	0.9358

Table 5.1. Results of precision and recall values from proposed work and Markov model.

Evaluation Parameter	**Dataset Size**		
	D1	**D2**	**D3**
Accuracy	92.9982	93.0131	93.5754
Error	7.0018	6.9869	6.4246

Table 5.1. Results of Accuracy and Error values from proposed work and Markov model.

Table 5.2 and 5.2 shows that proposed work accuracy is quite impressive in field of page prediction. As accuracy of 93% is much higher, this reduces the execution time and calculation cost as well. It has also been obtained that with the increase in dataset size for testing satisfaction of the proposed work get increase while previous work also get increase.

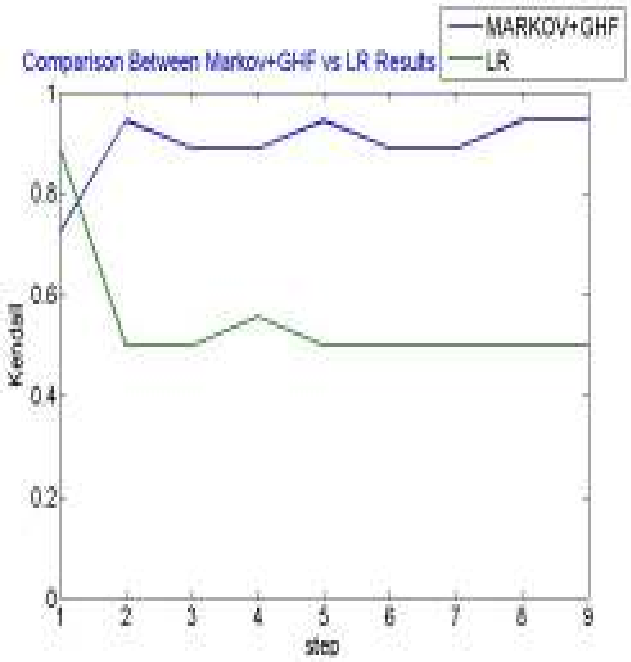

Fig. 2 Kendla Comparison between markov+GHF vs LR.

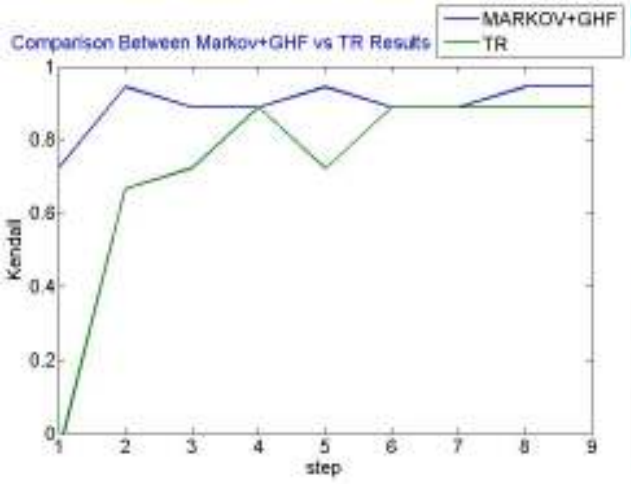

Fig.3 Kendla Comparison between markov+GHF vs TR.

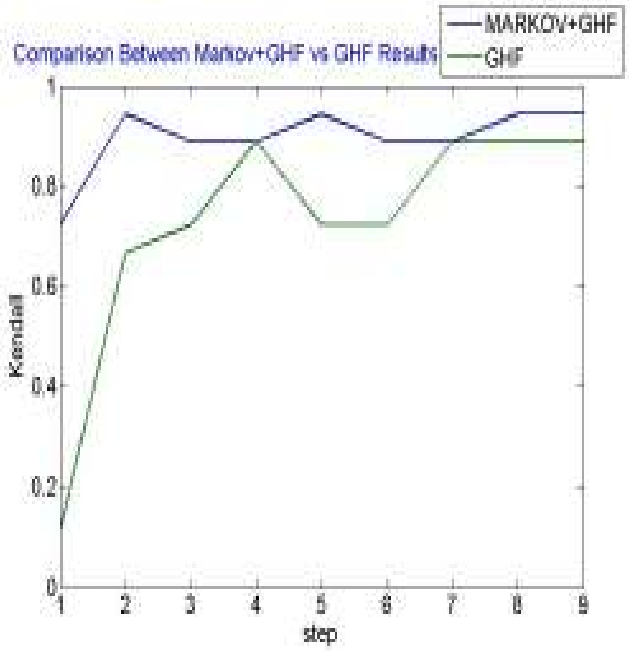

Fig. 4 Kendla Comparison between markov+GHF vs GHF.

Figure 2, 3 and 4 shows that proposed work kendlla values comparison is higher as compare to previous work. It has also been obtained that with the increase in step size for testing satisfaction of the proposed work remain consistent as compare to previous work.

V. CONCLUSIONS

As web mining is highly growing research area for different researchers. This work contributing web mining by proposing an web page recommendation of the user by utilizing the different features and techniques of web mining. As use of Google matrix with markov modal for page ranking increases the rank kendla value. While use of web page content for developing the relation of the user keywords with pages increase the page prediction accuracy as well, it is obtained that highly refined web data of logs and content can reduce the server execution time as well. Work shows that an accuracy of 93 percent was achieved which is highly recommendable. In future work can be increase by introducing more efficient pattern generation algorithm like association rule for further enhancement of accuracy.

REFERENCES

1. Adami, G., Avesani, P. & Sona, D. (2003), 'Clustering documents in a web directory', WIDM'03, USA pp. 66–73.

2. Agrawal, R., Imielinski, T. & Swami, A. (1993), 'Mining association rules between sets of items in large databases', ACM SIGMOD Conference on Management of data pp. 207–216.

3. Agrawal, R. & Srikant, R. (1994), 'Fast algorithms for mining association rules', VLDB'94, Chile pp. 487–499.

4. Chu -Hui Lee , Yu-lung Lo, Yu-Hsiang Fu, "A novel rediction model based on hierarchical characteristic of web site", Expert Systems with Applications 38 ,2011.

5. V. Sujatha, Punithavalli, "Improved User Navigation Pattern Prediction Technique From Web Log Data", Procedia Engineering 30 ,2012.

6. Blomqvist, E., Sandkuhl, K.: Patterns in ontology engineering: Classication of ontology patterns. In: Proceedings of the 7th International Conference on Enterprise Information Systems. (2005) 413-416
7. Gangemi, A.: Ontology design patterns for semantic web content. In: The Semantic Web{ISWC 2005, Springer (2005) 262-276

8. Nasraoui.O and Petenes.C,"Combining Web usage mining and fuzzy inference for Website personalization," in Proc.WebKDD, 2003, pp.37–46.

9. Nasraoui.O and Krishnapuram.R,"One step evolutionary mining of context sensitive associations and Web navigation patterns," in Proc.SIAM Int.Conf.Data Mining, Arlington, VA, Apr.2002, pp.531

10. Deshpande, M., & Karypis, G. (2004). Selective Markov Models for Predicting Web-Page Accesses. ACM Transactions on Internet Technology (TOIT), 4(2), 163-184.

11. Pitkow, J. & Pirolli, P. (1999), 'Mining longest repeating subsequences to predict www surfing', USENIX Annual Technical Conference pp. 139–150.

12. Computer Industry Almanac, http:// www.c-i-a.com

13. Chakrabarti, S: 2000, Data mining for hypertext: A tutorial survey. ACM SIGKDD Explorations, 1 (2), pp. 1-11