International Journal of Current Trends in Engineering & Technology
ISSN: 2395-3152
Volume : 01, Issue : 06 (September – October 2015)

To Enhance Website Performance for Caching with SP-SOM Technique using Progressive Database

Richa Mishra[1], Prof. Yogesh Rai[2]
Shree Institute of Science and Technology, Bhopal

Abstract: *Every Organization need to understand their customer's behavior, preferences and future needs, which depend on past behavior. Web Usage Mining is an active research topic in which user session clustering is done to understand user's activities. In this paper, we use Neural based approach Self Organizing Map for clustering of session as a trend analysis with some parameters. It depends on the performance of the clustering of the number of requests. Here we are using SOM algorithm in Most Frequent Sequential Traversal Pattern Mining called SP-SOM and generated cluster of web data. In this research we establish good prediction with quantity of data and the quality of the results.*

Keywords *– Web Usage Mining; Frequent Sequential Patterns; Sequence Tree; Web Log Data; Web Services; Neural Network; Clustering.*

I. INTRODUCTION

The WWW [2] is an immense source of data that can come either from the Web content, represented by the billions of pages openly available, or from the Web usage, represented by the register information daily collected by all the servers around the world. Web Mining is that part of Data Mining which deals with the extraction of interesting knowledge from the World Wide Web. Web usage mining [4] has many applications, e.g., personalization of web substance, support to the design, recommendation systems, pre-fetching and caching [23]. Kohonen Self-Organizing Maps (SOM) [8, 10] developed by Tuevo Kohonen, a professor emeritus of the Academy of Finland. SOMs learn unsupervised competitive learning. "Maps" is because they attempt to map their weights to conform to the given input data. The nodes in a SOM network attempt to become like the inputs presented to them. The topological relationships between input data are preserved when mapped to a SOM network.

II. RELATED WORK

Prefix Span [1], a more efficient pattern growth algorithm was proposed which improves the mining process. The main idea of Prefix Span is to examine only the prefix subsequences and project only their corresponding suffix subsequences into projected databases. The database projection growth based approach, Free Span [1], was developed. Although Free Span outperforms the Apriority based GSP algorithm, Free Span may generate any substring combination in a sequence. The projection in Free Span must keep all

International Journal of Current Trends in Engineering & Technology
ISSN: 2395-3152
Volume : 01, Issue : 06 (September – October 2015)

sequences in the original sequence database without length reduction.

In SPADE [11], a vertical id-list data format was presented and the frequent sequence enumeration was performed by a simple join on id lists. SPADE can be considered as an extension of vertical format based frequent pattern mining. The discovery of the user's navigational patterns using SOM [8, 10] is proposed by Etminani. Huge amount of information are collected repeatedly by web servers and gathered in access log files. Analysis of server access data can offer important and helpful data. The author used the Coonan's SOM (Self Organizing Map) to preprocessed web logs for extracting the common patterns.

In WUM [14] we find the behavior of user either it is registered or not. If a website requires users to sign in before they can start browsing, it will be very easy not only to differentiate between users but also to identify each single user. The problem arises when a website allows visitors to anonymously browse its content, which is common place. In this paper differentiate between visitors activity as a challenging task in the log using common log format [6].

III. PROPOSED WORK

In this paper, we have used Self Organizing Map (SOM) with frequent sequential pattern. SOM is a type of neural network. In the process of Web Usage Mining [14] to detect user's patterns it is usage as a trend analysis. It depends on the performance of the clustering of the quantity of requests. Here we are using SOM algorithm with SP-SOM (Frequent Sequential Traversal Pattern Mining with SOM) algorithm.

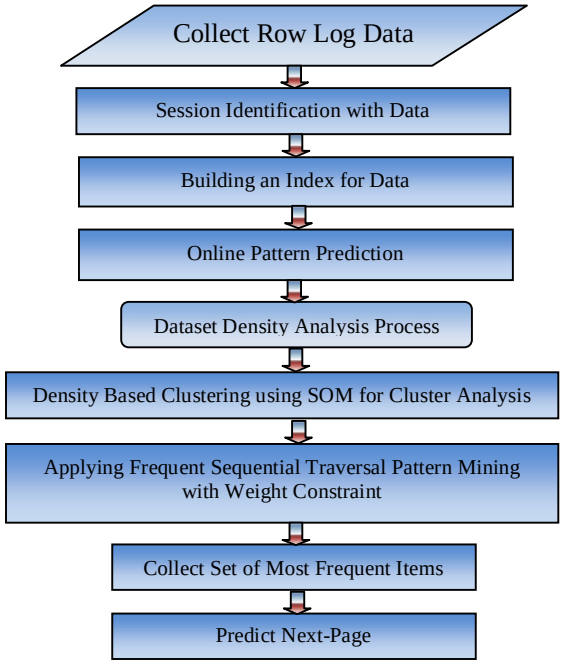

Figure-1: Proposed SP-SOM Approach

The procedure details the transformations essential to modify the data storage with clustered in the Web Servers Log files [6] to an input of SOM. By proceeding this way, first we use SOM algorithm and getting some cluster of web-data. Here we load the web-data cluster, which is almost related to frequent pattern. After that we are applying min-max weight of Page in Sequential Traversal Pattern. Finally we establish good prediction with quantity of results. The figure-1 shows the process of proposed work where it collect the sessional web

International Journal of Current Trends in Engineering & Technology
ISSN: 2395-3152
Volume : 01, Issue : 06 (September – October 2015)

data and applying SOM after preprocessing [7]. Here it mine density based clustering and then find the closed frequent item from the sessions web data for getting useful information.

3.1 SP-SOM Algorithm

The proposed algorithm is used for finding most frequent sequential traversal patterns with clustered index. To handle the ordered problem, the SP-SOM first filtered frequent sequential pattern by using support with min-max or average weight parameter [12] of item. After that it uses neural network algorithm for clustering of index with similarity of object. At last it create most frequent sequential pattern tree [21]. This tree is create less candidate set and also uses to predict next item in caching [23].

3.2 Procedure of Sequential Pattern with SOM Technique

The procedure for constructing the pattern tree in the proposed system is as follows:

Step-1: Collect the web logs of website.

Step-2: Apply preprocessing [7] to get useful sessional web data.

Step-3: Supply input as number of support by the user and checks Min-Max weight or Average weight of page and generate frequent sequential pattern.

Step-4: Apply SOM algorithm in Sessional Frequent Sequential Pattern Item. So here each and every item belongs to at-least one cluster according to similarity.

Step-5: Convert frequent sequential pattern into Frequent Sequential Pattern [18] and generate the Pattern-Tree for next item prediction.

Step-6: Finally establish good cluster items for prediction into the caching to improve the quality of the results and response time.

IV. EXPERIMENTAL RESULT

This paper showing the result of clustered web data using SOM and also used frequent sequential pattern for pre-fetch the next item in cache as on the behavior of similar pattern access of user. The Table-1 showing page details with Support and Min-Max weight range.

S. No.	Page ID	Page Name	Support	Min. Weight	Max. Weight
1	P1	Books	9	2	31
2	P2	Electronics	7	3	7
3	P3	Cloths	7	4	22
4	P4	Jeweler	6	5	9
5	P5	Furniture	6	3	10
6	P6	Toys	1	1	2
7	P7	Root	2	1	3

Table-1: The example of page with weight range

The Table-2 showing the Items details of every page which belongs to page.

International Journal of Current Trends in Engineering & Technology
ISSN: 2395-3152
Volume : 01, Issue : 06 (September – October 2015)

PageItemId	PageId	Item in Page
1	1	Item-1
2	1	Item-3
3	1	Item-4
4	2	Item-1
5	2	Item-1
6	2	Item-2
7	3	Item-1
8	3	Item-2
9	3	Item-1
10	4	Item-1
11	4	Item-1
12	4	Item-4
13	5	Item-1
14	5	Item-4
15	5	Item-2
16	6	Item-1
17	6	Item-1
18	6	Item-3

Table-2: The example of page with item

The Table-3 shows the Running time (in ms) when we having different database record size with different supports.

Support / Size	1%	2%	3%	4%	5%	6%
1033	6817	6708	6723	6505	6474	6708
2040	18408	18111	18252	18158	18376	18470
3050	15865	17953	18565	18764	18451	18487
4010	24310	24688	24927	25205	26365	24949
5030	83413	38079	84630	54116	40731	38391

Table-3 : Running Time (in ms) with different size and different support

The figure-2 shows the Running time (in ms) when we having different record size with different support.

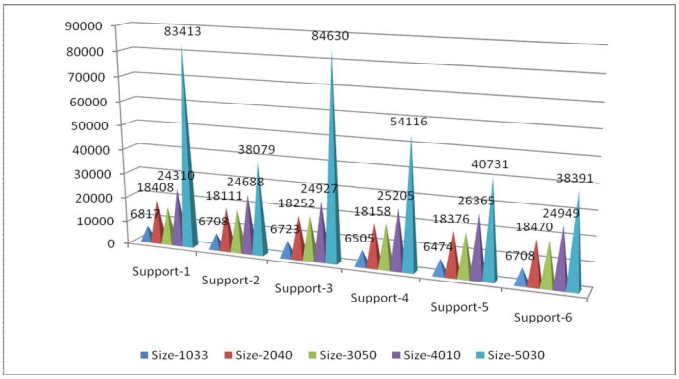

Figure-2: Running Time (in ms) with different size and different support

The Table-4 shows the probability of occurrence of each item with different support.

	Item-1	Item-2	Item-3	Item-4
Support-3%	0.44	0.19	0.15	0.22
Support-6%	0.50	0.17	0.08	0.25
Support-10%	0.43	0.20	0.13	0.23
Support-20%	0.33	0.00	0.33	0.33

Table-4: Probability of Items with different support

The Figure-3 shows the probability of occurrence of each item with different support.

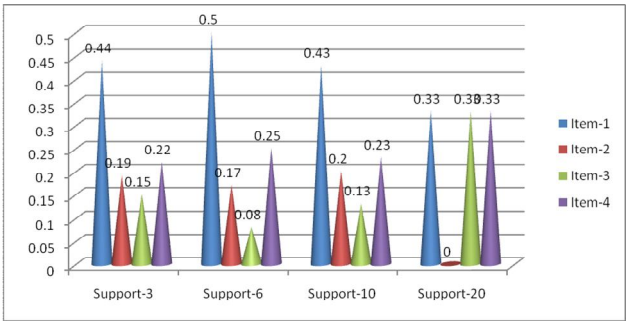

Figure-3: Probability of items with different support

International Journal of Current Trends in Engineering & Technology
ISSN: 2395-3152
Volume : 01, Issue : 06 (September – October 2015)

The Table-5 shows Info-Gain of Item with different support.

	Item-1	Item-2	Item-3	Item-4
Support-3%	07.02	06.08	05.51	06.51
Support-6%	12.00	10.34	07.17	12.00
Support-10%	07.84	06.97	05.81	07.35
Support-20%	01.00	01.00	00.00	01.00

Table-5: Info-Gain of items with different support

The Figure-4 shows Info-Gain of Item with different support.

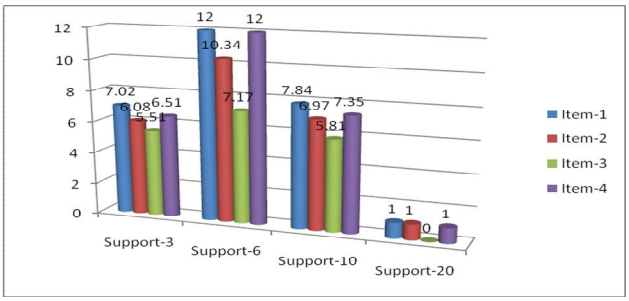

Figure-4: Info-Gain of items with different support

The Table-6 shows the comparison between WSpan and SP-SOM Algorithm with different support. Here record size 5030 is taken in the database.

Support→	**1%**	**2%**	**3%**	**4%**	**5%**	**6%**
WSpan	83413	83210	84630	83397	84645	83990
SP-SOM	82009	83100	80745	54116	40731	38391
Improvement Execution Time (in ms)	2%	0%	5%	35%	52%	54%

Table-6: Comparison of WSpan and SP-SOM Algorithm with different support (By using Record-Size 5030)

The Figure-5 shows comparison between WSpan and SP-SOM algorithm.

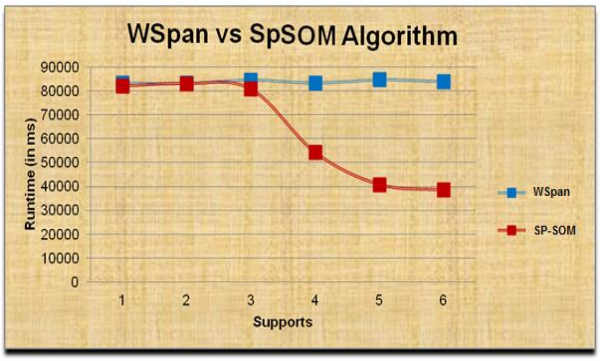

Figure-5: Comparison of WSpan and SP-SOM Algorithm with different support

The figure-5 showing the comparison between WSpan [26] and SP-SOM algorithm. If the support either 1 or 6 the execution time of SP-SOM algorithm is less. Thus proposed SP-SOM algorithm is more efficient.

V. ANALYSIS AND PERFORMANCE EVALUATION

In this section, we present performance study over various datasets (e.g. 1000, 2000, 3000, 4000 and 5000 sessions) and also with different support (e.g. 3, 6, 10 and 20). The experimental results explored for the performance of SP-SOM with a recently developed algorithm, WSpan [1], which is the fastest algorithm for mining sequential patterns. The main purpose of this experiment is to demonstrate how effectively the sequential traversal patterns with min-max weight constraint can be generated by incorporating a support and weight page with clustering. First, we shows how

the number of sequential traversal patterns can be adjusted through user allocate weights, the efficiency in terms of runtime of the SP-SOM algorithm, and the quality of sequential traversal patterns. Second, we show that SP-SOM has put related items in cache. Third we are using web services which provide automatically update min-max weight of every page in every fifteen days. It is also decrease back and forth time while finding next page from cache because it also store related page prior in cache [23].

VI. FURTHER EXTENSION

SP-SOM algorithm basically focuses on sequential pattern mining with average weight constraint uses a weight range to adjust the number of sequential traversal patterns with the clustering of session. SP-PM can be extended by considering levels of support and/or weight of sequential traversal patterns with number of clustering. We can also extended by using Distributed Weblog. There are many areas just like parallel sequential pattern, grouping of similar type of users in distributed servers.

VII. CONCLUSION

This proposed research just begins to touch on the possibilities of SOMs with the frequent sequential pattern mining. One of the main limitations of the traditional approach for mining sequential traversal patterns is that weight of every page is updated manually but here we updated automatically using web services. Second fully database scan is done while find the next item in traditional approach. Here we clustered the items so that clustered items are only scan not whole database. Third we use min-max weight and support of every page so that every page having different importance. So it is enough to perform extremely computationally expensive operations in a relatively short amount of time for finding next page prediction. .

REFERENCES

[1] R. Moriwal, V. Prakash (2013), *"An efficient Algorithm for finding frequent Sequential traversal Patterns from Web Logs Based on Dynamic Weight Constraint"*, *Proceedings of the Third International Conference on trends in Information, Telecommunication and Computing*. vol. 150, (Springer Science + Business Media New York 2013.

[2] Etzioni, O. (1996), *"The world-wide Web: quagmire or gold mine? Communications of the ACM"*, 39 (11).

[3] Kosala, R and Blockeel, H. (2000), *"Web mining research: a survey. SIGKDD Explorations"*, July, 2 (1).

[4] A. Guerbas et al.(2013), *"Effective web log mining and online navigational pattern prediction"*, Knowl. Based Syst.(2013), http://dx.doi.org/10.1016/j.knosys.2013.04.014.

[5] J. Ayres, J. Gehrke, T. Yiu, and J. Flannick,(2002), *"Sequential Pattern Mining*

using A Bitmap Representation", SIGKDD'02, 2002.

[6] J.R. Punin, M.S. Krishnamoorthy, and M.J. Zaki. (2001), *"LOGML - Log Markup Language for Web Usage Mining, in WEBKDD Workshop 2001: Mining Log Data across All Customer TouchPoints"* (with SIGKDD01), San Francisco, August, pp. 88–112.

[7] Zhang Huiying, Liang, Wei. (2004), *"An intelligent algorithm of data pre-processing in Web usage mining"*, (WCICA), Proceedings of the World Congress on Intelligent Control and Automation, vol.4, p3119-3123.

[8] Shyam M. Guthikonda, *"Kohonen Self Organizing Maps"*, Dec 2005.

[9] J. Pei, J. Han, and W. Wang, *"Mining Sequential Patterns with Constraints in Large Databases"*, ACM CIKf, Nov. 2002.

[10] Mishra et al., *"Web Usage Mining Using Self Organized Map"*, International Journal of Advanced Research in Computer Science and Software Engineering Vol. 3, Issue(6), June - 2013, pp. 532-539.

[11] M. Zaki, *"SPADE: An efficient algorithm for mining frequent sequences. Machine Learning"*, 2001.

[12] Parag Pande, Ravindra Gupta (2010), *"An Efficient Algorithm for Finding Users Behavior by Weight Alignment of Sequential Traversal Patterns"*, International Journal of Computer, Information Technology & Bioinformatics (IJCITB) ISSN:2278 7593, Volume-1, Issue-2.

[13] Julija pragarauskaitė, (Vilnius, 2013) , *"Frequent pattern analysis for decision making in big data Doctoral Dissertation Physical Sciences"*, Informatics (09 P) prepared at Institute of Mathematics and Informatics of Vilnius University in 2008 – 2013.

[14] P. Britos, D. Martinelli, H. Merlino, R. García Martínez, "Web Usage Mining Using Self Organized Map", PhD Computer Science Program, of La Plata National University. Software & Knowledge Engineering Center, Buenos Aires Institute of Technology, Intelligent Systems Laboratory, University of Buenos Aires , Argentina, 2007.

[15] S. Vijayalakshmi, V. Mohan, S. Suresh Raja, *"Mining of Users Access behavior for Frequent Sequential Pattern from Web Logs"*, International Journal of Database Management Systems (IJDMS) Vol.2, No.3, August 2010.

[16] Priyanka Makkar, Payal Gulati, Dr. A.K. Sharma, *"A Novel Approach for Predicting User Behavior for Improving Web Performance"*, International Journal on Computer Science and Engineering (IJCSE), Vol. 02, No. 04, 2010, 1233-1236.

[17] Panida Songram, *"Efficient Mining of Top-K Closed Sequences"*, Journal of Convergence Information Technology Volume 5, Number 5, July 2010.

[18] Gopala Niranjan, Dr.R.B.V. Subramanyam, Dr.V.Khanaa, *"An Efficient System Based On Closed Sequential Patterns for Web Recommendations"*, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 4, pages 26-34, May 2010.

[19] Mahdi Esmaeili and Fazekas Gabor, *"Finding Sequential Patterns from Large Sequence Data"*, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 1, No. 1, Pages 43-46, January 2010.

[20] Jinlin Chen, Member, IEEE, *"An Up-Down Directed Acyclic Graph Approach for Sequential Pattern Mining"*, IEEE Transactions on Knowledge and Data Engineering, Vol. 22, No. 7, pages 913-928, July 2010.

[21] D. Vasumathi, A. Govardhan, *"BC-WASPT: Web Access Sequential Pattern Tree Mining"*, International Journal of Computer Science and Network Security (IJCSNS), Vol.9, No. 6, June 2009.

[22] S. Taherizadeh, N. Moghadam, *"Integrating Web Content Mining into web usage mining for finding patterns and predicting users behaviors"*, International Journal of Information Science and Management, Vol. 7, No.1, June 2009.

[23] Qingqing Gan, Torsten Suel, *"Improved Techniques for Result Caching in Web Search Engines"*, ACM, pp. 20-24, 2009.

[24] Cui Wei, Wu Sen, Zhang Yuan and Chen Lian-Chang, *"Algorithm of mining sequential patterns for web personalization services"*, ACM SIGMIS Databases, vol. 40, No. 2, pp 57-66, May 2009.

[25] Hua Jiang,Dan Zuo, Xin Hu, Yong-Xin Ge, Bin Han, *"UAP-Minar: A Real-Time Recommendation Algorithm Based on User Access Sequences"*, 7th Int'l Conf. on Machine Learning and Cybernetics, Kunming, 12-15 July 2008.

[26] Unil Yun and John J. Leggett, *"WSpan: Weighted Sequential Pattern Mining in Large Sequence Databases"*, Proc. Of the Third Int'l Conf. on IEEE Intelligent Systems, pages 512-517, Sep. 2006.

[27] Ezeife, C. and Lu, Y. , *"Mining Web Log Sequential Patterns with Position Coded Preorder Linked WAP-Tree"*, International Journal of Data Mining and Knowledge Discovery (DMKD) Kluwer Publishers, pp.5-38, 2005.

[28] R. Baraglia, F. Silvestri, *"An Online Recommender System for Large Web Sites"*, in Proceedings of ACM/IEEE Web Intelligence Conference (WI'04), China, September 2004.