Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290

Frequent Sequential Traversal Pattern Mining for Next Web Page Prediction

Prabhat Kumar Sahu
Research Scholar, Rabindranath Tagore University, Bhopal
e-mail : jppr.sah@gmail.com
Dr. Rajendra Gupta
Rabindranath Tagore University, Bhopal
e-mail : rajendragupta1@yahoo.com

ABSTRACT

The web mining is a broad research area emerging to solve the issues that arise due to the WWW phenomenon. The Web mining research is a converging research area from several research communities, such as Databases, Information Retrieval and Artificial Intelligence. This work overview the most important issue of Web mining, namely sequential traversal patterns mining. In this paper, calculation of Weight and Support of every page is checked to know the importance of the web page and applied the Frequent Sequential Traversal Pattern Mining with Self Organizing Map (FSTSOM) algorithm. The performance of the proposed algorithm shows that the complete set of patterns runs considerably faster as compared to WAP Tree and FS-Tree algorithms.

Keywords : **Pattern Mining, Web Page Prediction**

Date of Submission: Dec 03, 2021 Date of Acceptance: Dec 21, 2021

I. INTRODUCTION

The World Wide Web (WWW) has grown in the past few years from a small research community to the biggest and most popular way of communication and information dissemination. Every day, the WWW grows by roughly a million electronic pages, adding to the hundreds of millions already on-line. A billion of web pages' increase with the rate of million pages per day increase. WWW serves as a platform for exchanging various kinds of information, ranging from research papers, and educational content, to multimedia content and software [1,2].

The continuous growth in the size and the use of the World Wide Web imposes new methods for processing these huge amounts of data. Because of its rapid and chaotic growth, the resulting network of information lacks of organization and structure. Moreover, the content is published in various diverse formats. Due to this fact, users are feeling sometimes disoriented, lost in that information overload that continues to expand. Issues that have to be dealt with are the detection of relevant information, involving the searching and indexing of the Web content, the creation of some Meta knowledge out of the information which is available on the Web, as well as the addressing of the individual users' needs and interests, by personalizing the provided information and services. A review is done for various algorithm for Data Exfiltration Prevention Techniques [21].

II. MOTIVATION

The research study focuses on the novel approach of Frequent Sequential Traversal Pattern Mining with Self Organizing Map (SOM) for data mining text patterns. The major confines of the traditional approach for mining patterns is that weight of every page is updated manually, but by proposed method it is updated automatically using web services. The objectives are as follows:

1. The formation of cluster of the items to reduce the whole data search
2. Calculation of Weight and Support of Every Page to check the importance of the web page
3. Performance evaluation of the proposed algorithm to analyze the next page prediction.

III. SEQUENTIAL TRAVERSAL PATTERNS WITH WEIGHT CONSTRAINT

This section presents the concept of sequential traversal patterns with weight constraint, and show their importance. In this proposed work, web pages of traversals are assigned with weights to show their importance. For example, when users traverse web site, they may have different interest in each page, and therefore stay for different times. Web pages can be assigned with a weight standing for the user stay time, frequency of pages, content of pages and type of web site [3, 5-6].

The proposed work generalizes the mining problem to the case where pages of traversals are given such weights showing their importance. The weights are taken into account in the measurement of support, the ratio of traversals which contains a candidate pattern. If a page of traversal has a weight which doesn't between the weight ranges then it is removed from session of user and treated as an outlier, and cannot consider for the support. For example, when users visit web site, they may traverse through a page very fast to another page, or do another work for a long time during web site visit. This type of page visit is not useful and consider as an outlier because the page is not attentively read by the user [4].

Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290

2.2 FREQUENT SEQUENTIAL PATTERN MINING

The proposed method is used for Sequential Pattern Mining. This approach uses Self Organizing Map (SOM), which is a kind of neural network. It is used for trend analysis to identify customer patterns in the process of Web Usage Mining. It depends on the performance of the clustering of the amount of requests. Here, SOM is used with FSTPM (Frequent Sequential Traversal Pattern Mining) algorithm to find more frequent sequential traversal patterns [7,8].

The existing algorithm access whole database multiple times. The proposed method clusters the session data using neural network algorithm which is called Self Organizing Map (SOM). It clusters the data according to similarity.

2.3 SOM ALGORITHM FOR GETTING CLOSED DATA FOR CLUSTERING

Self Organising Map havingmapping steps starts with initializing the weight vectors. In it, a sample vector is selected arbitrary and the map of weight vectors is searched to find the best weight to represents that sample. The steps of algorithm is as given below :

1. Node weights are defined for each node
2. Vector formation is done randomly from the set of training data.
3. Each node is treated to calculate for which weights are most likely to the input vector. The final node is commonly known as **Best Matching Unit** (BMU).
4. After that the neighbourhood of the BMU is calculated.
5. The final weight is rewarded with becoming more like the sample vector. The nighbour also become more like the sample vector. The closer a node is to the BMU, the more its weights get altered and the farther away the neighbor is from the BMU, the less it learns.
6. Repeat 2nd step for a number of N iterations

III. RESULT ANALYSIS

In this part of study, we presented the performance over various data sets. The proposed algorithm is implemented in ASP.Net with C# on a 3.0 GHz Core2 Duo system with 8 GB memory, running Microsoft Windows Operating System.

3.1 ABOUT DATA SET

The synthetic data set is used for evaluating the performance of the web recommendation system [10,11]. The data set taken from weblog.csv from kaggle.com. The used synthetic data set has a collection of web access sequence and it is spitted into two parts:

(1) Training data set: It is used to design the web recommendation system based on mined frequent sequential patterns on it and

(2) Test data set: It is used to test the designed web recommendation system. At first, the pattern tree is constructed by using the training data set and then, the proposed web recommendation system is evaluated with test data set.

The report of the experimental results on the performance of FSTSOM in comparison with a recently developed algorithm by WSpan, 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 weight constraint can be generated by incorporating a weight page, weight of sequence with a support. First, it shows how the number of sequential traversal patterns can be adjusted through user assign weights, the efficiency in terms of runtime of the FSTSOM algorithm, and the quality of sequential traversal patterns. Secondly, shows that FSTSOM has good scalability against the number of sequence transactions in the data sets with clustering [12,13].

The analysis of FSTSOMalgorithm is similar to *FP-growth* with minor improvement in weight gain. At first, given an FSTP-tree *T*, Weight Range WR, and Average Weight Range of Session AWR, mining the frequent sequential traversal patterns with weight constraintfrom *T* with traversal strategy. If *T* only contains a single path of FSTP-treein which each node only has a single child, then gets directly the frequent sequential traversal patterns with weight constraint. When *T* is multi-path FSTP-tree, each generate 1-FSTSOMand construct prefix traversal sequence pre. By adding the suffix, generates the frequent sequential traversal pattern.

3.2 EXPERIMENTAL ENVIRONMENT

All the experiments are performed on a 2.4GHz Pentium 4 processor with 512 MB memory, Microsoft Windows 2010. In addition, all the programs are written in Dot Net 2008. The experiments were carried out on real data sets to evaluate the performance of FSTSOMalgorithm. The data sets, which contain several months' worth of click sequence data from two e-commerce web sites. Collecting the same number sequence data of the two data sets, which are divided into the different length sessions, and the average session contains 5-11 pages [14].

The following Table-1.1 showing page details with Support and Min-Max weight range.

Table-1.1 Page Name with its 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	Jewelry	6	5	9
5	P5	Furniture	6	3	10
6	P6	Toys	1	1	2
7	P7	Root	2	1	3

Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290

The following Table-1.2represents the inputs that are given at the later stages of FSTSOM. The table contains Page ID from P1 to P7, with page name: Books, Electronics, Cloths, Jewelry, Furniture, Toys and Root. This page from P1 to P7 has different support with different minimum and maximum weight.

Table-1.2 Anillustration of page with item

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

The followingTable-1.2 showtheitem details of every page which belongs to page, this contains the Page Item Id from 1 to 18 and Page Id from 1 to 6 and different items in page.

The following Table-1.3 contain the Running time (in ms) of FSTSOM,under different database record size with different supports.

Table-1.3: Running Time (in ms) with different size and different support

Size	Supp-1	Supp-2	Supp-3	Supp-4	Supp-5	Supp-6
1000	3810	5708	5723	5505	5474	5708
1500	15404	15111	15252	15158	15376	15470
3000	14862	14953	14565	14764	14451	14487
4000	22310	22688	22927	22205	22365	22949
5000	63483	48079	64630	44116	50731	68391

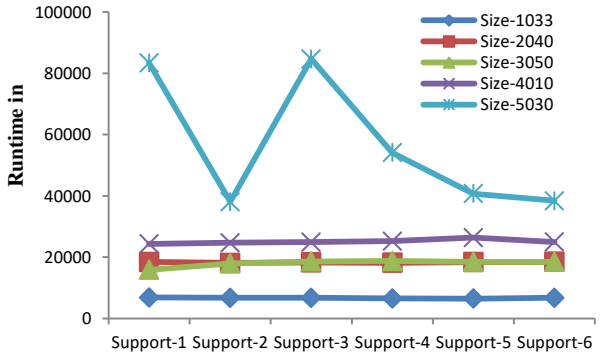

Figure-1.1 : Running Time (in ms) with different size and different support

The figure-1.1 shows the Running time (in ms) of FSTSOM under different record size with different support.Running time (in ms) of FSTSOM varies from record size 1033 to record size 5000 with support 1, support 2, support 3, support 4, support 5 and support 6. The following Graph shows that while taking record size 1000 with support 1 then running time of FSTSOM is 3810 ms, similarly with support 2,3,4,5 and 6 running time of FSTSOM is 5708 ms, 5723 ms, 5723 ms,5505 ms,5474 ms and 5708 ms respectively. In the above mention graph support 4 is taking minimum time and support 1 is taking max time with their corresponding size. Similarly, with record size 2040, FSTSOM running time with support 1 is minimum and with support 6 is maximum with their corresponding size. Similarly, with record size 3050, FSTSOM running time with support 2 is minimum and with support 4 is maximum with their corresponding size [15, 16]. With record size 4010, FSTSOM running time with support 1 is minimum and with support 5 is maximum with their corresponding size. With record size 5000, FSTSOM running time with support 2 is minimum and with support 3 is maximum with their corresponding size, which is desired.

The following Table-1.4 shows the probability of occurrence of each item with different support.

Table-1.4 : Probability of Items with different support

	Item-1	Item-2	Item-3	Item-4
Support-3	0.38	0.22	0.18	0.24
Support-6	0.62	0.20	0.05	0.15
Support-10	0.43	0.20	0.13	0.28
Support-20	0.38	0.05	0.28	0.35

Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290

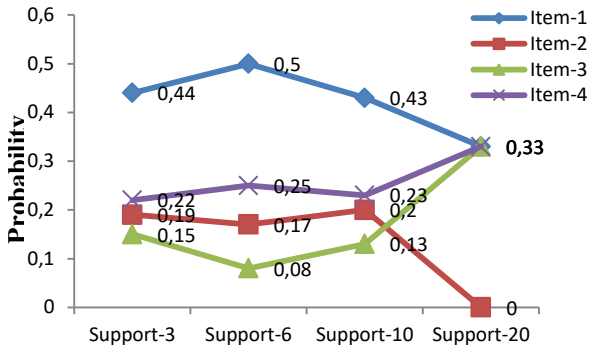

Figure-1.2 : Probability of items with different support

The figure-1.2 shows the probability of occurrence of each item with different support. The graph shows the probability of occurrence of item 1, item 2, item 3, and item 4 with support 3, support 6, support 10, and support 20. The probability of item 1 is highest with support 6, probability of item 2 is highest with support 10, similarly probability of item 3 & item 4 is highest with support 20. The highest probability of item 1, item 2, item 3 and item 4 with support 6, support 10 and support 20 is due to the appropriate data input with these support [17].

The following Table-1.5 contain Info-Gain of Item with different support.

Table-1.5 : Info-Gain of items with different support

	Item-1	Item-2	Item-3	Item-4
Support-3	05.02	08.08	06.51	07.51
Support-6	10.00	12.34	08.17	10.00
Support-10	08.84	07.97	06.81	08.35
Support-20	02.00	02.00	00.50	02.00

The figure-1.3 shows Info-Gain of Item with different support. This graph shows variation in Info-Gain of item1, item 2, item 3 and item 4, with support 3, support 6, support 10 and support 20. The support 6 show highest info-gain of item 1, item 2, item 3 and item 4 as compared with support 3, support 10 and support 20.

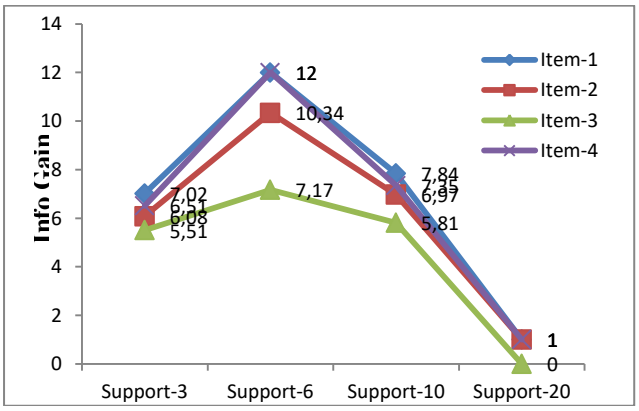

Figure-1.3: Info-Gain of Items with Different Support

The following Table-1.6 shows the percentage improvement in execution time (in ms) and comparison between WSpan and FSTSOM algorithm using record size 5000 in the database with different support.

Table-1.6 : Comparison of WSpan and FSTSOM Algorithm with different support (By using Record-Size 5030)

Support→	**1**	**2**	**3**	**4**	**5**	**6**
Wspan	63413	63210	64630	63397	64645	63990
FSTSOM	62009	63100	60745	44116	30731	28391
Percentage Improvement in Execution Time (in ms)	2.5%	0.8%	3.5%	25%	65%	68%

IV ANALYSIS AND PERFORMANCE EVALUATION

This sectionshowsthe performance analysis over various data sets (eg. 1000, 2000, 3000, 4000 and 5000 sessions) and also with different support (eg. 3, 6, 10 and 20). The report of experimental results on the performance of FSTSOM in comparison with a recently developed algorithm, WSpan [19], 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, it shows how the number of sequential traversal patterns can be adjusted through user allocate weights, the efficiency in terms of runtime of the FSTSOM algorithm, and the quality of sequential traversal patterns. Second, it shows that FSTSOM has put related items in cache [18, 20]. Third using web services which provide automatically update min-max weight of every page in every fifteen days. It is also decreases back and forth time while finding next page from cache because it also stores related page prior in cache.

V. CONCLUSION

The systematic performance study shows that FSTSOM mines the complete set of patterns and is efficient and runs considerably faster than both based WAP Tree and FS-Tree algorithms. This algorithm uses the pre-order linking of header nodes to store all events in the same suffix tree closely together in the link, making the process of searching more efficient. A simple method for allocating position codes to nodes of any tree has also appeared which can be used to decide the relationship between tree nodes without repetitive traversals.

The FSTSOM algorithm is able to quickly determine the suffix of any frequent pattern prefix under consideration by comparing the assigned binary position codes of nodes of the tree. Dealing with unvisited or recently added pages is one of the challenging problems in web page recommendation systems. So, further improvement was

Volume: 13 Issue: 03 Pages: 4983-4987(2021) ISSN: 0975-0290

hybridization of the efficiency of previous algorithm. The target web link is given to the new user by matching the pattern tree with the new user's current web access sequence.

REFERENCES

[1] Kesavan, S., Saravana Kumar, E., Kumar, A., & Vengatesan, K. "An investigation on adaptive HTTP media streaming Quality-of-Experience (QoE) and agility using cloud media", International Journal of Computers and Applications, pp.1–14, 2019

[2] Omar Zaarour, Mohamad Nagi, "Effective web log mining and online navigational pattern prediction", Knowledge Based Systems, ELSEVIER,2017, pp.50-62.

[3] Pan, L. "Research on Personalized Recommendation System Based on Web Mining", Jiangsu University of Science and Tech., 2019

[4] Masseglia,F., Poncelet,P., Teissseire,M.,"Efficient mining of sequential patterns with time constraint: reducing the combinations", Expert systems with applications Elsevier, Vol. 40, N. 3, 29 pp: 2677-2690, 2016.

[5] Charu C. Aggarwal. "Introduction to Special Issue on the Best Papers from KDD", ACM Transactions on Knowledge Discovery from Data (TKDD), Vol.11, pp.4, 2017

[6] Rahul Moriwal, Vijay Prakash, "An Efficient Algorithm for Finding Frequent Sequential Traversal Patterns from Web Logs based on Dynamic Weight Constraint", 2019.

[7] Chen, G.Y. "Research on computer information processing technology in the era of big data", Network Security Technology and Application, No.3, pp.44-52, 2019

[8] Chitraa,V., Antony Selvadoss Thanamani, "An Enhanced Clustering Technique for Web Usage Mining", International Journal of Engineering Research & Technology (IJERT), Vol.1, Issue 4, June-2018.

[9] Ketki Muzumdar, Ravi Mante, Prashant Chatur, "Neural Network Approach for Web Usage Mining", International Journal of Recent Technology and Engineering (IJRTE), Vol.-2, Issue-2, May-2017.

[10] Umapathi,C., Aramuthan, M., Raja,K.,"Enhancing Web Services Using Predictive Caching", International Journal of Research and Reviews in Information Sciences, Vol.-1, No.-3, Sept-2016.

[11] Song Sun, Joseph Zambreno, "Design and Analysis of a Reconfigurable Platform for Frequent Pattern Mining", IEEE, Vol.22, No.9, Sept-2016.

[12] Vijayalakshmi,S., Mohan,V., Suresh Raja,S.,"Mining of Users Access behavior for Frequent Sequential Pattern from Web Logs", International Journal of Database Management Systems (IJDMS) Vol.2, No.3, August 2016.

[13] Mahdi Esmaeili, Fazekas Gabor, "Finding Sequential Patterns from Large Sequence Data", International Journal of Computer Science (IJCSI), Vol.7, Issue 1, No.1, January 2014.

[14] 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, 1233-1236, 2014.

[15] Utpala 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.

[16] 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, pp. 913-928, 2014.

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

[18] Qingqing Gan, Torsten Suel, "Improved Techniques for Result Caching in Web Search Engines", ACM, pp. 20-24, 2015.

[19] WSpan, Rahgozar, M., Lucas, C., Chehreghani, M.H., Mining Maximal Embedded Unordered Tree Patterns. Paper presented at the Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), Honolulu, Hawaii, April 1-5, 2007.

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

[21] Peter S. Nyakomitta, Dr. Silvance O. Abeka, "A Survey of Data Exfiltration Prevention Techniques", Int. J. Advanced Networking and Applications, Volume: 12 Issue: 03 Pages: 4585-4591(2020)