EasyChair Preprint

№ 2370

To Enhancement the Click Stream of Website using GRC Constraints in Web Personalize Clustering Approach

Ganga Singh, Harsh Pratap Singh and Kailash Patidar

January 12, 2020

To Enhancement the Click Stream of Website using GRC Constraints in Web Personalize Clustering Approach

Ganga Singh
SSSIST College, Sehore, India
gangasngh444@gmail.com

Prof. Harsh Pratap Singh
SSSIST College, Sehore, India
singhharshpratap@gmail.com

Prof. Kailash Patidar
SSSIST College, Sehore, India
kailashpatidar123@gmail.com

Abstract: *In the current trends every organization manages work and its data online. Even though e-Commerce website maintaining data online in a distributed form. Online approach is very useful to interact with consumer and seller without any dependency of place and time. Every consumer can select product with any brand without wait for a time and produce the order for purchasing. Most of purchasing the product is done by using the website that produce some navigational or access pattern. This access pattern is used to produce some access rules. The proposed Constraint based Closed Sequential Pattern Mining using Self-Organizing Map Clustering (CBCSPMSC) approach first use some profile and GRC constraints for filtration of data between the duration and occurrence of item gap. Now applying closed pattern technique for minimizes the number of rules generation and execution time. At last SOM clustering technique is applied so that every item belong the cluster for partial database scan not whole data with less execution time.*

Keywords– *Data Mining, Web Usage Mining, Gap, Compactness, Data Stream, Closed Pattern, Personalization, Sequential Pattern Mining, NN-SOM Clustering.*

I. Introduction

The popular medium of publishing is the World Wide Web is very rich source of information gathering. It making sense of data is difficult because publication on the web is largely unorganized. Web mining is also knowledge extraction techniques which discover access patterns from the web. It is divided into three parts, a) web usage mining, b) web structure mining and c) web content mining. The commonly used data mining algorithms are Association Rule Mining (ARM), Sequential Pattern Mining, Clustering, and Classification.

An ARM technique is used to find out the rules between items found in a transaction database. In the context of web usage mining a transaction is a group of web page accesses with an item being a single page access. The problem of discovering sequential patterns is that of finding inter-transaction patterns such that the presence of a set of items is followed by another item in the time-stamp ordered transaction set. The data mining algorithms are used to generate the association rule between the items, sequential pattern of access of items, and clustering of items.

Web Usage Mining (WUM) is the application of data mining techniques to large Web data repositories in order to produce results that can be used in the design tasks and improve response time.

Clustering analysis is used to find out those items that have similar characteristics and group into it. It manages the group of user information or data from Web server logs. It also can facilitate the development and execution of future marketing strategies. It dynamically support or changing a particular site based on a visitor on a return visit. An application of existing data mining algorithms, e.g. discovery of association rules or sequential patterns, the overall task is not one of simply adapting existing algorithms to new data.

The WUM process is a file which having input from web user behavior as a user session files that gives an exact accounting of who accessed the website. It is also having the information just like what pages were requested and in what order, and how long each page was viewed. A user session is a time interval where a web user accesses the pages that occur during a single visit to a website. The web user's access related all the information contained in a raw web server log. It does not reliably represent a user session file for a number of reasons. So that selectively information converts into tabular form and after that apply data mining technique. After getting result it produces some meaningful and useful information.

The *Compactness Constrains* is applied to find the record between the dates so that it can generate latest and interestingness pattern. This constraint needs that the sequential patterns in the sequence database must have the property such that the time-stamp variance (variance of days) between the first and the former transactions in a discovered sequential pattern must not be greater than given period. For example, if our sequence database is from 31/12/2016 to 31/12/2017.

The *Recency constraint* defines when the last transaction happens. This constraint is stated by giving a recency minimum support (r_minsup), which is the number of days left from the starting date of the sequence database. For example, if our sequence database is from 31/12/2016 to 31/12/2017 and if we set r_minsup= 166 then the recency constraint ensures that the last transaction of the discovered

pattern must occur after 31/12/2016+166 days means till 15/06/2017.

Gap Constraint is the difference between two items in transaction occurrences. The gap constraint applies limit on the separation of two consecutive transactions of discovered patterns. For Example the sequence S=ACACBCB and subsequence S0=AB, there are 4 occurrence of S0 in S: (A1, B5) (A1, B7), (A3, B5), (A3, B7). Here only the occurrence (A3, B5) fulfill the 1- gap constraint. So, the subsequence S0 fulfills the 1-gap constraint since at least one of its occurrences does. No occurrence of S0 fulfills the 0-gap constraint and So S0 fails the o-gap constraint.

Self-Organizing Map is a Neural Networking Clustering technique which assigns each item to cluster-id. It is used to find out the number of cluster based on attribute distance value between of object.

II. Related Work

In 2013, Omar Zaarour, Mohamad Nagi [14] proposed an improvement of the web log mining procedure and to the prediction of online navigational pattern. It proposed for session identification using a refined time-out based heuristic. After detect the navigational pattern by using a specific density based algorithm. Now finally, a new proposed method for efficient online prediction is also recommended for applicability.

In 2016, Doddegowda B J, G T Raju, Sunil Kumar S Manvi [16] having approach to personalize the information available on the Web according to user requirements. It adjusts the information/services delivered by a Web to the needs of each user or group of users to find the behavioral patterns.

In 2016, Minubhai Chaudhari and Chirag Mehta [17] proposed a prefixspan algorithm with GRC constraints which generates sequential patterns by using prefix projected pattern growth approach. It uses gap, compactness and recency constraints during sequential pattern mining process. The gap constraint applies limit on the separation of two consecutive transactions of discovered patterns, recency constraint makes patterns to quickly adapt the latest behaviors and compactness constraint make sure reasonable time spans for the discovered patterns.

In 2016, Fan Muhan, Shao Sujie, and Rui Lanlan [18] proposes a method for mining the frequent closed patterns in a sliding window to capture information timely and accurately when new data stream arrives. Here each basic window is used to store the Closed Pattern-tree in sliding window updates which is incrementally updated and delete the infrequent or unclosed patterns.

In 2017, H. Ryang [19] propose a novel algorithm for finding high utility patterns in the list structure over data streams on the basis of a sliding window mode. It avoid the generation of candidate patterns to improve the efficiently works in complex dynamic systems.

In 2017 Bing Zhang and Guoyan Huang [20] proposed an approach to efficiently mine sequential pattern using influential functions based on software execution sequence. It can occur multiple times in a trace, which leads to high cost of time and extreme complexity of the research.

In 2018 Pasi Franti [21] proposed random swap algorithm is also very helpful to solve the clustering by using a sequence management of prototype swaps method.

III. Problem Description

Frequent sequence mining is an important part related to web data and now yet a challenging data mining work. The frequent sequence mining has become an important component of many prediction or recommendation systems. The online stores every time want customer's next item prediction as web pages likely to visit. It also likes to buy together which products. The existing algorithms used for frequent sequence mining could be classified either as exact or approximate algorithms. Accurate frequent sequence mining algorithms usually read the whole database several times, and if the database is very large, then frequent sequence mining is not compatible with limited availability of computer resources and real time constraints. So the problems in the current scenario are –

1) Web data partition is not used some conditional parameters just like profile constraint (Income, Age, and Experience etc.) which support as **conditional parameter** for partition of web data.
2) Many previous sequential mining algorithms show no reflection of **importance of pages** whereas every page has different importance. So the existing methods perform response time is also slow. Website required reasonable approximate methods for analyzing data where the **computation speed** is more important than the precision.
3) Every time the whole database scans for searching the frequent pattern not partial database. At the time of program execution number of cluster required as a input parameter.

IV. Proposed Approach

The proposed work uses GRC constraint for collect the correct data after that apply the Profile matching based on similarity of profile attribute. The proposed approach is used to improve the web response for the online navigational pattern forecasting. It is showing the combination of two approach-Closed Sequential Pattern and Self Organizing Map Clustering for finding frequent sequence traversal pattern with GRC constraint. Here every item belongs one cluster for partial data scan. So at this time this cluster data tree having the web pages of website in proportional sessions can access partially.

This research using a novel approach Profile based **Constraint based Closed Sequential Pattern Mining using SOM Clustering** (CBCSPMSC). It is used as a trend analysis to identify customer's patterns in the process of web usage mining. It depends on the performance of the clustering of the amount of requests. Here the proposed approach using SOM

clustering for accesses the partial data of web data. The data is collected from the website www.getglobalindia.org in the form of web log. Every web log is having 13 parameters eg. IP-Address, Web Browser, Version, OS etc. So at preprocessing the weblog is filter by selected attributes and get only required data after that collecting the required information according to user session-wise for finding the user behavior. In the next step the input support gather by user using Interface, if the item support is greater than and equal to given support then it produces the frequent item using pruning strategy of the item.

Proposed CBCSPMSC Algorithm Description - The following Fig. 1 shows that the process of CBCSPMSC algorithm which generate useful closed sequential pattern using web data.

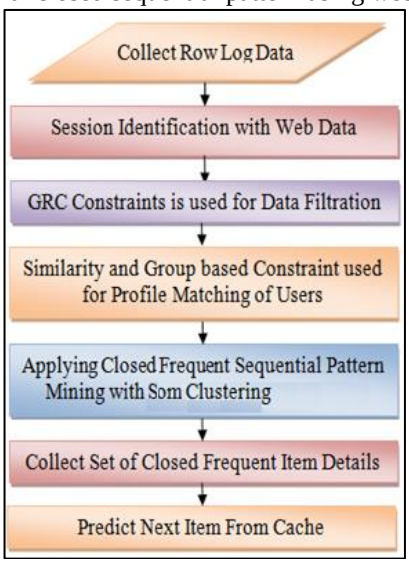

Fig. 1: The Process of Proposed Algorithm

Step 1: Collection of web navigation history of website.
Step 2: Now choose the GRC Constraint for filter the correct data and then apply attributes of profile for similarity matching and input the support value. So here it is matching the attribute similarity to other user navigation pattern group wise. It is also find the frequent pattern using given support threshold value in the model.

TID	Web Access Sequence	Access Date	Compactness
100	ABDAC	02/01/2017	Valid
200	EABECA	24/02/2017	Valid
300	BABFAE	19/04/2017	Valid
400	ABACFC	09/05/2017	Valid
500	EAEBCA	13/06/2017	Valid
600	ACEBEA	21/07/2017	Valid
700	BCCAEB	28/08/2017	Valid
800	AEACBF	30/01/2018	Invalid

Table 1: Compactness of Web Access Sequence

TID	Web Access Sequence	Access Date	Recency
100	ABDAC	02/01/2017	Valid
200	EABECA	24/02/2017	Valid
300	BABFAE	19/04/2017	Valid
400	ABACFC	09/05/2017	Valid
500	EAEBCA	13/06/2017	Valid
600	ACEBEA	21/07/2017	Invalid
700	BCCAEB	28/08/2017	Invalid

Table 2: Recency of Web Access Sequence

TID	Web Access Sequence	Access Date	Gap Constraint
100	ABDAC	02/01/2017	Valid
200	EABECA	24/02/2017	Valid
300	BABFAE	19/04/2017	Valid
400	ABACFC	09/05/2017	Valid
500	EAEBCA	13/06/2017	Invalid

Table 3: Web Access Sequence using Gap Constraints

Step 3: Now select different size of web data set and generate rules using closed sequential pattern of web data set.

TID	Web Access Sequence	Freq. Subsequence
100	ABDAC	ABAC
200	EAEBCA	ABCA
300	BABFAE	BABA
400	ABACFC	ABACC

Item	Count
A	4
B	4
C	3
D	1
E	2
F	2

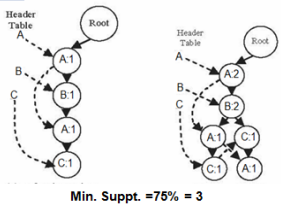

Fig. 2: The web Access Pattern (WAP)- Tree

Table 4 : Sequence database of WAP-tree

TID	Web Access Sequence	Freq. Subsequence
100	ABDAC	ABAC
200	EAEBCA	ABCA
300	BABFAE	BABA
400	ABACFC	ABACC

Table 5: Sequence database of WAP-tree

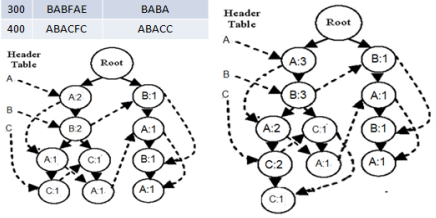

Fig. 3: Steps of WAP-Tree

TID	Web Access Sequence
100	ABDAC
200	EAEBCA
300	BABFAE
400	ABACFC

Table 6: Sample Sequential database of web access sequence

Minimum Support : 3

Frequent Set (FS) are -

A:4 B:4 C:3
AB:3 BA:3
ABA:3

Table 7: Closed Sequential Pattern

The set FS consists of 6 frequent sequential patterns, the closed sequential patterns discovered are:
FS = {A:4, B:4, C:3, AB:3, BA:3, ABA:3} then AB is merge to ABA because both having same count. So that CS is subset of FS.
CS = {A:4, B:4, C:3, BA:3, ABA:3}

Step 4: For clustering the web data is set based on similarity of attributes value in the form of cluster.
Step 5: Put those item in the cache which are having higher frequency.
Step 6: For the next item prediction put some items in the cache which is having higher frequency. Sometimes if next item not in the cache so that it scan the related item from the cluster web data set not the whole data set.

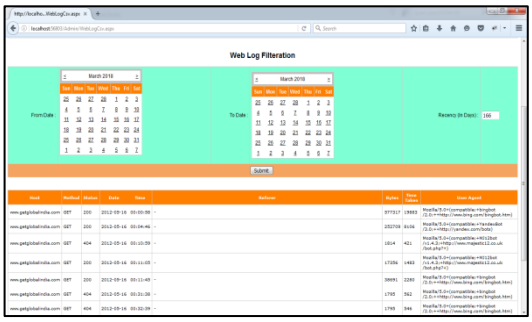

Fig. 4: Web Log filter using Compactness and Recency Constraint

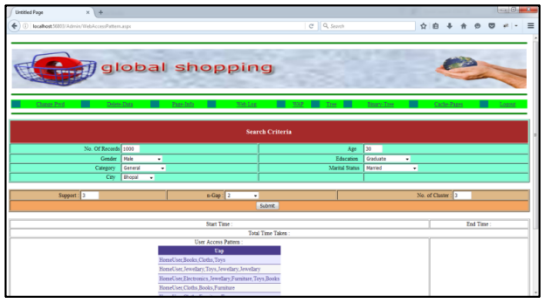

Fig. 5: Web Access Filter with n-Gap and Profile Attributes

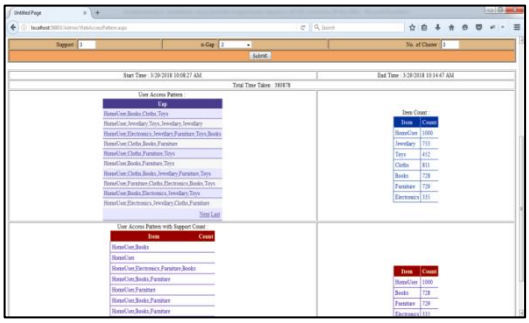
Fig. 6: Web Access Pattern

Pseudo code of CBCSPMSC Algorithm
The Profile based Closed Sequential Pattern Mining using SOM Clustering (CBCSPMSC) approach is applied for discover frequent sequential patterns by using SOM algorithm for producing the cluster of web data set. This cluster is used to access the partial web data set not whose web data set. By using closed sequential, it generates fewer candidates set for generating the rules so that response time is increases. The merging method in realism is reconstructing a small Pattern tree. So at this time this tree having the web pages of website in proportional sessions.

The pseudo code of the proposed algorithm is –
Algorithm (**Constraint based Closed Sequential Pattern Mining using SOM Clustering** (CBCSPMSC))
Procedure: Proposed Method(Support, Attributes, Web Dataset)

```
{
  D = filter the web data based on GRC constraints
  DS = find the web data based on similarity in group of attributes in D
  int i=0;
   do
   { // compute the mean weight of prefix item x
            if (item.weight >= support)
            {   output prefix;  }
            i=i+1;
  }
  while(x.item.count>=i);
 // for finding closed sequential pattern in tree
  int j=0;
  do { //      generate the rules or pattern of given frequent item x
            if (sub-pattern.count <= super-pattern.count)
            {  merge sub-pattern;  }
   j=j+1;
  }
  while(x.item.count>=j);
//clustering of web data set
//generate the cluster of given frequent item
  int k=0;
  int totCluster = CountNoOfItem();
  int sdv = search(smallest distance value);
  int nsdv = searchNearstItem(sdv);
do
  {
            if (sdv <= nsdv)        {   Merge-cluster(sdv);  }
   k=k+1;
  }
  while(k<= totCluster);
SetCacheItem(ClosedPatterTree);
```

V. Result Analysis

All experiments were conducted on a 2GHz Intel Core2 Duo processor PC with 4GB main memory running Microsoft Windows-7. The algorithms were implemented in Asp.Net with C# and were executed using 10% support value. In this experiment a real data set of www.getglobalindia.com is used, which is having click stream data from an e-commerce web store and it has been used widely to assess the performance of frequent pattern mining. This dataset contains sequences of 5701 customers with a total of 56000 purchases.

In this part, presenting functioning analysis on these assorted datasets (eg. 1050, 2050, 3050, 4050 and 6050 sessions) and also with different support (eg. 5, 10, 20, 40, 60 and 80). The description of research results on the functioning of CBCSPMSC in contrast with a newly developed pattern mining algorithm is the fastest pattern mining algorithm. The main reason of this research is to illustrate that, the sequential traversal patterns with weight constraint can be generated by incorporating a support and weight page with clustering is effectively. Initially, showing the number of sequential patterns can be regulated through customers allocate weights, the efficacy in terminology of runtime of the CBCSPMSC algorithm, and the excellence of sequential patterns. Secondly, showing the CBCSPMSC has put related items in the cache. Third, it is using web services which update automatically weight of every page in every week. It also decreases back and forth time while finding next page from cache because it also having related pages prior in cache.

The following Table-8 is showing the Running time (in ms) when the database record size is different with different supports.

Support /Size	5%	10%	20%	40%	60%	80%
1050	6716	6607	6722	6507	6378	6709
2050	18507	18123	18351	18152	18278	18376
3050	15768	17854	18667	18753	18443	18488
4050	24412	24787	24826	25306	26345	24887
6050	82107	81206	80153	76012	70724	67297

Table-8: Running Time (in ms) with different size and different support.

The following Fig. 2 is showing the Running time (in ms) when record size is different with different support.

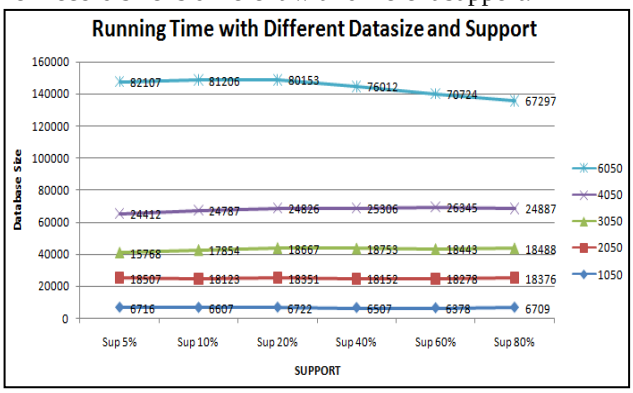

Fig. 7: Running Time (in ms) with different size and different support

The following Table-9 is showing the comparison between WAP-Tree and CBCSPMSC Algorithm with different support. Here it is using record size 6050 in the database.

Support→	5%	10%	20%	40%	60%	80%
WAP-Tree	83413	83210	82630	82102	81645	80790
CBCSPMSC	82107	81206	80153	76012	70724	67297
Percentage Improvement in Execution Time (in ms)	1.57%	2.41%	3.00%	7.42%	13.38%	16.70%

Table-9: Comparison of WAP-Tree and CBCSPMSC Algorithm with different support (By using Record-Size 6050)

The following Table-9 is showing the comparison between WAP-Tree and CBCSPMSC Algorithm with different support. Here it is using record size 6050 in the database.

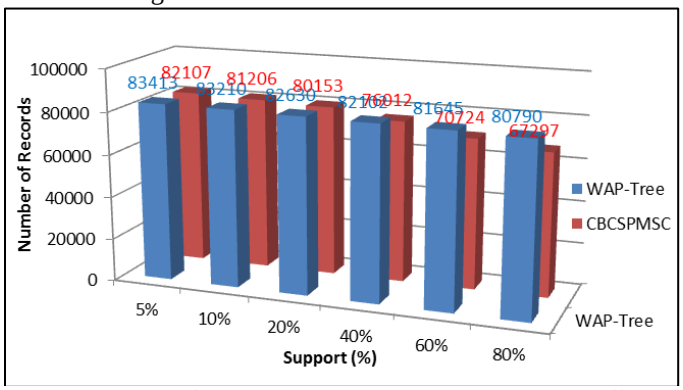

Fig. 8: Comparison of WAP-Tree and CBCSPMSC Algorithm with different support (By using Record-Size 6050)

In data mining most of researcher is uses algorithms to identify previously unrecognized patterns and trends hidden within vast amounts of structured and unstructured information. These patterns are used to create predictive models that try to forecast future behavior.

VI. CONCLUSION

In this research a novel approach CBCSPMSC is proposed for finding closed sequential patterns and scan only partial web data for next item prediction. It filtering large web data by matching user profile similarity based on some attributes. It is having minimum support and each item belongs to cluster so that partial web data is scan. Closed frequent pages are very less and useful rules in the form of clustered by SOM clustering.

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. If web data size is 6050 and support is 10% then percentage improvement in execution time (in ms) is 2.41%. Similarly if the support is 40% then percentage improves in execution time (in ms) is 7.42%. It perform fast response and accurate result because of this it is influential adequate to carry out enormously calculation costly operations in a relatively short amount of time for finding next page prediction.

In future work, other data mining algorithms can be implemented in cloud to efficiency handle big data of many Hospital website in distributed environment for finding any critical diseases with grouping of similar type of customers.

REFERENCES

1) R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases", in Proc. Int. Conf. Very Large Data Bases, pp. 487–499, 1994.
2) M. N. Garofalakis, R. Rastogi, K. Shim, "SPIRIT: Sequential Pattern Mining with Regular Expression Constraints", In Proceedings of 25th VLDB Conference, pp. 223-234, San Francisco, California, 1999.
3) M. J. Zaki, "SPADE: An Efficient Algorithm for Mining Frequent Sequences", Machine Learning Journal, Vol. 42, Issue (1-2), pp. 31-60, 2001.
4) Jian Pei, Jiawei Han and Helen Pinto, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth", In Proceedings of 12th International Conference on Data Engineering, pp. 215-224, Heidelberg, Germany, 2001.
5) Freire J., Kumar B., and Lieuwen D., "WebViews: Accessing Personalized Web Content and Services", In Proceedings of the Tenth International World Wide Web Conference, 2001.
6) Antunes, A. L. Oliveira, "Generalization of Pattern-growth Methods for Sequential Pattern Mining with Gap Constraints", Machine Learning and Data Mining in Pattern Recognition, Third International Conference,

MLDM 2003, Leipzig, Germany, July 5-7, 2003, Proceedings 2003.
7) J. Han, J. Pei, Y. Yin, and R. Mao, "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach", Data Mining Knowledge Discovery, vol. 8, no. 1, pp. 53–87, 2004.
8) J. Pei et al., "Mining Sequential Patterns by Pattern-Growth: The Prefix Span Approach", IEEE Trans. Knowledge Data Eng., vol. 16, no. 11, pp. 1424–1440, Nov. 2004.
9) Yen-Liang Chen, Ya-Han Hu, "The Consideration of Recency and Compactness in Sequential Pattern Mining", In Proceedings of the second workshop on Knowledge Economy and Electronic Commerce, Vol. 42, Iss. 2 , pp. 1203-1215, 2006.
10) T.-P. Hong, C.-W. Lin, and Y.-L. Wu, "Incrementally Fast Updated Frequent Pattern Trees", Expert System Application, vol. 34, no. 4, pp. 2424–2435, May 2008.
11) Krzysztof D., Wojciech K., Marcin S., "Effective Prediction of Web User Behaviour with User-Level Models", Fundamental Informatics , IOS Press , Vol. 89, No. 2-3, pp. 189, 2008.
12) K. R. Suneetha, Dr. K. R. Krishnamoorthy, "Identifying User Behavior by Analyzing Web Server Access Log File", IJCSNS International Journal of Computer Science and Network Security, Vol. 9, No.4, pp. 327, 2009.
13) Dhirendra Kumar Jha, Anil Rajput, Manmohan Singh. & Archana Tomar, (2010) "An Efficient Model for Information Gain of Sequential Pattern from Web Logs based on Dynamic Weight Constraint", IEEE International Conference on Computer Information Systems and Industrial Management.
14) Omar Zaarour, Mohamad Nagi, "Effective Web Log Mining and Online Navigational Pattern Prediction", ELSEVIER, 2013.
15) Jerry Chun, Wensheng Gan, Tzung Pei Hong, "Efficiently Maintaining the Fast Updated Sequential Pattern Trees With Sequence Deletion", IEEE Access - The Journal for Rapid open access publishing, Vol. 2, pp. 1374-1383, 2014.
16) Doddegowda B. J., G. T. Raju, Sunil Kumar, "Extraction of Behavioral Patterns from Pre-processed Web Usage Data for Web Personalization", IEEE International Conference on Recent Trends in Electronics Information Communication Technology, pp. 494-498, 2016.
17) Minubhai Chaudhari, Chirag Mehta, "Extension of Prefix Span Approach with GRC Constraints for Sequential Pattern Mining", International Conference on Electrical, Electronics, and Optimization Techniques, pp. 2496-2498, 2016.
18) Fan Muhan, Shao Sujie, Rui Lanlan, "A Mining Algorithm for Frequent Closed Pattern on Data Stream based on Sub Structure Compressed in Prefix Tree", IEEE Proceedings of CCIS, pp. 434-439, 2016.
19) H. Ryang and U. Yun., "Efficient High Utility Pattern Mining for Establishing Manufacturing Plans with Sliding Window Control", IEEE - Expert Systems with Applications, Vol. 57, pp. 214-231, 2017.
20) Bing Zhang, Guoyan Huang, Haitao He, Jiadong Ren, "Approach to Mine Influential Functions Based on Software Execution Sequence", International Journal of Engineering and Technology, Vol. 11, Issue 2, pp. 48-54, 2017.
21) Pasi Franti, "Efficiency of Random Swap Clustering", Journal of Big Data, Springer, Vol. 5, Issue 13, pp. 2-29, 2018.
22) Liwen Peng, Yongguo Liu, "Feature Selection and Overlapping Clustering-Based Multilabel Classification Model", Hindawi, pp. 1-13, 2018.
23) Dr. S.K. Jayanthi, C. Kavi Priya, "Clustering Approach for Classification of Research Articles based on Keyword Search", IJARCET, Vol. 7, Issuse 1, pp. 86-90, 2018.