INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume3, Issue3)
Available online at www.ijariit.com

Closed Sequential Pattern Mining Using Binary Representation by Same Support Threshold Value

N. Jayaveeran
Khadir Mohideen College
nJayaveeran@gmail.com

S. Ramesh
Khadir Mohideen College
ramesh1237777@gmail.com

Abstract: This paper is Closed Sequential Pattern Mining by Same Support Threshold Value (ClospamSSV). The main intent is to find the closed frequent item sets from a huge sequence database. The ClospamSSV discovers ideally Minimal Closed Sequential Patterns in long sequence databases. The experimental evaluation shows that the proposed algorithm outperforms the Clospam. The Same Support Value is used to prune search spaces according to the given supported threshold value.

Keywords: Pattern Mining, Same Support Value, Sequential Pattern, Closed Sequential.

.

I. INTRODUCTION

The Data Mining is one a part of human life, and especially in all the business. The real problem about the sequential data mining was introduced by Agarwal [1]. It is most powerful to make a good decision for the business. Several types of research are finding new pattern mining algorithm for large sequence databases. There are lots of algorithms created and used for developing the business into next level. But still, the problem is to generate candidates to mine sequential patterns in large sequences of the database and the execution speed of algorithm for the same. The task of discovering the set of all frequent sequences in a large database is demanding as the search space is tremendously large. There are thousands of hidden pattern in sequence databases.

In SPAM, the largest size of patterns is less but it has many closed patterns in it. Initially, the mining closed frequent sequential patterns by Pasquier et al. in ICDT'99[2]. If a pattern and its support value can derive by its super-pattern with the same support value, then it is redundant. The closed frequent sequential patterns are the frequent patterns which do not have any frequent super-pattern with the same support threshold value. In this method, there is no generating those redundant patterns; this mining procedure is more efficient. Mostly Colspan just incorporates some pruning techniques into prefixSpan to find the closed set of frequent sequential patterns.

II. PRELIMINARY CONCEPTS

Let the I={I1, I2,.IM} be a set of items on M length, an item set is a subset of I. A sequence S ={S1, S2,...SN} is N order list of itemsets. The length of S is N, which is the number of the item set, and S is also called N-Sequences. A Sequence a=(a1, a2,...,ai) is a sub-sequence of another sequence b=(b1,b2,...,bj) if there exists a set of indices m1,m2,..mi, n<=m1<=m2<=...<mi<=j. such that a1<=bm1,a2<=bm2,..., ai<=bmi. On the other hand, b is called a super-sequence of a. we can say that b contains a, or a is contained by b. A sequence database D contains a set of sequences, and the support of a sequence S is the number of sequences that contain S. A frequent sequence (or sequential pattern is a sequence with support not less than the minimum support threshold, min_sup.

A closed frequent sequential pattern is a frequent sequence that has no frequent super-sequence with the same support value. [3] This paper define the base table for closed set, CS={S | support(S) >= min_sup and $\nexists$ S' S $\subseteq$ S' and I (D_s) $\subseteq$ I (D_s')}.

III. PROBLEM STATEMENTS OF CLOSED PATTERN MINING

Most of the previously proposed methods carry out the two factors to a certain degree, the property of item ordering in a sequence are not utilized in the mining process. The actual problem in mining sequential pattern mining is to generate candidates. Even, some of the efficient algorithm needs to scan many times the database. The algorithm Colspan is to be generated the lexicographical tree and prefix search tree for closed patterns.

The Pattern-growth method usually starts with a frequent pattern and grows the frequent patterns while traversing the pattern search space using depth-first search. FreeSpan [4], PrefixSpan [5], CloSpan, BIDE and etc are in the same this type. PrefixSpan finds the frequent-1 sequences in the database and builds projected databases for each sequence.

Closed item set mining is used to locate closed itemsets in a transaction database. A number of algorithms were developed for closed itemset mining. A-Close [10] is the first algorithm developed for mining closed item sets. It works on the level-wise frequent item sets using Apriori approach, and mines all minimal generators. In the second step, it computes the closure of all minimal generators. The performance of A-Close degrades due to the huge cost of the off-line closure computation.

IV. PROPOSED APPROACH

A new approach is introduced in this paper, to find closed frequent sequential pattern by the same support threshold value combinations. The proposed approach ClospamSSV is to generate new kinds of the pattern from very large sequence database in less amount of time with compare previous closed frequent pattern mining. In this paper, there is no candidate generation, instead of generating the candidates and scans the database multiple times. The Database scanned only once to generate the base table. The base table has the complete distinct values and the number of times an item presented in all the sequences from the Database SD. The proposed method is (ClospamSSV) for mining the closed sequential patterns from same supported values. In addition, the pruning methods according to different support value data are used. Hence, this should be efficient than previously proposed methods for a smallest number of ideal frequent sequential patterns.

Consider a sample Database from the table1, Sequence S1=<1(3 4)1 4>. The items are found exploring the sequence left to right and the distinct values and its count stored in the same column. The length of the base may vary depends upon the patterns are presented in the Database. The table2 shows that the base table looks alike.

First, scan the sequence database once. While scanning the sequences, the distinct item values have been taken into the record the counts of every different item in the database. Then it can easily get all frequent items or length 1-sequences and its support value by adding the counted values, even if it is in different sequences. The positional information of item i, denoted by POSi, consists of a lot of pairs of (sid,eid), where sid is the sequence identifier and eid is the element identifier. Because sid points out that sequence the item lies in and eid indicates in which order the item lies in the sequence, this representation can reserve the information of item ordering without loss. Consider Table1 as a sample sequence database. And minimum support threshold value 2, the base or base table information of items as shown in table2 and table3. In table3, it clear about length 1-sequences with its support value {1:5, 2:4, 3:6, 4:6,5:1},

TABLE I
SAMPLE SEQUENCE DATABASE D

Sid	**Sequence**
S1	<1(3 4)1 4>
S2	<1 3 1 5>
S3	<3 1 4 (2 3 4)>
S4	<2 2 3>
S5	<(2 3 4)4>

TABLE II
BASE TABLE FOR A SEQUENCE

Sequence 1 < 1 (3 4) 1 4 >					
S1	**1**	**3 4**	**1**	**4**	**Supp.**
1	1	0	1	0	2
3	0	1	0	0	1
4	0	1	0	1	2

TABLE III
BASE TABLE FOR DATABASE D

Eid	**S1**	**S2**	**S3**	**S4**	**S5**	**Sup.**
1	2	2	1	0	0	5
2	0	0	1	2	1	4
3	1	1	2	1	1	6
4	2	0	2	0	2	6
	0	1	0	0	0	1

The candidates are directly from the base table. In tables are explained evidently. The proposed approach directly generates the candidates using the distinct values and presented times in the table. From the Base table, the item "5" is removed since the support corresponding to "5" is 1, which is less than 2. (i.e.) sup (5) <min_sup.

Now consider the projected base table to create candidates, first the item 3 and 4 are considered, because the elements 3 and 4 have the same support values. 3 = {1 1 2 1 1}, 4 = {2 0 2 0 2} and here to find (3) (4) an operation is performed in the base table values of (3) and (4). So 3 = {1 1 2 1 1}, 4= {2 0 2 0 2}. After the ColspamSSV operations (3) (4) = {1 0 2 0 1} support =4.

A. There are three conditions in ColspamSSV as follows:

1. If the element position and the support value are same then keep the same value as output support value (same support value).
2. If any one of the element is zero for the same position, then the output support value is zero (0).
3. The same position of both elements if it is in different values, the output support value is the smallest value from both elements.

Example1: Table 4 is a sample sequence database, referred as D. Minimum support threshold is (min_sup) =2, the base table showing in table5. It shows that, in a single scan of the Database, can build the base table easily. From the base table the length 1-sequences are{A:4, B:3, D:2, E:4, F:2}. The same supported threshold values are {A, E}=4, {D, F}=2. Here B is closed by unique support value in 1-Sequence. AE and DF have to process the operation of the ColspamSSV. That is shown in Table 6. Final closed frequent patterns are {B: 3, AE: 3, DF: 2}. these are the minimal ideal closed frequent sequential pattern by same support value. In fact, the Closed Sequence has the precise same information as Frequent Sequence but includes very less but ideal patterns.

TABLE IV
SAMPLE DATABASE

SID	Sequence
1	<(AF)DEA>
2	<EAB>
3	<E(ABF)(BDE)>

TABLE V
BASE TABLE FOR DATABASED

Eid	**S1**	**S2**	**S3**		**Sup.**
A	2	1	1		**4**
B	0	1	2		**3**
D	1	0	1		**2**
E	1	1	2		**4**
F	1	0	1		**2**

TABLE VI
COLSPAM SSV OPERATION ON SAME SUPPORT VALUE.

Eid		**S1**	**S2**	**S3**	**Sup.**
AE	**A**	2	1	1	4
	E	1	1	2	4
	AE	**1**	**1**	**1**	**3**
DF	**D**	1	0	1	2
	F	1	0	1	2
	DF	**1**	**0**	**1**	**2**

V. THE CLOSPAM SSV ALGORITHM

In step1, scan the complete sequence database to get the distinct itemset (element) with its count value. It is being built into a base table. The step2 involves the operation is called ClospamSSV with same supported value of the itemsets. it repeats until the same supported threshold value for the output itemsets. step4, the closed minimal patterns are discovered from a large database.

A. Pseudo Code for ClospamSSV

ClospamSSV (SD, min_sup)
INPUT: SD – Sequence Database, min_sup – Minimum Support,
OUTPUT: Smallest number of Closed Sequential patterns
BEGIN:
/*scan the Sequence Database to Generate length -1 Candidates and to build base table*/
For each Sid <= SD begin
For each Si <=Sid begin
Distinct (Si) = Si
If Distinct (Si) = Si+1 then
Distinct (Si) Count++
Else
Distinct (Si) = Si
End If
End For
End For
// Closed Pattern
For each Si < = SN begin
If Si>=min_sup then
For each Sj < = SN begin //Sj = Si+1
If Disticnct(Si)Supp_Count=Distinct(Sj)Supp_Count then
Call OperationESV(Si,Sj)
Pattern(si)=Si
Else
ClosedPattern = Si
Supp_Count(Si) = Distinct(si)Supp_Count
End if
End for
Else
Pruned (Si) /* Because min_sup is less than user given minimum support value.*/
End if
End for
END
//Function OperationESV
Function OperationESV(Si, Sj, min_sup)
BEGIN:
If POS(Si) = 0 OR POS(Sj)=0 then
SiSj=0 // support value=0
End if
If POS(Si) = POS(Sj) then
SiSj=Si // Si or Sj the same support count value
End if
if POS(Si) != POS(Sj) then
SiSj=min(Si,Sj) //Least or minimum support value from both.
End if
Si=SiSj //with its support value
Return Si
END

VI. EXPERIMENTAL EVALUATION

In this ClospamSSV is implemented in Visual Basic programming on a personal computer of Intel Dual core 2.66 GHz processors, 1 GB RAM on Windows7 32bit Ultimate Operating System. The experimental evaluation performed on Real Data. The comparison experimental shows in table7. The Real Data has downloaded from the website: https://archive.ics.uci.edu/ml/datasets.html. The Experimental Analysis is done on the real world Online Retail and Gazelle KDDCup2000 datasets. The transformed Online Retail dataset contains 5, 41,909 transactions 2,603 unique items. The sample meaning of each item is given in table 6. And the Gazelle BMS Web view of KDDCup2000 data details shown in Table 7.

TABLE VII
SAMPLE ONLINE RETAIL DATA SETS

Item	ItemID
'Apples'	195
'Backpack'	222
'Balloons'	234
'Balls'	235

'Butter'	402
'Cheese'	516
'Cosmetics'	623
'Hairband'	1068
'Hairclip'	1070
'Handbag'	1080
'Wildflower'	2381
'Woodland'	2400

TABLE VIII
THE DETAILS OF BMS WEB VIEW DATA SETS.

SNo	Descriptions	Value
1	Total No. of Sequences	77512
2	Number of Distinct Items	3340
3	Average Length of Sequence	4.62
4	Standard Deviation of Items	6.07

The figure1 shows the performance analysis, the experimental evaluation concerning the running time is compared on real world datasets. These graphs show the results as the minimum support is changed from 0.01 to 0.05 percentage. Fig. 1, the experiments are carried out with varying min-sup values. The proposed algorithm ClospamSSV is showcased.

In Fig. 2 Shows when the min-sup value is lower, the ClospamSSV outperforms the previous Closed SPAM algorithms are Colspan and Clasp. Fig.2 shows as the average support value increase, the running time of ClospamSSV is decreased.

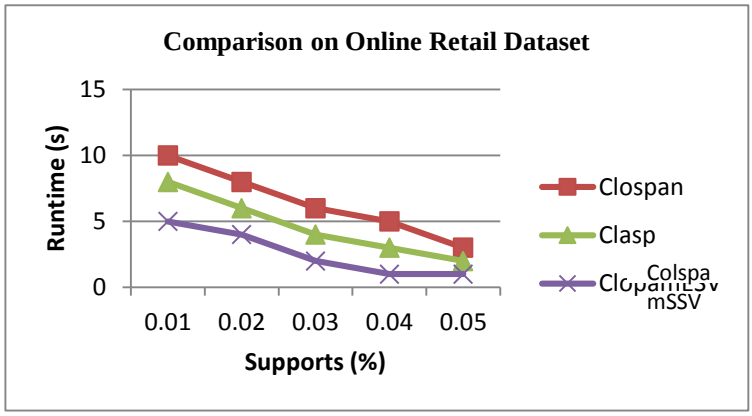

Fig1: Performance Analysis on Online Retail Datasets

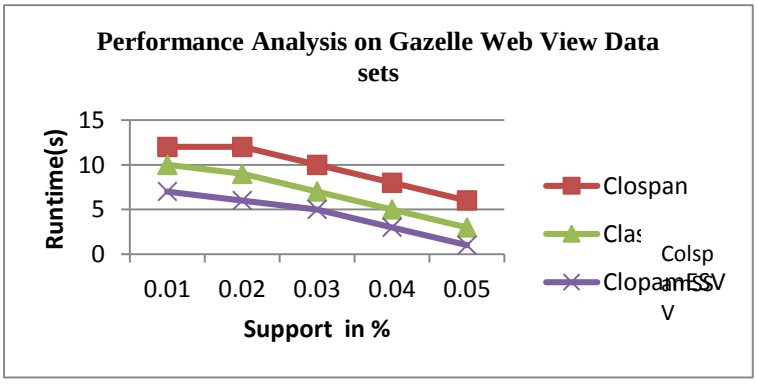

Fig2: Performance Analysis on Gazelle Web Click Stream Datasets

CONCLUSION

This algorithm ClospamSSV is most useful where minimum numbers of closed frequent sequential patterns need to be discovered. The primary challenge in closed sequential pattern mining depends on the size of the candidates generated, the size of the Database and squeezes the computations involved for the support count. The memory needed to store the data is much less or equal to the real database memory. The results table 8 shows that the proposed algorithm able to get the complete fewest number of Closed Sequential Pattern from the given sequence datasets with user-defined minimum support threshold.

TABLE VIII
OUTPUT FOR SAMPLE DATABASE TABLE1

Closed Frequent Patterns by ESV	**Support**
1	5
2	4
34	4
234	2

REFERENCES

[1] R. Agarwal and S.Arya, .Mining multiple level Association Rules to mining Multiple level Correlation to discover complex patterns. In Proc. 2012, International Journal of Computer Science, 2012.
[2] X. Yan, J. Han, and R. Afshar. CloSpan: Mining Closed Sequential Patterns in Large Datasets. SDM'03
[3] J. Wang and J. Han, BIDE: Efficient Mining of Frequent Closed Sequences, ICDE'04
[4] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.C. Hsu, "FreeSpan: Frequent pattern projected sequential pattern mining," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD'00), pp. 355-359, Aug. 2000.
[5] J . Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, "PrefixSpan : Mining Sequential patterns efficiently by prefix-projected pattern growth," Proc. Int'l Conf. Data Engineering (ICDE '01), pp. 215-224, Apr. 2001.
[6] R. Srikant and R. Agrawal, "Mining sequential patterns: Generalizations and performance improvements," Proc. Int'l Conf. Extending Database Technology (EDBT '96), pp. 3-17, Mar. 1996.
[7] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lot Lakhal, "Discovering frequent closed itemsets for association rules," Proceedings of the 7th International Conference on Database Theory (ICDT '99), pp. 398-416, 1999.
[8] K. Subramanian, E. Elakkiya, Modified Sequential Pattern Mining Using Direct Bit Position Method", International Journal of Science and Research (IJSR), ISSN (Online): 2319-7064, 2016.
[9] J. Wang, J. Han, and Chun Li, "Frequently closed sequence mining without candidate Maintenance," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 8, pp. 1042-1056, Aug. 2007.
[10] J. Pei, J. Han, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.C. Hsu, PrefixSpan: Mining Sequential patterns efficiently by prefix-projected pattern growth. In ICDE'01, Heidelberg, Germany, April 2001

