PSEUDO PROJECTION BASED APPROACH TO DISCOVERTIME INTERVAL SEQUENTIAL PATTERN

Dvijesh Bhatt

Department of Information Technology , Institute of Technology, Nirma University Gujarat,(India)

ABSTRACT

Data mining is the process to find out mysteriousand interesting patterns from transactional database. Sequential mining is the one of the major sub-area of data mining to find out the frequent sequences. As straight sequential pattern mining methods do not consider transaction occurrence time intervals, it is impossible to predict the time intervals of any two transactions mined as frequent sequences. There are several constraints to find out the effective and frequent sequential patterns. In this paper, I take time interval between two successive transactions. Time interval sequential mining is the process to find out the frequent sequential patterns with consideration of time interval constraint between two successive truncations. This paper proposed the modify version of the I-Prefixspan algorithm, is called as NI-PrefixSpan, to find out the time interval sequential pattern with pseudo projection table. Later, in this paper identify various advantage and drawback with this approach.

Keywords: Constraints, Data mining, Sequential mining, Time Interval, Time Stamp

I. INTRODUCTION

Data mining [1] is the process of extracting interesting, non-trivial, previously unknown and possibly useful information or patterns from large information sources and data warehouses. For the different types of applications, data and as per user needs, we need different techniques, so many sub-domains of mining was introduced as times goes on. Some of them are Sequential mining, Graph mining, Text mining and information retrieval, Web mining, spatial temporal mining etc.

Sequential mining is process to find out allfrequent sub-sequences from the given sequential dataset [2]. The outcome of this mining process are set of frequent transactions or events,which occurring in an order. Example: ‘A’ event is occurred first and after ‘B’ will occur. In this way sequence patterning is different compare to frequent patterns mining. In frequent patterns mining, order of events does not matter, while in sequential pattern mining order of the events are matter. Sequences always represent in angular brackets. i.e.<a,(ab),c>. Here in this sequence represents three transactions. In first transaction item ‘a’ had been purchased by the customer after that in second transactioncustomer bought both item ‘a’ and ‘b’ in single transaction which is represented by(ab) and in the last transaction customer came to buy item ‘c’. Here, <ab> is totally different sequence than<ba> because here the order of occurrence is mattered. Here, sequence <ab> indicates that item ‘a’ and ‘b’ bought by the customer in two different transactions and sequence <(a,b)> indicates that item ‘a’ and item ‘b’ bought by the customer in single transaction. In this area there are so many algorithms discovered to find out sequence patterns. Algorithms are based on Apriori approach or pattern growth approach. Some of the wildly used Apriori based algorithms are Apriori, GSP [3], SPADE [4], SPAM [5], LAPIN [6], etc. Some of the wildly used pattern growth algorithms are FreeSpan [7] and PrefixSpan [8]. Out of them PrefixSpan is one of

the decent algorithm due to faster calculation and new constraint adopting nature.All these methods do not consider the time interval between any two transactions mined as frequent patterns. In general terms, based on the output patterns, no buddy can judge that how much time will be taken by the customer to do two successive transactions while held 'a' and '(ab)'. To find these kind of frequent sequential pattern, Dr. Chen have introduced the Time Interval Sequential Mining [9] algorithms using both approaches i.e Apriori and pattern growth.. Dr. Chen and his teamshave introduced two algorithms to find out the time interval Sequential mining patterns and those are called as I-Apriori and I-PrefixSpan. So output of this algorithm will be like,having bought a car, a customer will come back to buy a seat cover with in three months and then come back to buy car mobile charger within six months from purchased of seat cover. Here we can see the particular time interval is given between two items, which has been purchased in particular one order. These algorithms are facing sharp boundary problem. To overcome the sharp boundary problem updated version of I-Apriori and I-PrefixSpan had introduced and those algorithms called as FTI-Apriori and FTI-PrefixSpan [10]. In that paper, fuzzy logic was used to overcome sharped boundary problem. Second problem with the I-Apriori and I-PrefixSpan is, these algorithms only show the time interval between two successive events i.e those algorithm never show the how much time will be taken by customer to buy car and car mobile charger. To overcome this restriction, one more version of I-Apriori and I-PrefixSpan was introduced which are called as MI-Apriori and MI-PrefixSpan [11]. Also MLTI-PrefixSpan [12] algorithm was introduced to find out the cross level time interval sequential patterns. Also some on the algorithm also discover integrated sequential patterns mining with fuzzy time interval [13].

II. ANALYSIS OF CURRENT ALGORITHMS

In sequential mining, basically two approach are used to discover frequent sequential patterns i.e Apriori and pattern growth. Both the approaches have some of the advantage and disadvantage with respect to time to generate all sub-sequences, memory is used by algorithm, no of time database scan etc.

First of all let's take a look of advantages and disadvantages of Apriori based approach

Advantage

1. It is really easy algorithm to understand.
2. It is very simple algorithm and find out all frequent sub-sequences from the given sequential data repository.

Disadvantage

1. You have to scan the dataset multiple times to find out frequent sub-sequences, due to this, algorithms takes lots of time to generate the frequent patterns.

   Now let's take a look of advantage and disadvantage of second approach which is pattern growth algorithm which is basically I-PrefixSpan

Advantage

1. It will not scan the main dataset multiple times. Instead of that, algorithm creates projected table for each and every sub-sequences and have to scan projected tables to generate the new frequent sub-sequences.
2. It take comparatively less time to generate all frequent sub-sequences.

   Disadvantage

1. Main drawback of this kind of approach is that you have to generate the projection table for every frequent sub-sequence. For each sub-sequences new projection table are created, due to this you need large amount of main memory to store all the projection tables in it.

   There are many versions of algorithm based on this two approaches to find out the time interval sequential mining. Here in table-1 I list out all them.

III. SOLUTIONS OF THE CURRENT SYSTEM

Here in this section, some of suggestions has been mentioned to overcome the problems of current time interval sequential mining algorithms.

1. So far as per our study, no algorithm has been implemented the pseudo projection table [14] and bi-matrix table which deal with the time interval constraints.
2. PrefixSpan is really good algorithm but still have lack even if we use the pseudo projection. Because even pseudoprojection I-PrefixSpan is stored projected tables in main memory which cost the memory usages and time as well. Also if we have extra-large database then all projection tables are not fit in main memory. To solve problem of PrefixSpan's projection tables one algorithm was proposed to find out frequent sequential patterns and it is called as MEMISP [15], which works as same as the PrefixSpan with pseudoprojection but deal with large database by partition and combine technique.
3. Third solution which is describe, as future enhancement of time interval sequential mining is that use some of the taxonomy or constraints which can eliminate the some of the candidate sequences, which are not frequent, at the beginning of the stage. So it will not waste the time and memory. You can use some of constrains like used in the GSP to make algorithms faster and find only the interesting patterns [16].

Table-1 : List of algorithms for time interval sequential mining

Time Interval Sequential Mining Algorithms			
Sr. No	Name of Algorithms	Year	Descriptions
1	I-Apriori	2003	First algorithm which introduced time interval sequential patterns. It is use typical Apriori base algorithm for that.
2	I-PrefixSpan	2003	First pattern growth algorithm to find the time interval sequential pattern. It is extend version of PrefixSpan.
3	FTI-Apriori	2005	In this algorithm, the boundary of time intervals are not fixed so sharp boundary problem has been resolved. For that fuzzy logic is used. This is Apriori based algorithm.
4	FTI-PrefixSpan	2005	In this algorithm, the boundary of time intervals are not fixed so sharp boundary problem has been resolved. For that fuzzy logic is used. This is pattern growth based algorithm.
5	MI-Apriori	2009	With this algorithm time interval between two non-successive tractions has been calculated. This is Apriori based algorithm.
6	MI-PrefixSpan	2009	With this algorithm time interval between two non-successive tractions has been calculated. This is pattern growth based algorithm.

7	FuzzMI-Apriori	2010	With this algorithm time interval between two non-successive tractions has been calculated. Also the boundary for time interval is not fixed and for that fuzzy logic has been used. This is Apriori based algorithm.
8	FuzzMI-PrefixSpan	2010	With this algorithm time interval between two non-successive tractions has been calculated. Also the boundary for time interval is not fixed and for that fuzzy logic has been used. This is pattern growth based algorithm.
9	MLTI-PrefixSpan	2010	With the help of this algorithm, time interval sequential patterns of cross level can be discovered. For this pattern growth base algorithm PrefixSpan was used.

IV PROPOSED SOLUTION

Now in this paper,we proposed new pseudo projection table approach to generate time interval sequential patterns. As PrefixSpan uses the pseudoprojection to generate the sequential patterns but in this proposed solution the structure of the pseudoprojection table is changes as it uses now generate the time interval sequential patterns. Previously proposed I-PrefixSpan algorithm did not use the pseudo projection table approach so it required much amount of memory to store the data in to main memory. Pseudoprojection table is stored the indexes of the occurrence of the items in the dataset [3]. Pseudo projection table stored (SID, offset). SID indicates specific sequences ID of sub sequences and offset indicates at which place sub-sequence is been in the sequential data repository. In the proposed solution, changes are made the structure of the pseudoprojection table so that we can use the structure to generate the time interval sequential patterns. In proposed structure of the pseudo projection table is contained offset and associate time unit for the specific item in the sequences.In the Table-2is having dataset. In this dataset, every item have time stamp along with it. In the first step, scan the dataset to find out all frequent items.

TABLE-2 DATASET OF TIME INTERVAL SEQUETIAL MINING

Sequence ID	Sequence
10	*<(a,1),(c,3),(a,4),(b,4),(a,6),(e,6),(c,10)>*
20	*<(d,5),(a,7),(b,7),(e,7),(d,9),(e,9),(c,14),(d,14)>*
30	*<(a,8),(b,8),(e,11),(d,13),(b,16),(c,16),(e,20)>*
40	*<(b,15),(f,17),(e,18),(b,22),(c,22)>*

Now consider minimum support is 50%so based on thatitems (a), (b), (c), (d) and (e) are found as frequent items. Item (f) has been eliminated as it do not have the minimum support thresh hold value. Now generate the projection table for all frequent items. Instead of, storing all projection tables in main memory, make the pseudoprojection table. In Table-3 represents modify pseudo projection table. First cell contains 1, 4, 6 that means item ‘a’ is present in sequence ID 10 with 1, 4 and 6 time stamps. Modified pseudo projection table stores the time stamp of the items instead of index of the item. If both items are having same time stamp so it indicates that both items are belongs to same transaction.To generate Time interval sequential pattern, time interval has to be define. Here fix time interval are

$I_0 = 0$; $I_1 = 0 < t \leq 3$ $I_2 = 3 < t \leq 6$ $I_3 = 6 < t \leq$

Number of time intervals and their range are fixed.In this example, three time intervals are there as I_0, I_1, I_2, and $I_3$and their ranges are 0, (0,3] , (3,6] and (6, respecetively. If time interval gap between two successive events is 4 so both items associate with each other with time interval I_2.

TABLE -3 PSEUDOPROJECTION TABLE

SID	<a>	<b>	<c>	<d>	<e>
10	1, 4, 6	4	3,10	Ø	6
20	7	7	14	5,9,14	7,9
30	8	8,16	16	13	11,20
40	Ø	15,22	22	Ø	18

Here [*Sid* : *Pos*] structure has been used**,** where *Sid* is indicating sequence ID and *Pos* is indicating position of item. Based on this, five postfix sequences in the projected database is generated. In proposed solution, this kind of projection table would not store in main memory. This is just for the easy understanding.

[10:1] ((*c*, 3)(*a*, 4)(*b*, 4)(*a*, 6)(*e*, 6)(*c*, 10)),

[10:4] ((*b*, 4)(*a*, 6)(*e*, 6)(*c*, 10)),

[10:6] ((*e*, 6)(*c*, 10)),

[20:7] ((*b*, 7)(*e*, 7) (*d*, 9)(*e*, 9)(*c*, 14)(*d*, 14)),

[30:8] ((*b*, 8)(*e*, 11)(*d*, 13)(*b*, 16)(*c*, 16), (*c*, 20)).

Now generation of the frequent sequential pattern, time interval table has been generated, which looks like table-4.In the table, 1st column indicated the time interval and 1st raw will indicate the frequent items. Remaining cells indicate the frequency of data occurrence with particular time interval. Using this data, 2-length sub-sequences is generated. Time interval gap can be calculated by the simple subtraction of the time stamp associate with respective items. If time interval between both items is zero, so it indicates that both the items have been purchased in the single transaction. From table-3, we can calculate time interval between two itemswhich is useful to generate table-4. Here in sequence ID 10, item ‘a’ is occurred with time stamp 1, 4 and 6 and ‘b’ occurred at time 4 as well. Now considering time stamp of item ‘a’ as 1 and time stamp of item‘b’ as 4, so time interval between them is: | 1– 4 | = 3 which belongs to I_1. So, now just increase the counter of cell B-I_1 by one. Based on this time interval table is created. Observing from the table-4, cellB-I_0 is frequent because it is stratifying minimum support criteria. Now from table-4, length-2 time interval sequential patter has been generated. Thus, (a, I_0, b) is the frequent sequence. That means customer bought item ‘a’ and item ‘b’ in single transactionmore frequently.

TABLE-4The table constructed in NI-PrefixSpan((*a*), 1, *S* | (*a*))

Table	A	B	C	D	E
I_0	0	3	0	0	2
I_1	1	1	1	1	3
I_2	1	0	1	1	1
I_3	0	1	3	1	0

So here (a, I_0, b), (a, I_0, e), (a, I_1, e) and (a, I_3, c) are the frequent sub-sequences. Now we also have to make the index of these frequent items. Suppose we take (a, I_0, b) as one sequence so we have to make the index for it.

And it will be 4, 7, 8 respectively for the sequence ID 10, 20 and 30. Now repeat this step until discovering of all frequent sub-sequence. Suppose now we take (a, I_0, b) and make projection table of this. So we will get

[10:4]: ((*a*, 6)(*e*, 6)(*c*, 10))

[20:7]: ((*e*, 7)(*d*, 9)(*e*, 9)(*c*, 14)(*d*, 14))

[30:8]: ((*e*, 11)(*d*, 13)(*b*, 16)(*c*, 16)(*c*, 20))

Base on that we make another table same like Table-4.

Table-5 NI-PrefixSpan(((a, I_0, b), 4, S | (a, I_0, b)))

Table	B	C	E
I_0	0	0	1
I_1	0	0	3
I_2	0	1	0
I_3	1	2	0

Now, from the Table-5 that cell (I_1, e) and (I_3, c) is satisfying minimum support thresh hold. Base on that find 3-length sub-sequences are (a, I_0, b, I_1, e) and (a, I_0, b, I_3, c).And again have to make the pseudoprojection table or index for the new sub-sequences.

Advantage of pseudoprojection table is that projection table is not stored in main memory. Thus, it is utilizationof main memory. Also, do not need to scan main database multiple times and there is no need to generate the projection table for each and every sub-sequences. But one disadvantage of this method is stored whole dataset into main memory. One possible solution for the this problem is, store some patterns in pseudo projection table and some patterns in physical projection. Combination of both projection techniques can solve the current shortcoming of NI-PrefixSpan. When data shrink enough to store in main memory all dataset will be move to the main memory otherwise both memory can be used to store the dataset.

IV. CONCLUSION

In this paper, Iproposed new algorithm NI-PreifixSpan to discover time interval sequential patternswith the help of modify pseudo projection table which stores the time stamp value of different items instead of its index in sequence ID. Also algorithm solved two major problem which are related to memory and no of time main dataset will be scannedwill be reduced. In future extension of the NI-PrefixSpan can be in two way, one is to improve the performance of the current algorithm and second apply same algorithm to find multiple time interval sequential patterns.

REFERENCES

[1] J. Han and M.Kamber, “Data Mining – Concepts & Techniques”, Morgan Kaufmann Publishers (Academic Press), 2001.

[2] R. Agrawal and R. Srikant, “Mining sequential patterns”, International Conference of Data Engineering (ICDE
’95), 1995.

[3] R. Srikant and R. Agrawal, “Mining sequential patterns: Generalizations and performance improvements”, Proceedings of the 5th International Conference Extending Database Technology, 1057, pp. 3-17, 1996.

[4] M. Zaki, "SPADE: An efficient algorithm for mining frequent sequences", Machine Learning, vol.40 pp. 31-60 2001.

[5] J.Ayres, J. Flannick, J. Gehrke, and T. Yiu, "Sequential pattern mining using a bitmap representation", Proc.OfInternational Conference on Knowledge Discovery and Data Mining, 2002.

[6] Zhenglu Yang; Kitsuregawa, M., "LAPIN-SPAM: An Improved Algorithm for Mining Sequential Pattern," *Data Engineering Workshops, 2005. 21st International Conference on* , vol., no., pp.1222,1222, 05-08 April 2005

[7] J. Pei, J. Han, B. Mortazavi-Asi and H. Pino, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth", International Conference of Data Engineering(ICDE'01), 2001.

[8] J. Han, G. Dong, B. Mortazavi-Asl, Q. Chen, U. Dayal and M.-C. Hsu,"FreeSpan: Frequent pattern-projected
sequential pattern mining", Proc. 2000 International Conference of Knowledge Discovery and Data Mining (KDD'00), pp. 355-359, 2000.

[9] Chen, Y.L., Chiang, M.C. and Ko, M.T. (2003). "Discovering time- interval sequential patterns in sequence databases," Expert Syst. Appl., Vol. 25, No. 3, (pp. 343-354).

[10] Chung-I Chang; Hao-En Chueh; Lin, N.P., "Sequential Patterns Mining with Fuzzy Time-Intervals," *Fuzzy Systems and Knowledge Discovery, 2009. FSKD '09. Sixth International Conference on* , vol.3, no., pp.165,169, 14-16 Aug. 2009

[11] Ya-Han Hu, Tony C. Huang, Hui-Ru Yang, Yen-Liang Chen, "On mining multi-time-interval sequential patterns", Data & Knowledge Engineering, Vol. 68, No. 10. pp. 1112-1127,(23 Oct 2009).

[12] Ya-Han Hu; Fan Wu; Chieh-I Yang; , "Mining multi-level time-interval sequential patterns in sequence databases," Software Engineering and Data Mining (SEDM), 2010 2nd International Conference on , vol., no., pp.416-421, 23-25 June 2010.

[13] Chung-I Chang; Hao-En Chueh; Yu-Chun Luo, "An integrated sequential patterns mining with fuzzy time-intervals," *Systems and Informatics (ICSAI), 2012 International Conference on* , vol., no., pp.2294,2298, 19-20 May 2012.

[14] Dvijesh Bhatt and Kiran amin;, "A-Survey Time Interval Sequential Mining", International Conference on recent trend in computer science and engineering (ICRTCSE 2012), May-2012.

[15] M.Y. Lin and S.Y. Lee, "Fast Discovery of Sequential Patterns through Memory Indexing and Database Partitioning," *J. Information Science and Eng.*, vol. 21, pp. 109-128, 2005.

[16] J. Pei, J. Han and W. Wang, "Constraint-based sequential pattern mining: the pattern growth methods", J Intell.Inf. Syst, Vol. 28, No.2, pp. 133 –160, 2007.