International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 12, December-2015

Closed Sequential Pattern Mining using Length Constraint

Manika Verma[1],
[1]Assistant Professor
[1]Department of Computer Science Engineering
[1]Kadi Sarva Vishwavidyalaya,
Gandhinagar, India,

Dr. Devarshi Mehta[2]
[2]Associate Professor
[2]GLS Institute of Computer Technology,
Ahmedabad, India

***Abstract* - Sequential Pattern Mining is an approach to find sequences that occurs frequently in a dataset. These sequences are later used for predicting occurrence of next event/item in sequences. Sequential Pattern Mining is widely used in Areas like Healthcare, Education, Web Usage Mining, text mining, bioinformatics and telecommunications. Traditional approaches mines frequent closed sequences (sequence that do not have any superset sequence with same support). BIDE is an efficient algorithm for mining closed sequences. These approaches generate large number of sequences many of them are useless. This paper proposed an approach to efficiently mine sequences by incorporating constraint length in algorithms, while mining sequences. Incorporating length constraints reduces number of patterns generated and thus produces less number of sequences. Length constraint is incorporated in an algorithm BIDE which mines frequent closed sequences from multidimensional dataset. In other algorithm, which mines sequences considering time, from dataset, length constraint is added. This also reduced the number of sequences generated.**

1. INTRODUCTION:

Sequential pattern mining is widely used in various areas like Healthcare, Education, Web Usage Mining, text mining, bioinformatics and telecommunications. Rakesh Agrawal et.al introduced Sequential pattern mining in 1995[1]. Since then various work had been done to make improvement in Sequential Pattern Mining. Later method for fast discovery of sequences using vertical database format was proposed [3]. In traditional sequential pattern mining large number of candidate sets were generated, so the concept of Projected Database was introduced [4]. These traditional approaches of mining sequences use to generate large number of redundant frequent sequences. To overcome this problem the concept of Closed Sequential Pattern Mining was introduced by X.Yan et.al[2]. To improve efficiency and to reduce execution time while mining closed sequences the method of Bidirectional Execution was proposed [5]. Though closed sequential pattern mining didn't generated redundant patterns but use to generate large number of non-redundant patterns among which few are of no use. Mining sequences considering constraint reduced the number of non-redundant sequences generated. Jian Pei et.al [6] proposed an algorithm for mining sequences considering regular expression constraint, length constraint and duration constraint. This algorithm used Pattern growth method (projected database) which generated less number of sequences as compared to those algorithms which generates sequences without considering constraints. Sequences Mining algorithm considering constraints like weight, gap were also introduced [7][8][9].

In this paper we propose an algorithm that consider length constraints and generates only those sequences which satisfies particular length. We adopt[5] BIDE algorithm for mining frequent closed sequences. This paper consists of two algorithms.

1. An algorithm that mines closed frequent sequences considering length constraint from multidimensional dataset
2. An algorithm that mines closed frequent sequences considering length constraint from **two-dimensional** dataset which contains time-wise sequences (sequences generated in particular time).

2. PROBLEM DEFINITION

The problem of mining frequent sequences considering length constraint can be stated as follow. Let I={I1,I2,….,In} be set of n distinct items comprising numbers. An event is a non-empty collection of items. A **sequence** is an ordered list of events. An event is denoted as (I1I2…Ik), where Ij is an item. A sequence is denoted as {S1->S2->S3…. ->Sn} where Si is event.
A sequence containing events {S1, S2, S3, S4} is said to be super-set of sequence containing events {S1, S2} and a sequence containing events {S1, S2} is said to be subset of sequence containing events {S1, S2, S3, S4}.

An input sequence Database is a set of tuples containing sequences. The **absolute support** of a sequence S1 in a Database is number of tuples in Database that contains sequence S1.

Given a support threshold Min_Sup, a Sequence S1 is a frequent sequence in Database if Support of S1>= Min_Sup. If Sequence S1 is frequent and there exists no sub-sequence of sequence S1 with same minimum support then S1 is called **frequent closed sequence**. The problem of mining frequent closed sequences is to find complete set of frequent closed sequences for an input sequence database, given a minimum support threshold, min_sup.

Table 1	
1	1 1 1 -3 2 4 -1 3 -1 2 -1 1 -1 -2
2	1 2 2 -3 2 6 -1 3 5 -1 6 7 -1 -2
3	1 2 1 -3 1 8 -1 1 -1 2 -1 6 -1 -2
4	* 3 3 -3 2 5 -1 3 5 -1 -2
	Multidimensional Database Sequence

Table 2	
1	<0> 1 -1 <1> 1 2 3 -1 <2> 1 3 -1 -2
2	<0> 1 -1 <1> 1 2 -1 <2> 1 2 3 -1 <3> 1 2 3 -1 -2
3	<0> 1 2 -1 <1> 1 2 -1 -2
4	<0> 2 -1 <1> 1 2 3 -1 -2
	Time-Extended Database Sequence

Example 1: Table 1 shows the input sequence database in running example. The database has totally 8 unique items, four input sequences. It has multiple dimensions specified before -3. Suppose min_sup= 0.20 (50%). A frequent closed sequence generated without considering length constraint is [1 * 1] {t=0, 1} #Sup: 2. When length constraint (In running example, length>14) is incorporated in algorithm the above mentioned sequence is not generated and thus total number of sequences generated are less.

Example 2: Table-2 shows Time-Extended Sequences. The database has 3 unique items and 4 input sequences. It has time specified as <0> <1>. Suppose min_sup= 0.55. A frequent closed sequence generated without considering length constraint is <0> 2 -1 <1> 1 -1 #SUP: 4 . When length constraint (In running example, length>2) is incorporated in algorithm the above mentioned sequence is not generated and thus total number of sequences generated are less.

3. RELATED WORK

The Sequential Pattern Mining was first proposed by Agrawal and Srikant[1]. Since then many algorithms have been proposed for improvement. Various sequential pattern mining algorithm are GSP[10],SPADE[3],Prefix-Span[4],CloSpan[2],BIDE[5], Mining Sequential Patterns with Constraints in Large Databases[6]. GSP used Apriori property for mining frequent sequences, but the disadvantage with this approach is that lots of candidate sets are generated. SPADE used vertical database for mining frequent sequences and outperforms GSP. Prefix-Span mines frequent sequences using projected database and outperforms GSP. GSP, Prefix-Span and SPADE do not generate frequent closed sequences (sequences that do not have any super-set with same support). CloSpan was introduced for mining frequent closed sequences. Later to improve performance for mining Frequent Closed Sequences BIDE was introduced.

Various work related to Constraint had been done. An algorithm “Mining Sequential Patterns with Constraints in Large Databases” for mining frequent sequences considering various constraints like item-constraint, length-constraint, super-pattern constraint, aggregate constraint, regular expression constraint was proposed [6]. This algorithm uses prefix-growth method. Prefix-growth method for mining sequences generates redundant sequences. To overcome this problem, super-pattern constraint is pushed in this algorithm. Pushing Weight constraint along with sequences generates more important patterns. Unil Yun proposed an algorithm for mining closed frequent sequences with weight constraint [7].

An algorithm for extracting coding patterns was proposed by Hiromasa et.al [11]. The new constraint named as “intensity constraint” was pushed in this algorithm.

Vangipuram et.al[12] proposed a pattern mining algorithm which may be embedded in SQL or MySQL, so that one can search presence of sequential pattern in time-series database. User defined constraints were pushed in this approach for generating restricted number of sequences.

An algorithm for mining frequent sequences with non-user defined Gap constraint was proposed [13]. This approach is appropriate, if user has no knowledge regarding pre-defined gap

But these constraints are incorporated in CloSpan or PrefixSpan algorithm. These algorithms are less efficient as compared to BIDE algorithms.

Contribution: In this paper, we used BIDE algorithm and incorporated length constraint.

For mining sequences from multi-dimensional dataset, the algorithm using concept BIDE, named as MainTestMultiDimSequentialPatternMiningClosed in SPMF (An-Open Source data mining library) [14] is used. Length constraint is incorporated in this algorithm

For mining sequences from time-extended dataset, the algorithm using concept of BIDE, named as MainTestSequentialPatternMining2_saveToFile in SPMF (An-Open Source data mining library) [14] is used. Length constraint is incorporated in this algorithm.

1. A BIDE[5][14] algorithm for mining sequences from multi-dimensional dataset is used and length constraint is incorporated in it.
2. A BIDE[5][14] algorithm for mining sequences from time-extended dataset is used and length constraint is incorporated in it.

4. ALGORITHM: MINING FREQUENT CLOSED SEQUENCES CONSIDERING LENGTH CONSTRAINT

In proposed approach, length constraint is incorporated in BIDE algorithm. BIDE, using various technique itself mines frequent closed sequences efficiently. Incorporating length constraint in BIDE algorithm enables it to produce only those sequences which satisfy length constraint.

Bide algorithm first scans database once to find frequent-1 sequences, and then projected database is for each frequent 1- sequences is built. Each frequent 1-sequences is treated as a prefix and BackScan pruning method is used to check if it can be pruned. If not, then Bide computes the number of backward-extension-item, and calls subroutine bide (Sp SDB, Sp, min sup, BEI, FCS). Subroutine bide(Sp SDB, Sp, min sup, BEI, FCS) recursively calls itself and works as follows:

1. For prefix Sp, scan its projected database Sp SDB once to find its locally frequent items,
2. Compute number of forward-extension items, if there is no backward-extension-item nor forward-extension-item, output Sp as a frequent closed sequence is generated,
3. Grow Sp with each locally frequent item in lexicographical ordering to get a new prefix and build the pseudo projected database for the new prefix, for each new prefix, first check if it can be.[5]

Step 1: Input a Sequence Database, a minimum support threshold min_sup
Step 2: Frequent Closed Sequences =NULL
Step 3: F1: Frequent 1 sequences generated
Step 4: For each Frequent 1 sequence F1, Pseudo Projected database is created.
Step 5: For each F1
Check using BackScan Pruning if it can be pruned.
Step 6: If not then BIDE computes backward-extension-item and call subroutine BIDE. Subroutine bide recursively calls itself and works as follows:

Subroutine BIDE
Step 7: For prefix Sp, scan its projected database to find locally frequent items.
Step 8: If there is neither backward-extension-item nor forward-extension-item, output Sp as a frequent closed sequences are obtained.
Step 9: Grow Sp with each locally frequent item in lexicographical ordering to get a new prefix and build the pseudo projected database for the new prefix,
for each new prefix, first check if it can be pruned, if not, compute the number of backward extension-items and call itself .
Step 10: Finally frequent closed sequences are obtained and Length of these frequent sequences are checked. If these sequences satisfy length then finally these sequences are generated and stored as final sequences that had satisfied length constraint. //Addition made in existing BIDE algorithm

5. EXPERIMENTAL RESULTS

The BIDE algorithm Main Test Multi Dim Sequential Pattern Mining Closed) Implemented by SPMF (An-Open Source data mining library) [14] applied on multidimensional dataset given in Table-1 results into sequences generated as shown in Fig.1. 11 sequences are generated.

In the algorithm Main Test Multi Dim Sequential Pattern Mining Closed, length constraint is added. The proposed algorithm considers length constraint while mining frequent sequences when applied on multidimensional dataset given in Table-1 generated 8 sequences shown in Fig.2. Thus the numbers of sequences generated are reduced by incorporating length constraint in algorithm.

Frequent sequences generated without considering length constraint from multidimensional dataset
[1 2 1]{t=0, 1 8 }{t=0, 1 }{t=0, 2 }{t=0, 6 } #SUP: 1
[1 1 1]{t=0, 2 4 }{t=0, 3 }{t=0, 2 }{t=0, 1 } #SUP: 1
[1 2 2]{t=0, 2 6 }{t=0, 3 5 }{t=0, 6 7 } #SUP: 1
[* * *]{t=0, 2 }{t=0, 3 5 } #SUP: 2
[* * *]{t=0, 2 }{t=0, 3 } #SUP: 3
[1 * *]{t=0, 2 }{t=0, 3 } #SUP: 2
[* 3 3]{t=0, 2 5 }{t=0, 3 5 } #SUP: 1
[1 2 *]{t=0, 2 }{t=0, 6 } #SUP: 2
[1 * 1]{t=0, 1 } #SUP: 2
[* * *]{t=0, 2 } #SUP: 4
[1 * *]{t=0, 2 } #SUP: 3
Fig 1

In the running example, the frequent closed sequences generated after incorporating length constraints in implemented BIDE algorithm [14] are as specified below. (Here, sequences having length >14 are only generated)

Frequent sequences generated without considering length constraint from time-extended dataset
[1 2 1]{t=0, 1 8 }{t=0, 1 }{t=0, 2 }{t=0, 6 } #SUP: 1
[1 1 1]{t=0, 2 4 }{t=0, 3 }{t=0, 2 }{t=0, 1 } #SUP: 1
[1 2 2]{t=0, 2 6 }{t=0, 3 5 }{t=0, 6 7 } #SUP: 1
[* * *]{t=0, 2 }{t=0, 3 5 } #SUP: 2
[* * *]{t=0, 2 }{t=0, 3 } #SUP: 3
[1 * *]{t=0, 2 }{t=0, 3 } #SUP: 2
[* 3 3]{t=0, 2 5 }{t=0, 3 5 } #SUP: 1
[1 2 *]{t=0, 2 }{t=0, 6 } #SUP: 2
Fig 2

Example 2: Table-2 shows Time-Extended Sequences. The database has 3 unique items and 4 input sequences. It has time specified as <0> <1>. The algorithm (MainTestSequentialPatternMining2_saveToFile) available at SPMF (An-Open Source data mining library) [14] mines time-extended dataset and the complete set of frequent closed sequences without considering length constraints are

<0> 1 -1 <1> 1 2 -1 #SUP: 3
<0> 1 2 -1 <1> 1 -1 #SUP: 3
<0> 1 2 3 -1 #SUP: 3
<0> 1 2 -1 #SUP: 4
<0> 2 -1 <1> 1 3 -1 #SUP: 3
<0> 2 -1 <1> 1 2 -1 #SUP: 3
<0> 2 -1 <1> 1 -1 #SUP: 4
Fig 3

In the algorithm Main Test Sequential Pattern Mining2_saveToFile length constraint is added. In the running example, the frequent closed sequences generated after incorporating length constraints are (Here, sequences having length >2 are only generated)

Frequent sequences generated considering length constraint from time-extended dataset
<0> <1> 1 2 -1 #SUP: 3
<0> 1 2 -1 <1> #SUP: 3
<0> 1 2 3 -1 #SUP: 3
<0> 1 2 -1 #SUP: 4
<0> <1> 1 3 -1 #SUP: 3
<0> <1> 1 2 -1 #SUP: 3
Fig 4

6. CONCLUSION

In this paper, we proposed a method for efficiently mining closed frequent sequences considering length constraint. We have made changes in existing BIDE algorithm. 1. In a BIDE algorithm, that mines frequent sequences from multidimensional dataset, a length constraint is incorporated. BIDE algorithm generates more number of sequences as compared to BIDE algorithm with length constraint.

2. In a BIDE algorithm, that mines frequent sequences from time-extended dataset, a length constraint is incorporated. BIDE algorithm generates more number of sequences as compared to BIDE algorithm with length constraint.

By incorporating length constraint in Closed Sequential Pattern Mining Algorithm, the large number of patterns generated can be reduced. In proposed work, the number of patterns generated by algorithm after incorporating length constraints is less than as compared to sequences generated by algorithm without considering length constraint.

7. FUTURE WORK:

Various work related to Constraint had been done. But these constraints are incorporated in CloSpan or PrefixSpan algorithm. These algorithms are less efficient as compared to BIDE algorithms. Other constraints like gap, regular expression constraints can be incorporated in BIDE algorithm to reduce total number of patterns generated by algorithm. These constraints restrict generation of useless patterns and reduce total number of patterns generated.

REFERENCES

[1] Agrawal,R.&Srikant.R. “Mining Sequential Patterns”. In 11thIntl.Conf on Data Engineering
[2] X. Yan, J. Han, and R. Afshar. Clospan: Mining closed sequential patterns in large datasets. In SDM, pages 166–177, 2003.
[3] Mohammed J.Zaki. “SPADE:An Efficient Algorithm for Mining Frequent Sequences” Machine Learning Volume 42, Issue 1-2, January 2001
[4] J. Pei, J. Han, B. Mortazavi-asl, H. Pinto, Q. Chen, U. Dayal, and M. chun Hsu. Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth. In Data Engineering, 2001. Proceedings. 17th International Conference on, pages 215–224, 2001.
[5] J. Wang and J. Han. “Bide: efficient mining of frequent closed sequences”. In Data Engineering, 2004. Proceedings. 20th International Conference on, pages 79–90, 2004.
[6] Jian Pei, Jiawei Han,Wei Wang “Mining Sequential Patterns with Constraints in Large Databases”. : CIKM’02, November 4–9, 2002, McLean, Virginia, USA,2002
[7] Unil Yun,” Mining Lossless Closed Frequent Patterns With Weight Constraints”, Elsevier. Knowledge based system (2007)
[8] Hiromasa TAKEI, Hayato YAMANA, “IC-BIDE: Intensity Constraint-based Closed Sequential Pattern Mining for Coding Pattern Extraction”, IEEE 27th International Conference on Advanced Information Networking and Applications,2013
[9] V.Purushothama Raju, Dr.G.P Sarachi Varma, “A Survey on Closed Sequential Pattern Mining”, International Conference on Information Communication and Embedded Systems (ICICES) , 2014, Page(s): 1 – 6
[10] R. Srikant, and R. Agrawal, Mining sequential patterns: Generalizations and performance improvements. In EDBT’96, Avignon, France, Mar. 1996
[11] Hiromasa TAKEI, Hayato YAMANA, “IC-BIDE:Intensity Constraint-based Closed Sequential Pattern Mining for Coding Pattern Extraction”, IEEE , 2013
[12] Vangipuram Radhakrishna, Chitakindi Srinival, Dr.C.V.Guru Rao, “Constraint Based Sequential Pattern Mining in Time Series Databases- A two way Approach” , 2013 AASRI Conference on Intelligent Systems and Control, 2013
[13] Wentao Wang, Lei Duan1,Jyrki Nummenmaa, Song Deng, Zhongqi Li, Hao Yang, and Changjie Tang, “Mining Frequent Closed Sequential Patterns with Non-user-defined Gap Constraints”, ADMA 2014 Springer International Publishing Switzerland, 2014
[14] An Open Source Data Mining Library, http://www.philippe-fournier-viger.com/spmf/