Volume 3, Issue 12, December 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Survey of Sequential Pattern Mining Algorithms and an Extension to Time Interval Based Mining Algorithm

V. Uma[1], M. Kalaivany[2]
Department of Computer Science
Pondicherry University, India

G. Aghila[3]
Department of CSE,
NIT Puducherry, Karaikal, India

Abstract*—Sequential pattern mining finds the subsequence and frequent relevant patterns from the given sequences. Sequential pattern mining is used in various domains such as medical treatments, natural disasters, customer shopping sequences, DNA sequences and gene structures. Various sequential pattern mining algorithms such as GSP, SPADE, SPAM and PrefixSpan have been proposed for finding the relevant frequent patterns from sequences. Survey of all these algorithms is provided in this paper. PrefixSpan, TPrefixSpan algorithms discover patterns from point and interval based event data respectively.This paper proposes a constraint based TPrefixSpan algorithm that generates only interesting patterns which eventually reduces the computational cost and enhances the performance. The various constraints like item, length, and aggregate constraints can be introduced in TPrefixSpan algorithm and this forms the basis of our proposed work.

Keywords*—Sequential pattern mining, PrefixSpan, Temporal patterns, Constraints, Interval based events

I. INTRODUCTION

Data mining [1, 10] is useful in various domains such as market analysis, decision support, fraud detection, business management and so on. Many approaches have been proposed to extract information from input sequencesand sequential pattern mining is one of the most importantmethods. Various methods have been proposed for mining temporalpatterns in sequence databases such as mining repetitive patterns,trends and sequential patterns. S*equential Pattern Mining* [2] is a popular technique which consists of finding subsequences appearing frequently in a set of sequence. However, knowing that a sequence appear frequently is not sufficient for making predictions. Sequential pattern mining [2] approaches are classified as Apriori or generate and test approach, patterngrowth or divide-and-conquer approach. The Apriori and AprioriAll [3] algorithms are based on apriori property and use the generate join procedure to form the candidate sequence. It identifies frequent item set in the database and extends it to a larger item set as those item set appears sufficiently in the database. Some of the widely used apriori based algorithms are GSP [6], SPADE [7] and SPAM [8].
Pattern growth algorithms[4] allow the frequent item set discovery without candidate item set generation. They first build the data structure called FP-tree. Frequent Pattern tree consists of nodes corresponding to items and counters. This tree reads only one transaction at a time and maps it to a path. Then it extracts the frequent item set directly from the FP-tree. Some of the widely used pattern growth algorithms are PrefixSpan [5] and FreeSpan [9]. A projection based pattern-growthmethod is used in PrefixSpan (**Prefix**-projected **S**equential **pa**ttern mining) [5] algorithm for mining sequential patterns. Its major idea is that, instead of projecting sequence databases by considering all the possible occurrences of frequent subsequences, the projection is done on frequent prefix which results in higher efficiency of the algorithm in terms of processing time. The TPrefixSpan algorithm [10] is developed to mine the temporal patterns from interval-based events. Interval-based events are defined as the pair of time values associated with each event. Constraints based PrefixSpan algorithm [11] discovers sequential patterns which are frequent and also satisfy aggregate, length, and item constraint. In this work, Constraint based TPrefixSpan algorithm is proposed to discover frequent temporal patterns considering item, length and aggregate constraints. Sequential pattern mining approaches and algorithms are shown in Fig. 1.

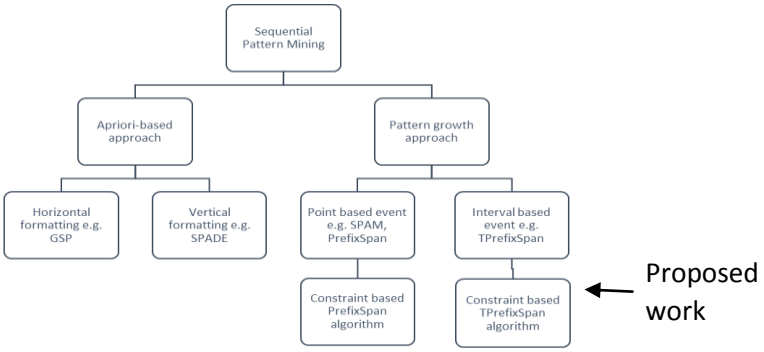

Fig. 1 Sequential Pattern mining approaches and algorithms

II. DATA MINING AND PATTERN MINING

Data mining is the analysis step of theKDD (Knowledge Discovery in Databases) process. It is the intersection of artificial intelligence, machine learning, statistics and database management system (DBMS) [1]. The overall goal of data mining process is to extract information from a data set and transform it into an understandable structure for further use.

Data mining includes

- Medical data mining
- Spatial data mining
- Sensor data mining
- Visual data mining
- Music data mining
- Sequence data mining

Data mining is used in financial data analysis, retail industry, telecommunication industry and biological data analysis [1]. Data mining task includes

1. Sequential pattern mining
2. Association rule mining
3. Frequent item set mining
4. Sequential rule mining
5. Clustering

Next section discusses about Sequential pattern mining and the various existing sequential pattern mining algorithms.

III. SEQUENTIAL PATTERN MINING

Sequential pattern mining deals with finding statistically relevant patterns between data examples where the values are delivered in sequence. It is closely related to time series mining and special case of structural data mining. Some of the applications are analysis of customer purchase patterns or Web access patterns, analysis of time related processes involved in scientific experiments, disease treatments,DNA sequencing etc.

A. Sequential pattern mining algorithms

There are two approaches for Sequential pattern mining. They are Apriori-based approach and Pattern Growth based approach. Most of the earlier algorithms follow an Apriori-based approach.

1) ***GSP:*** The GSP (Generalized Sequential Pattern) algorithm [6] is an Apriori based sequential pattern mining algorithm.It is much faster than AprioriAll algorithm presented by Agarwal [2]. Two steps involved in GSP are Candidate Generation and Candidate pruning method. It has a very good scale up properties with respect to the number of transactions per datasequence and number of items per transaction. But it is not efficient in mining large sequence of databases having numerous patterns or long patterns as it cannot generate more candidate sequence and also multiple scans of database is needed because the length of each candidate grows by one at each database scan.

2) *SPADE:* SPADE (Sequential Pattern Discovery using Equivalence classes) [7] is an Apriori based vertical format sequential pattern mining algorithm i.e. the sequences are given in vertical order instead of horizontal format. In addition, this algorithm uses the *ID-List* technique to reduce the cost for computing support counts. It consists of ID-List pairs where the first value stands for customer sequence and the second value refers to a transaction in it. The algorithm can use a breadth-first or a depth-first search method for finding new sequences. It needs multiple scans of database in mining. Mining long sequential patterns using SPADE isnot possible as it needs an exponential number of short candidates.

3) *SeqDIM:* Previous algorithms are for mining sequential patternsin a single dimension. But Sequential Multi-dimensional mining algorithm [8] is useful in mining multi-dimensional patterns. The main objective of web frequent multi-dimensional sequential pattern mining is to provide the end user with more useful and interesting patterns. It is more efficient and scalable when compared to previous GSP and SPADE algorithms. SeqDIM algorithm is not effective when web frequent multi-dimensional sequences are given as input.

4) *SPAM:* SPAM (Sequential Pattern Mining) [9] uses a vertical bitmap data structure representation of database which is similar to the given id-list of SPADE. It integrates the concept of GSP [7], SPADE [8] and FREESPAN [9] algorithms. SPAM uses a depth-first traversal to increase its performance. SPAM reduces the cost of merging but takes more time and space when compared to other algorithms which can be completely stored in the main memory.

5) *CloSpan:* Instead of mining the complete set of frequent subsequences, CloSpan (Closed Sequential Pattern mining) [12] algorithm mine frequent closed subsequence only, i.e., those containing no super-sequence with the same support. When mining long frequent sequences the performance of previous algorithms often degrades dramatically. CloSpan algorithm will generate significantly less number of sequences than the existing methods.

6) *CMDS:* CMDS (Closed Multidimensional Pattern Mining) [13] is an integration method of closed sequential pattern mining and closed item set pattern mining. This method consists of two major steps: 1. Combination of closed sequential pattern mining with closed item set pattern mining. 2. Elimination of redundant patterns. The number of patterns in CMDS is not larger than the number of patterns in multidimensional pattern mining. The set of CMDS patterns can cover the set of MDS patterns.

7) *BIDE+:* Bi-Directional Extension mining [14] is an efficient algorithm for mining frequent closed sequences without candidate maintenance.This algorithm uses a Back scan pruning technique and the Scan-Skip optimization

technique. It adopts a strict depth-first search order and generates the closed frequent patterns. The cost and execution time is high for mining the patterns.Scalability is poor and occupiesmore memory when compared to other algorithms.

8) *Fournier et al. Framework* [15]*:* It is the combination of Sequential pattern mining algorithms which includes the following features:

i) Mining sequences with minimum support by database-projection (based on PrefixSpan)
ii) Mining sequences with minimum/maximum time interval between events
iii) Mining closed sequences
iv) Mining multi-dimensional sequences
v) Mining closed multidimensional sequences
vi) Mining sequences with items having integer values and performing automatic clustering of these values.

The above discussed algorithms follows a Apriori based approach. The next section discuss about projection based PrefixSpan algorithm which is the fastest algorithm with respect to speed and requires less number of database scans.

IV. PREFIXSPAN ALGORITHM

This is the only projection-based algorithm among the sequential pattern mining algorithms. PrefixSpan is the fastest algorithm among all the algorithms [5]. It outperforms algorithms like Apriori, FreeSpan, SPADE (vertical data format) [7]. It uses divide and search space technique. But in PrefixSpan there are only limited insertions, deletions, and mutations in their sequential patterns. It is anefficient pattern growth method. It outperforms both GSP and FreeSpan. The main idea of prefix span algorithm is thatit explores prefix-projection in sequential pattern mining and mines the complete set of patterns, but reduces the effort of candidate subsequence generation. Prefix-projection reduces the size of projected database and leads to efficient processing. Using bi-level projection and pseudo-projection in PrefixSpan algorithm may improve mining efficiency. PrefixSpan algorithm can be extended in many ways considering point based events, interval based events and also by adding constraints to interesting patterns.

A. I-PrefixSpan algorithm

It is the improved algorithm of PrefixSpan algorithm. The idea of this I-PrefixSpan algorithm [16] is to use sufficient database for Sequential Tree framework and separator database to reduce the execution time andmemory usage. In I-PrefixSpan there is no in-memorydatabase stored after the construction of index set. This I-PrefixSpan algorithm improves PrefixSpan in two ways: (1) to build in-memory database sequence and to construct the index set, it implements sufficient database for Sequential Tree framework and (2) instead of whole in-memory database, to store the transactionalteration signsit implements Separator Database.In this algorithm there is no time constraint and sliding window areused to improve the performance of the output.

B. P-PrefixSpan algorithm

There is no method for extracting a probability of time in the sequential pattern mining [17] process. Besides minimum support-count constraint, this approach imposes minimumtime-probability constraint, i.e., the P-PrefixSpan algorithm is developed by modifying the well-known PrefixSpan algorithm. The new algorithm can discover frequent sequential patterns with probability of inter arrival time of consecutiveitems.The added constraints could filter out less important patterns andreduce the memory space required in storing projected databases.This algorithm is more efficient and scalability is also high when compared to other PrefixSpan algorithms.

C. CFM-PrefixSpan algorithm

This algorithm is designed for mining all CFM (Compact Frequent Monetary Prefix Span) [18] sequential patterns from the given customer transaction database. The CFM-PrefixSpan algorithm employs a pattern growth methodology that finds sequential patterns by utilizing a divide-and-conquer strategy. Besides discovering CF-sequential patterns the compact frequent items and CFM sequential patterns are also discovered. The CFM algorithm has been validated on real and syntheticsequences. The result of this algorithm shows that the effectiveness of sequential pattern mining algorithm can be improved significantly by incorporating monetary and compactness into the mining process.

D. DRL-PrefixSpan algorithm

DRL (Downturn, Revision, and Launch) [19] PrefixSpan is designed specifically to incorporate the specific constraints which involves many steps: i) Product Downturn ii) Product Revision iii) Product Launch. Each of these scenarios is characterized by distinct item and adjacency constraints. This algorithm was developed for mining all length DRL patterns. It has been validated on synthetic sequential databases. It gives the effectiveness of incorporating the promotion-based marketing scenarios in the sequential pattern mining process.

E. C-PrefixSpan algorithm

To save the computation cost and enhance the performance, many types of constraints can be used in sequential pattern mining like item constraint, aggregate constraint, length constraint and gap constraint. Constraint based sequential pattern mining [20] extracts the patterns according to the user's interest. The patterns obtained from C-PrefixSpan are comparatively very less and more valuable than PrefixSpan algorithm. When the number of transaction per sequence increases the performance of C-PrefixSpan algorithm also increases.

F. TPrefixSpan

TPrefixSpan [10] is very similar to PrefixSpan. The TPrefixSpan algorithm is developed for mining the new temporal patterns from interval-based events.Mining temporal patterns is much morecomplicated than mining sequential patterns and themethods for discovering a sequential pattern can neither beused directly nor be applied with slight modifications todiscover temporal patterns.

V. PROPOSED ALGORITHM

Although Sequential Pattern Mining improves the efficiency in many circumstances it still faces tough challenges in terms of effectiveness and efficiency. To overcome this problem and to reduce patterns in large database, various constraints are included. Traditional sequential pattern mining only distinguishes whether a pattern appears or not, while multiple constraints pattern mining approach not only determines the existence of a pattern but also checks whether it satisfies the item, length, and aggregate constraints. Similarly to enhance the effectiveness of TPrefixSpan algorithm the item, length and aggregate constraints are included in the interval based temporal pattern mining algorithm in the proposed work. The architecture of the proposed work is shown in Fig. 2. The input sequences are mined for frequent patterns and using the length, item and aggregate constraints specified by the user the interesting patterns are mined. The resulting interesting patterns can be evaluated using various performance measures.

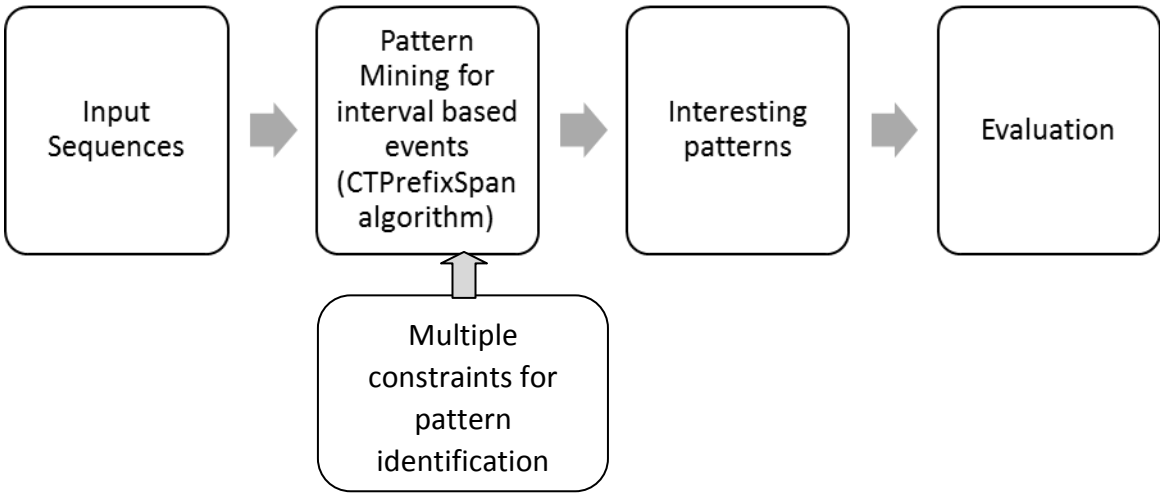

Fig. 2 Architecture of the proposed work

Item Constraints: An item constraint is defined as the subset of items that should or should not be present in the patterns [20].For example, when mining sequential patterns over a disease, a user may be interested in only patterns that are relevant to the disease. The patterns that contain details about the symptoms related to the disease are considered as interesting patterns. This enhances the effectiveness of the algorithm by providing patterns according to user's interest.

Length Constraints: A *length constraint* specifies the requirement on the length of the patterns, where the length can be either the number of occurrences of items or the number of transactions [20]. For example, a user may want to find longer patterns (i.e., at least 4 symptoms) in disease analysis. Such a requirement can be expressed by a length constraint and thus interesting patterns can be identified.

Aggregate Constraint: An *aggregate constraint* is the constraint on an aggregate of items in a pattern, where the aggregate function can be *sum*, *average*, *max*, *min*, *standard deviation* [20]. For example, a user may want frequent sequential patterns where the average number of symptoms related to the disease is over 3.

The proposed CTPrefixSpan algorithm is shown in Fig. 3.

ALGORITHM CTPrefixSpan ( )
Input: Temporal Sequence Database S.
Output: Frequent patterns, Interesting patterns satisfying user specified constraints
Begin
Step 1: Identify frequent temporal 1-patterns in S.
Considering the frequent temporal 1-pattern generated identify the projected database.
Step 2: Generate frequent patterns with frequent 1-pattern as prefix.
Identify the projected database.
Repeat Step 1 and 2 until all the frequent temporal patterns are mined.
Input the item, length, and aggregate constraints specified by the user.
Generate interesting patterns that satisfy the constraints
End

Fig. 3 CTPrefixSpan algorithm

The advantage of the proposed algorithm over TPrefixSpan algorithm is given in Table 1.

TABLE I
COMPARISON OF TPREFIXSPAN ALGORITHM AND PROPOSED CONSTRAINT BASED TPREFIXSPAN

TPrefixSpan	Constraint based TPrefixSpan
Discovers frequent temporal patterns	Discovers frequent patterns and also discovers patterns satisfying length, item and aggregate constraints.
It uses only time constraints	It uses length, item and aggregate constraints along with time constraints.
Efficiency is less (generation of patterns according to user's interest is not possible).	Efficiency and effectiveness is more as it generates patterns according to user's interest.

VI. CONCLUSION

In this paper, we have discussed about various types of sequential pattern mining algorithms. Among all, PrefixSpan algorithm has more advantages like speed,less database scans and high performance. Various types of PrefixSpan algorithms are discussed. Temporal PrefixSpan algorithm mines the temporal patterns from interval based events.Scalability and Data generation are high in this algorithm when compared to PrefixSpan algorithm. The TPrefixSpan algorithm is extended by including various constraints like aggregate, length and item constraints. Using these multiple constraints, it is expected to mine temporal patterns according to user's interest and also it is expected to give more information about the 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. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules", Proc. 1994 International Conference of Very Large Data Bases (VLDB '94), pp. 487-499, Sept. 1994.

[4] J.Han, J.Pei and Y.Yin, "Mining frequent pattern without candidate Generation", Proceeding of ACM SIGMOD International Conference Management of Data, pp.1-12, 2000.

[5] 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.

[6] 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.

[7] M. Zaki, "SPADE: An efficient algorithm for mining frequent sequences", Machine Learning, 2001.

[8] 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.

[9] 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.

[10] Shin-Yi Wu and Yen-Liang Chen, "Mining Non-ambiguous Temporal Patterns for Interval-Based Events", IEEE Transactions on Knowledge and Data Engineering, Vol.19, No.6, June 2007.

[11] Irfan Khan. "PrefixSpan Algorithm Based on Multiple Constraintsfor Mining Sequential Patterns",International Journal of Computer Science and Management Research, Vol. 1, Issue 5, December 2012.

[12] X. Yan, J. Han andR. Afshar, "CloSpan: Mining closed sequential patterns in large datasets", Third SIAM International Conference on Data Mining (SDM), San Francisco,pp. 166–177, 2003.

[13] C.-C. Yu and Y.-L. Chen, "Mining Sequential Patterns from Multi-Dimensional Sequence Data", IEEE Trans. Knowledge and Data Eng., Vol. 17, No. 1, pp. 136-140, Jan. 2005.

[14] J. Wang and J. Han, "BIDE: Efficient Mining of Frequent Closed Sequences", IEEE Trans. Knowledge and Data Eng., 2004.

[15] K.C.Srikantaiah, N. Krishna Kumar, K.R Venugopal and L M Patnaik, "Bidirectional growth based mining and cyclic behavior analysis of web sequential patterns", IEEE Trans. Knowledge and Data Eng., April 2013.

[16] R. DhanySaputra, Dayang, A. Rambli and O.Foong, "Mining Sequential Patterns Using I-PrefixSpan", International journal of electrical and electronics engineering", pp. 338-342, 2008.

[17] H.Shyur, C. Jou and K. Chang, "A data mining approach to discovering reliable sequential patterns", pp 08, 2008.

[18] B.Mallik and D.garg, "CFM PrefixSpan: A pattern growth algorithm incorporating monetary and compactness", pp 4509-4555, July 2012.

[19] A. George and D. Binu, "DRL-Prefixspan: A novel pattern growth algorithm for discovering downturn, revision and launch (DRL) sequential patterns",pp. 426-439, Dec 2012.

[20] 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.

