International Journal of Electrical, Electronics and Computer Engineering **4**(1): 14-19(2015)

ISSN No. (Online): 2277-2626

A Review on Sequential Pattern Mining Algorithms

Sushila S. Shelke* and Suhasini A. Itkar**

**ME Student, Department of Computer Engineering,*
PES Modern college of Engineering, Pune, (MH), India
***Assistant Professor, Department of Computer Engineering,*
PES Modern college of Engineering, Pune, (MH), India

(Corresponding author: Sushila S. Shelke)
(Received 04 December, 2014 Accepted 14 February, 2015)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: Sequential Pattern Mining is one of the data mining technique which determines sequential patterns from immense sequential database. Many wide applications follows sequential pattern mining which includes web usage mining, customer's behavior in shopping history, weather forecasting, medical disease analysis, scientific experiments, etc. In the area of sequential pattern mining abundant research has been performed. In this paper we review a progress on some of the sequential pattern mining algorithms. This paper will explain the evaluation factors, the approach they are following and pros and cons of those algorithms. Finally paper will explain a little of demanding issues that need to be solved in future.

Keyword: Frequent patterns, Sequences, Sequential Pattern Mining, Sequence Database

I. INTRODUCTION

Frequent patterns are nothing but the regular samples which occur frequently in sequence database. Those sample patterns are item sets, subsequences and substructures with rate no less than user defined threshold [1]. The major example of market basket analysis in the structure of association rule mining was first initiated by Agrawal *et al.* (1993) for frequent pattern mining. For example, in the database of shopping history a set of items, such as milk, tea or coffee and sugar that appear frequently together to prepare coffee or tea, is a *frequent item set*. Frequent pattern mining can be classified based on variety of data to be mined, as sequential pattern mining and structural pattern mining.

Sometimes customers first buy a computer, then digital camera followed by memory card; followed by printer to take printout of photo taken by camera; such a frequently happening subsequence in a shopping history database is called (*frequent*) *sequential pattern*. Various structural forms, like subtrees, subgraphs, or sublattices, which may be joined with itemsets or subsequences, are called as substructures. If a substructure comes to mind repeatedly in a database of graph, it is called a (*frequent*) *structural pattern*. To find associations, correlations, and many other exciting relationships between data finding frequent patterns plays very important role in data mining.

The simple definition of Sequential pattern mining say that it is the method used in huge sequence database to discovering sequential patterns. An example of sequential patterns is that in a book store's transaction database history, 80% of customers brought the books in sequences and after certain time break like if any customer wants to study for competitive exam then he first brought the book of mathematics, then after some days they brought the book for aptitude test and the finally he go for buying general knowledge book with certain time gap. All those books need not to be brought at the same time or one after the other, the most important thing is the order or sequence of transaction in which those books are brought and they are bought by the same customer. 80% here represents the percentage of customers who use this purchasing practice [8]. Those sequences will be helpful for the shopkeeper to arrange the books in racks according to this sequence to increase the profit. Sequential prototype can be broadly used in consumer purchase patterns for stock control, for websites to find web access patterns, in scientific research, natural tragedy, medical treatment, investigation of DNA sequences for analysis of sequences or time related processes need to be done**.**

The rest of the paper is structured as follows. We formulate the problem statement and related work of sequential pattern mining in section 2. The progress on sequential pattern mining algorithms is mentioned in Section 3. In section 4 addresses comparative analysis of sequential pattern mining on algorithms. Section 5 concludes the study and explains some challenging issues for future scope.

II. SEQUENTIAL PATTERN MINING

This segment represents the problem announcement and characteristic for sequential pattern mining and related work done in area of sequential pattern mining.

A. Problem Statement

Structured elements or events are part of a sequence database which is documented with or without real time view. The *Sequential pattern mining* is mining of often happening ordered measures or subsequences as patterns was initially projected by [2]. First we introduce the beginning definition of sequential pattern mining.

Let $I = \{i_1, i_2, \dots, i_k\}$ be a set of all substance. A division of I is known as an *itemset*. A *sequence* $\alpha = \langle t_1, t_2, \dots, t_m \rangle (t_i \subseteq I)$ is a prearranged list [3]. All itemsets in a series of sequence correspond to a set of measures experience at the identical timestamp; while unlike itemsets arise at dissimilar times. For example, a sequence of customer purchasing items could be buying several goods on one tour to the stock up and building several successive purchases, e.g., buying a Laptop and some software, go after by buying a digital camera and a memory card, and finally buying a printer and some manuscript.

Without failure of majority, we imagine that the items in every itemset are sorted in definite order (such as alphabetic rate or ascending order). A sequence $\alpha = \langle a_1, a_2, \dots, a_m \rangle$ is a *sub-sequence* of an additional sequence $\beta = \langle b_1, b_2, \dots, b_n \rangle$, indicated by $\alpha \sqsubseteq \beta$ (if $\alpha \neq \beta$, written as $\alpha \sqsubset \beta$), if and only if $\exists i_1, i_2, \dots, i_m$, such that $1 \leq i_1 < i_2, \dots < i_m \leq n$ and $a_1 \subseteq b_{i_1}$, $a_2 \subseteq b_{i_2}, \dots, and\ a_m \subseteq b_{i_m}$. We also call β a *supersequence* of α, and β *contains* α. Given a sequence database $D = \{s_1, s_2, \dots, s_3\}$, the *support* of a sequence α is the number of sequence in D which contain α. If the support of a sequence α satisfies a pre-defined *min_sup* threshold, is a *frequent* sequential pattern.

B. Related Work

There are three main categories where conventional frequent itemset mining methods are divided; which are apriori involves candidate generation, pattern growth which contains no candidate generation and itemset mining for data having vertical format [8].

The main representative of Apriori-based sequential pattern mining approach is proposed in Generalized Sequential Patterns (GSP) [9] which uses a multiple rounds, generation and test approach for candidate set to implement the sequential patterns downward closure property. Further Apriori based algorithms includes SPAM [11] stand for sequential pattern mining using bitmap representation which make use of depth first strategy and vertical bitmap representation for accumulating transactional database and SPIRIT [10] (stand for Sequential Pattern mining with Regular expression constraints) which uses regular expression restriction for search space pruning . Owing to advantages of pattern growth approach like Divide and conquers policy, No candidate generation and compressed database our paper will focus on this approach. Based on pattern growth approach FreeSpan (Frequent Pattern Projected Sequential Pattern Mining) [4] algorithm was developed which uses frequent items to project sequence database into smaller projected database and discover the subsequences in each projected database recursively. Due to generation of many nontrivial projected databases FreeSpan is less efficient. The third approach vertical format-based sequential pattern mining called SPADE [12], which is an lean-to frequent itemset mining of vertical format, like CHARM [14], Eclat [13] and is developed by making use of vertical data to reduce the number of scans of database.

Methods of identifying sequential patterns from database are classifieds in two wide categories. The first one is sequential pattern mining on static database and other one is over the incremented or updated database [3]. The algorithms already discussed above come under static databases. The first work done in incremented or updated database is implemented in FASTUP (Fast Sequential Pattern Update Algorithm) [15] algorithm.

Based on completeness of patterns to be mined are classified as closed sequences are conversed in closed frequent itemset in [16][22-24], constraint based in [19][20] and maximal frequent itemset in [17] [18].

III. SEQUENTIAL PATTERN MINING ALGORITHMS

A. PrefixSpan

Prefix-Projected Sequential Pattern Mining [5]. The main focus of PrefixSpan algorithm is on producing sequential pattern from prefix projected database instead of projected database declared in FreeSpan recursively. First it finds the frequent items having length-1 which supports minimum threshold value from sequential database. Algorithm builds prefix projected database (Database which contains collection of suffixes of sequences with regards to prefix as frequent item for which projected database is going to generate) for each length-1 frequent item and project into smaller prefix projected databases for which it finds the subset of sequential patterns recursively. To progress the major cost of continually generating projected database algorithms suggest a fresh method called Pseudoprojection. Pseudoprojection technique means instead of copying complete suffix sequences in projected database one can register only the index of sequence and staring position of projected suffix in the sequence.

Pseudoprojection technique reduces the size and number of projected database however it is applicable only when projected database is small and can fit into main memory.

Performance of PrefixSpan is tested for with versus without Pseudoprojection. Scalability test is done for different min_support count. Due to pattern growth approach, projection based divide-and-conquer methodology, stable space as no candidate generation PrefixSpan has high performance compare to GSP, SPADE. This algorithm requires more memory for storage of projected database and scanning time of projected database is also large.

B. Sequential Pattern Mining using Rough set theory [6]

Sequential pattern mining using rough theory focused on discovering partial patterns from sequence database by using decision rules. This algorithm calculates subsequences of an unchanging size that are treated as local patterns which are hidden inside sequences. A sequential information system involves the subsequences obtained from a set of sequences so that we can apply sequential data to the rough set data mining. The sequential decision rules are the rules generated from a sequential information system. This algorithm is divided in three main features as Occurrences of Local Patterns, Granularities of Sequences and Reduced and Consistent Decision Rules. In Occurrences of Local Patterns the set of sequences are given then a sequential information system is assembled from the attributes that denote the subsequences of a permanent size, where each attribute value characterized the number of occurrences of a local pattern in a sequence. In Granularities of Sequences the diversity of granularities in a sequential information system is determined by variant sizes of local sequence patterns. In rough set theory, reduced decision rules are generated by attribute reduction. Also logically inconsistent rules are disqualified due to the discernibility of decision modules so the decision rules are consistent in nature.

This paper produces a substitute method for sequential pattern mining using rough set theory. This method represents the local characteristics of sequences by using a sequential information system where attributes correspond to the amount of size k subsequences as local patterns. The proposed mining algorithm computes sequential decision rules according to the size of subsequences by varying the size from 2 to a maximal number in order to check different granulates for sequential data. They evaluate the occurrence-based accuracy and coverage of the sequential decision rules so that local patterns of sequences that result in a decision can be discovered.

C. SPAM (Sequential Pattern Mining using bitmap representation):

This algorithm is suitable for patterns having maximum length in sequence database. It uses lexicographic sequence tree to find the sequential patterns where it uses depth first search strategy to mine the patterns. Each sequence in the sequence tree is either s-extended sequence extended or i-extended i.e. item extended. A sequence-extended sequence is a sequence generated by together with a new operation consisting of a distinct item to the last part of its parent's sequence in the tree. An itemset-extended sequence is a sequence created by totaling an item to the most recent itemset in the parent's sequence, such that the item is superior to any item in that last itemset. SPAM algorithm improves the performance by pruning s-extension and i-extension on each node in tree. Experiment result shows that SPAM performs better than SPADE. But space or storage efficiency of SPAM degrades due to dept-first search strategy.

D. UDDAG (UpDown Directed Acyclic Graph) [7]:

This algorithm uses new data structure, UpDown Directed Acyclic Graph for efficient mining of sequential patterns. DAG (Directed Acyclic Graph) represents patterns as vertexes with ids of transaction containing the pattern and directed edge as relationship of patterns. Up DAG represents DAG for patterns found in Prefix projected database while Down DAG represents DAG for patterns found in Suffix projected database of Frequent item x and by merging both DAG represents UDDAG with root vertex x. PrefixSpan detects the sequential patterns in unidirectional manner whereas UDDAG allows bidirectional pattern growth from both the ends for detected sequential pattern. UDDAG algorithm first assigns unique id to all frequent items in sequence database and converts the original sequential database into transformed database by replacing item sets in each sequence with ids of frequent items involved in the item set. Based on the transformed database, for each frequent item x, the algorithm creates a root vertex for x detects all the patterns from prefix projected database and suffix projected database of x and then it creates directed acyclic graph to represent the containing relationship of frequent items. UDDAG generates less projected database compare to PrefixSpan as UDDAG consider prefix and suffix projected database at the same time. UDDAG eliminates unnecessary candidate generation. Data structure in UDDAG requires more memory storage than PrefixSpan but it involves major cost of storing projected database. UDDAG has some challenging issues like independent frequent item detection, large memory usage for data structure UDDAG.

E. (ISE) Incremental mining of sequential patterns in large databases [21]:

This paper proposed an efficient algorithm, called incremental sequence extraction (ISE), for calculating the frequent sequences in the modified database which is updated by adding new transactions and new customers to the original database.

ISE reduces the computational costs by re-using the least information from the mature frequent sequences, i.e. the support count of frequent sequences. The set of candidate sequences to be experienced is noticeably minimized and this is the main characteristic of ISE. This system uses two optimization techniques. In first optimization reduces the number of candidates significantly by avoiding candidate generation. Second optimization avoids generating candidate sequences that have already been found to be frequent in a previous phase.

ISE is compared with GSP and it shows very good performance over GSP. There is one issue in ISE; this algorithm only considers newly added transactions or customers to the original databases and not for deletion or modification in original database. However for both electronic commerce and web usage mining require deletion or modification to be taken into consideration in order to save storage space or because information is no longer of interest or has become invalid.

F. Mining sequential patterns using Graph Search Techniques [25]:

This paper proposed a new algorithm to find sequential patterns using graph search techniques (GST). First, it finds large 2-sequences (L2), and then makes use of L2 to build the item relation graph (IRG). Then all other large k-sequences (K>=3) can be found by searching through the graph. Since the relation information about items are all in the graph. It is different from the Apriori like algorithms, it can out of order find large k-sequences (k >= 3); i.e., large k-sequences can be found not directly through large (k-1)-sequences. This leads to show that GST algorithm gives better performance than the Apriori-like algorithms. The GST algorithm works in four steps to mine the sequential patterns. First it scans the database to find large-1(L1) sequences from sequence database. Then it joins the L1 with itself to get large-2 sequences L2.In third step it searches the item relation graph to produce large sequences and sequential patterns.

Experiment result shows that GST is superior than Apriori based algorithms.GST algorithm have advantages over the Apriori-like algorithms is that it can generate large sequences without constructing candidate sequences and generate large k-sequences without following from large (k-1)-sequences step-by-step.

IV. COMPARATIVE STUDY

This section will provide the comparative study of progress on methods used in sequential pattern mining algorithms. In this paper we already studied different algorithms in sequential pattern mining like PrefixSpan, Sequential pattern mining using rough set theory, SPAM, UDDAG, Incremental mining of sequential pattern mining in large database and mining sequential patterns using Graph search techniques. Following table provide the comparative study of sequential pattern mining algorithms.

Table 1: Comparison of Sequential Pattern Mining Algorithms.

Algorithms	**Parameters**		
	Strategy used	**Disadvantage**	**Evaluation Parameter**
PrefixSpan	Prefix Projected Database	Big storage for Projected database, Extra scanning time of database	Execution time, Memory usage, Scalability
SPM using Rough Set Theory	Decision Rules	Less efficient for large support count	Minimum Support
SPAM	Depth first strategy	Required more space for search storage	Execution Time, Minimum Support
UDDAG	UDDAG data structure	Large memory for UDDAG data structure, Autonomous Frequent item recognition	Execution Time, Memory Usage
ISE	Incremental Mining	Only focus on added transaction in database not for deletion and modification	Execution Time, Size of updates in original database
GST	Apriori based Approch	Avoid sub path generation in GST	Execution Time, Minimum Threshold

V. CONCLUSION AND FUTURE SCOPE

The main intension of this paper is to review the progress on techniques used in sequential pattern mining and distributed sequential pattern mining. Our studies in sequential pattern mining conclude that pattern growth approach is best suitable for further research effort in this region due to divide and conquer policy, no candidate generation and compressed database. However, few demanding issues in frequent sequential pattern mining which can be comprehensive for future research are good strategy to decrease the scanning time and number of scan in sequential or projected database, design a space resourceful data structure, result of mining process should be independent of predefined parameters like volume of dataset or minimum support threshold. If possible one can move towards the distributed environment and make use of variety of data storage structure to solve the dilemma of sequential pattern mining. More research is required in constraint based and closed sequences in sequential pattern mining.

REFERENCES

[1]. Agrawal R, Imielinski, T, Swami A (1993). Mining association rules between sets of items in large databases. *In: Proceedings of the 1993 ACM-SIGMOD international conference on management of data (SIGMOD'93), Washington, DC*, pp 207–216.

[2]. Rakesh Agrawal, (1995). "Mining Sequential Patterns," Proc. of the Int'l Conference on Data Engineering, pp. 3-14.

[3]. Jiawei Han, Hong Cheng, Dong Xin, Xifeng Yan," Frequent pattern mining: current status and future directions", Data Min Knowl Disc (2007) 15:55–86.

[4]. Han J., Dong G., Mortazavi-Asl B., Chen Q., Dayal U., Hsu M.-C., "Freespan: Frequent pattern-projected sequential pattern mining" *In Proceedings 2000 Int. Conf. Knowledge Discovery and Data Mining (KDD'00), 2000*, pp. 355-359.

[5]. Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, Mei-Chun "Mining Sequential Patterns by Pattern – Growth: The PrefixSpan Approach", IEEE Transaction on Knowledge and Data Engineering, Vol. 16, No. 10, Oct 2004.

[6]. Ken Kaneiwa, Yasuo Kudo, (2011). "A sequential pattern mining algorithm using rough set theory", *International Journal of Approximate Reasoning* **52:** 881–893.

[7]. Jinlin Chen, "An UpDown Directed Acyclic Graph Approach for Sequential Pattern Mining", *IEEE Transaction on Knowledge and Data Engineering*, Vol. **22**, No. 7, Oct 2010.

[8]. Rajesh Boghey, Shailendra Singh," Sequential Pattern Mining: A Survey on Approaches", 2013 International Conference on Communication Systems and Network Technologies, 978-0-7695-4958-3.

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

[10]. M. Garofalakis, R. Rastogi, and K. Shim, (1999). "SPIRIT: Sequential pattern mining with regular expression constraints," VLDB'99, 1999.

[11]. J. Ayres, J. Gehrke, T. Yu, and J. Flannick, (2002). "Sequential Pattern Mining Using a Bitmap Representation," Proc. *Int'l Conf. knowledge Discovery and Data Mining 2002*, pp. 429-435, 2002.

[12]. ZakiM(2001). SPADE: an efficient algorithm for mining frequent sequences. *Mach Learn*, **40**: 31–60.

[13]. ZakiMJ(2000). Scalable algorithms for association mining. *IEEE Trans Knowl Data Eng*, **12**: 372–390.

Zaki MJ,Hsiao CJ (2002). CHARM: an efficient algorithm for closed itemset mining. *In: Proceeding of the 2002 SIAM international conference on data mining (SDM'02), Arlington*, A, pp 457–473.

[14]. M. Lin, S. Lee, "Incremental Update on Sequential Patterns in Large Databases," *In Proceedings of the Tools for Artificial Intelligence Conference (TAI'98), 1*998, pp. 24–31.

[15]. Yan, X., Han, J., and Afshar, R., "CloSpan: Mining closed sequential patterns in large datasets," *In Third SIAM International Conference on Data Mining (SDM), San Fransico, CA*, 2003, pp. 166–177.

[16]. Burdick D, Calimlim M, Gehrke J (2001), "MAFIA: a maximal frequent itemset algorithm for transactional databases", *In Proceeding of the 2001 International conference on data engineering (ICDE'01), Heidelberg, Germany*, pp 443–452.

[17]. Luo C, Chung S (2005), "Efficient mining of maximal sequential patterns using multiple samples", In Proceeding of the 2005 SIAM international conference on data mining (SDM'05), Newport Beach, CA, pp 415–426.

[18]. Jian Pei, Jiawei Han, Wei Wang, "Constraint-based sequential pattern mining: the pattern growth methods," *J Intell Inf Syst.*, Vol. **28**, No.2, 2007, pp. 133 –160.

[19]. Di Wu, Xiaoxue Wang, Ting Zuo, Tieli Sun, Fengqin Yang (2010). "A Sequential Pattern Mining algorithm with time constraints based on vertical format", *2nd International Conference on Information Science and Engineering ICISE)*, 4-6 Dec. 2010, pp. 3479-3482.

[20]. Florent Masseglia, Pascal Poncelet, Maguelonne Teisseire, (2003) "Incremental mining of sequential patterns in large databases", *Data & Knowledge Engineering*, **46:** 97–121.

[21]. Chun-Sheng Wanga, Ying-Ho Liub, Kuo-Chung Chuc, (2013). "Closed inter-sequence pattern mining", *The Journal of Systems and* Software **86**: 1603– 1612.

[22]. Panida Songram, Veera Boonjing, Sarun Intakosum,"Closed Multidimensional Sequential Pattern Mining", In *Proceedings of the Third International Conference on Information Technology: New Generations (ITNG'06)*, 0-7695-2497-4.

[23]. Jinhong Li, Bingru Yang,Wei Song, (2009). "A New Algorithm for Mining Weighted Closed Sequential Pattern", 2009 Second International Symposium on Knowledge Acquisition and Modeling, 978-0-7695-3888-4.

[24]. Yin-Fu Huang, Shao-Yuan Lin," Mining Sequential Patterns Using Graph Search Techniques", *Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC'03).*