International Journal of Advance Research in Science and Engineering
Volume No.10, Issue No. 05, May 2021
www.ijarse.com
IJARSE
ISSN: 2319-8354

A Survey of Mining Sequential PatternsfromUncertain Databases

JAYSHRI HARDE

Computer Science and Engineering Department,
G.H.RaisoniUniversity, Madhya Pradesh, India .

ABSTRACT

Data mining consists of extracting information from data stored in database. Sequential pattern mining is a special case of structured data mining which finds the sub-sequences and the relevant patterns that occurred frequently in the given sequence. It forms base for various fields and applications like medical treatments, customer shopping sequences, DNA sequences and gene structures, weather prediction etc. These applications have data which is uncertain. Many researchers have proposed various algorithms for efficient frequent sequential pattern mining. This paper reviews the classification of various sequential pattern mining algorithms based on two approaches; Apriori based algorithms and Pattern growth algorithms. It also provides brief overview of U-PrefixSpan and FreeSpan which comes under the category of Pattern growth algorithm.

Keywords-*Apriori-based algorithm, pattern growth algorithm, Sequential pattern mining, Top K pattern.*

1. INTRODUCTION

The problem of sequential pattern mining was first addressed by Agrawal and Srikant [1995] [1, 2] and was defined as "consider a database consisting of sequences, where each sequence has a list of transactions ordered by transaction time and each transaction is a set of items, sequential pattern mining is to discover all sequential patterns based on user-defined minimum support, where the support of a pattern is calculated through the number of data-sequences that the pattern contains in the sequences." Sequential Pattern Mining is a well-known data mining technique which consists of finding frequent sub-sequences and patterns which are appearing in a given set of sequence. The pattern observed most often in the sequences is called frequent sequential pattern. The mining of frequent sequential pattern has attracted many researchers attraction due its wide usage in real time application. Frequent Sequential Patterns can be useful for finding DNA sequences, for location tracking of moving objects, in stock market analysis and weather forecasting as well.Uncertainty means simply the events which is not predictable. In real life application especially scientific research, wireless sensor network innate the data uncertainty. Consider the case of wireless sensor network (WSN) where the nodes continuously record the reading for weather temperature, humidity, light within its detection range. The data is uncertain in wireless sensor network because of noise in sensor input and error in wireless transmission. Previously, [1] is the work that studies the mining of sequential pattern on certain data. However this work

adopts the measure of pattern frequentness as expected support which has inherent weaknesses with respect to probabilistic databases and hence ineffective for mining frequent sequential patterns from uncertain databases.

There are two basic approaches for Sequential pattern mining which categorized as Apriori based approach (generate and test approach) and pattern growth approach (divide-and-conquer). The Apriori and AprioriAll [2] algorithms are based on apriori property which states that any super-pattern of a nonfrequent pattern may not be frequent. Some of the widely used apriori based algorithms are AprioriAll, AppriorSome, DynamicSome [1], GSP [2], and SPADE [3]. The efficiency of Apriori based algorithms get affected when the sequence database provided as input is large. Multiple scanning of large databases and generating huge candidate sequences are also other problems for them. So to overcome these issues the Pattern growth algorithms are proposed. These algorithms discover the frequent item set by avoiding candidate item set generation. Thus provide the appropriate sequential pattern in time and space efficient way. FP-tree data structure forms base for deriving sequential patterns from given dataset. PrefixSpan [5] and FreeSpan [4] are well known examples of pattern growth algorithms which are widely used for finding sequential patterns. A pattern-growth method based on projection is used in PrefixSpan (Prefix-projected Sequential pattern mining) [5] algorithm for mining sequential patterns and another algorithm based on PrefixSpan framework i.e U-PrefixSpan [6] develop for uncertain databases. The basic idea behind this method is, rather than projecting sequence databases by computing the frequent occurrences of sub-sequences, the projection is made on frequent prefix. This helps to reduce the processing time which finally increases the algorithm efficiency. The rest of the paper organized as follows: Section 2 reviews the related work and the classification of sequential pattern mining algorithm discuss in section 3. Finally in section 4 concludes the paper.

2. RELATED WORK

Firstly, Sequential pattern mining [1, 2] was introduced by Agrawal and Srikant in 1995, and three algorithms as AprioriSome, AprioriAll and DynamicSome [1] were also proposed by them. Then different parameters such as time constraints, sliding window time are used to generalise the definition of sequential pattern mining and then proposed an Apriori-based, improved algorithm as GSP (Generalized Sequential Patterns). Zaki brought up SPADE [3] algorithm which was based on the equivalence of classes. It was simply the expansion of vertical data format sequential pattern mining method. Then pattern growth method come into exists. Two pattern growth algorithms were proposed by Han, which included FreeSpan [4] and PrefixSpan [5]. Comparing projected databases and subsequence connections, PrefixSpan was more efficient than FreeSpan. Though PrefixSpan is superior but it can handle only deterministic databases and its pattern frequentness measure is expected support which fails to identify the probabilistically frequent sequential patterns. To overcome this problem the new algorithm is studied [6] which is able to capture the intricate relationship between uncertain sequences. U-PrefixSpan uses the prefix projection recursion framework of PrefixSpan algorithm and effectively avoids the problem of possible world explosion. This algorithms can become slow and produce an extremely huge amount of results or too few results, neglecting useful information. This is a critical problem because users have scarcity of resources for analyzing the results. As a result users are only interested in generating a certain amount of results, and refining the parameters can be very time-consuming. Addressing

this problem, an efficient algorithm Top k Sequential Pattern Mining algorithm will be applied to mine only the top-k sequential patterns from many frequent sequential patterns which is more sophisticated for prediction.

3. CLASSIFICATION OF SEQUENTIAL PATTERN MINING ALGORITHMS

Sequential pattern mining algorithms are categorized as Apriori-based algorithms and Pattern-growth algorithms. Apriori-based algorithms are based on generate and test approach. The generate join procedure is used to form candidate sequence by Apriori algorithms. Pattern growth algorithms derive the frequent item sets from given sequences without candidate generation. FP tree data structure which is having nodes corresponding to item and counters is derived first in pattern growth approach. Only one transaction at a time is read and maps it to the path. At last the frequent item set is extracted directly from the FP tree.

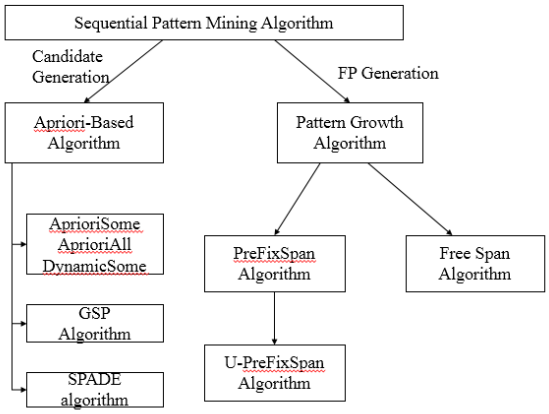

Figure 1. CLASSIFICATION OF SEQUENTIAL PATTERN MINING ALGORITHM

3.1 Apriori based algorithms

Apriori based algorithms are based on association rules which are used to mine the frequent item sets over transactional datasets. The intra-transaction associations are discovered first and then rules are generated about the existing associations. The frequent individual items are identified first then these items are extended to larger item sets which appear often in given dataset.

3.1.1 AprioriAll, AprioriSome, and DynamicSome

These AprioriAll, AprioriSome and DynamicSome algorithms are based on following phases as [1];

i. Sort Phase. This phase transforms the dataset from the original transaction database to a customer sequence database by sorting the dataset by customer id and then by transaction time.

ii. Litemset (large itemset) Phase. This phase checks the item set against the given minimum support. If that item set meets the minimum support value then it gets added in litemset. The litemsets are mapped to a set of contiguous integers so as to optimize the future comparison.

iii. Transformation Phase. Each transaction is replaced by the set of litemsets contained in that transaction so as to transform each customer sequence.

iv. Sequence Phase. In this phase the set of litemsets are mined so as to get the frequent sub-sequences. Potential large sequences (candidates) are produced from the seed set. Those sets which values do not match with the minimum support threshold are deleted and those that remain will be the seed set for the next pass. At start this phase has large 1-sequences and it terminates when neither any candidate is generated nor any candidates meet the minimum support criteria.

v. Maximal Phase. All maximal sequences are found from the set of large sequences in this phase. This phase applied by AprioriSome and DynamicSome algorithms along with the sequence phase so as to save time by not counting non-maximal sequences. The phase works in similar way by which all subsets are derived from given itemset and due to this the algorithm for performing this task is also similar.

3.1.2 Apriori based GSP Algorithm

The Apriori basedGSP (Generalised Sequential Pattern) algorithm [2] is based on Apriori property and works faster than AprioriAll. Candidate Generation and Candidate Pruning are two basic steps for GSP algorithm. Following are the steps in Apriori based GSP algorithm;

i. Through first scan, all the frequent items are identified and are store as the set of single item frequent sequences.

ii. The sequential pattern's set which is found in the previous pass are stored in seed set. Each subsequent pass starts with this seed set.

iii. Candidate sequences are derived from this seed set. More than one item is present in candidate sequence than a seed sequential pattern. The Lengthof the sequence is calculated through the total number of items in a sequence. The support for each candidate sequence is found out by scanning the database in one pass.

iv. The support value for all the candidates which is more than minimum support, are categorised into the newly found sequential patterns set which will act as the seed set for the next pass.

v. When there will be no new sequential pattern is found and candidate sequence will not be generated, the algorithm gets terminated.

The performance of algorithm affected while mining long sequential patterns. It also performs multiple scans of databases as the size of candidate sequence goes on increasing with each scan which increases the time complexity.

3.1.3 SPADE Algorithm

Zaki proposed SPADE (Sequential Pattern Discovery using Equivalence classes)algorithm [3]. In this algorithm sequences are provided in vertical format rather than horizontal form. SPADE has a large set called as vertical id-list database that consists of Sequence ID (SID) and Event ID (EID). The algorithm maps the sequential dataset to this id list. The sub-sequences are grown by Apriori based candidate generation through the one item at time scheme. Breadth first search and depth first search methods are used to get new sequences.

SPADE has following advantages over GSP;

a. Temporal joins are used by SPADE on id lists so as to reduce its size when length of frequent sequence goes on increasing.

b. Generation of pattern and sequential pattern searching is easy as simple data structure is used which helps to reduce the overhead.
c. I/O cost is also reduced by restricting database scans.

Transformation of horizontal format into vertical layout requires extra computation time which reduces SPADE's efficiency. It also needs additional space for storage.

3.2 Pattern-growth algorithms

Apriori based algorithms face the problem of huge candidate sequence generation. 2^{200} candidate sequences are generated for 200 elements of frequent sequence. It is very time consuming and space affecting process to mine such enormous number of candidate sequences to mine sequential patterns. The repetitive scanning of dataset is also another problem for Apriori-family algorithms. That scanning is based on some pattern matching method and it is done to find large set of candidates.

Many researchers and scientists work on these problems and come up with new approach termed as Pattern growth approach. The large candidate sequence generation is eliminated by pattern growth algorithms. As frequent item sets are derived by avoiding the candidate generation, the repetitive scanning problem is also resolved. FP tree data structure forms the base for finding frequent item set from given sequence. FreeSpan and PrefixSpan are two well known examples of pattern growth algorithm.

3.2.1 FreeSpan algorithm

FreeSpan (Frequent pattern-projected Sequential Pattern Mining) algorithm [4] was given by Han et al. so as to reduce the cost for scanning multiple projected databases. The divide and conquer methodology is used so as to derive projected databases from given sequence database recursively. It is helpful to grow sub-sequences into each projected database. The data and the set of frequent patterns which are going to be tested are partitioned first. The smaller projected databases are responsible for conducting the test.

Initially the support for each item is calculated by scanning data sets and frequent item sets are derived then. The frequent items are listed based on descending order of their support. At each projection smaller and manageable units are derived by partitioning the database and restricting the testing.

FreeSpan has following advantages;

i. Candidate sequence generation is avoided by providing small projected database from large sequence database.
ii. The cost for scanning multiple projected databases is also reduced.
iii. Large sequential patterns are processed efficiently.

It faces the problem of sequence duplication due to which the same sequence can be occurred in multiple projected databases.

3.2.2 PrefixSpan algorithm

Jian Pei et al. proposed a novel algorithm named PrefixSpan (Prefix-projected Sequential Pattern Mining) algorithm [5] which works on projected database and sequential pattern growth. The divide and search space technique is implemented by PrefixSpan. Algorithm mines sequential patterns through following steps;

i. Find length-1 sequential patterns. The given sequence S is scanned to get item (prefix) that occurred frequently in S. For the number of time that item occurs is equal to length-l of that item. Length-l is given by notation <pattern> : <count>.

ii. Divide search space. Based on the prefix that obtained from first step, the whole sequential pattern set is partitioned in this phase.

iii. Find subsets of sequential patterns. The projected databases are constructed and sequential patterns are mined from these databases. Only local frequent sequences [6] are explored in projected databases so as to expand the sequential patterns. The cost for constructing projected database is high. Bi-level projection and pseudo-projection methods are used to reduce this cost which ultimately increases the algorithm's efficiency.

The PrefixSpan has following advantages:

a. No candidate generation.
b. The frequency of local items only countable.
c. Divide-and-conquer search methodology is used.
d. It is superior to GSP and FreeSpan.

But still there is need to improve the PrefixSpan algorithm because it is not applicable to uncertain databases.

3.2.2.1U-PrefixSpan Algorithm

Z. Zhao et al. proposed algorithm named U-PrefixSpan adopts the prefix-projection recursion framework of the PrefixSpan algorithm that conform to Sequence-Level uncertain model which uses pattern growth algorithm called SeqU-PrefixSpan applied to mine the frequent sequential patterns from uncertain sequences. This algorithm effectively avoids the problem of "possible world explosion". Steps are as follows:

1)Scan S|α once,

find the set of frequent items b such that:

- b can be assembled to the last element of α to form a sequential pattern; or
- <b> can be appended to α to form a sequential pattern.

2) For each frequent item b:

- append it to α to form a sequential pattern α' and output α';
- output α';

3)For each α':

- construct α'-projected database S|α' and
- call PrefixSpan(α', L+1, S|α').

4. TOP K SEQUENTIAL PATTERN MINING ALGORITHM

U-PrefixSpan algorithm generates large amount of frequent sequential patterns and therefore can become slow and produce an extremely huge amount of results or too few results, neglecting useful information. This is a critical problem because users have scarcity of resources for analyzing the results. As a result users are only interested in generating a certain amount of results, and rectifying the parameters can be very time-consuming. Addressing this problem, an efficient algorithm Top-K sequential Pattern Mining algorithm will be

applied to mine the top-k sequential pattern from sequential databases. The main procedure for Top KSequential pattern mining is as follows:

- Firstly, Set minimum support is 0.
- Then applySequence Pattern Mining using A Bitmap Representation (SPAM) to explore the pattern search space.
- Set the variable L which contains the current top-k patterns found until now.
- When k patterns are found, raise minsup to the support of the least frequent pattern in L.
- Then for each pattern added to L, raise the minsup threshold.

Top K sequential pattern mining algorithm applied over the U-PrefixSpan algorithm achieves better performance and has following advantages:

1. The Top K Sequential pattern mining uses SPAM approach which performed well on large dataset due to vertical representation of the data for efficient counting.
2. The counting process is critical because it is performed many times at each recursive step and SPAM handles it in an very efficient manner.

5. CONCLUSION

In this paper, various sequential pattern mining algorithms like AprioriAll, AprioriSome, DynamicSome, GSP, SPADE, FreeSpan and PrefixSpan, U-PrefixSpan are discussed. All these mining algorithms are categorised as Apriori-based algorithms and Pattern-growth algorithms. PrefixSpan algorithm works well for deterministic databases but fail to work on uncertain databases hence to overcome the problem of PrefixSpan, the U-PrefixSpan is applied for mining probabilistically frequent sequential patterns from uncertain databases and TKSPM is also discussed in brief. TKSPM mines only top k patterns from many frequent sequential patterns which is more sophisticated for prediction. By combining U-PrefixSpan and Top K sequential Pattern Mining algorithms reduces the size of projected database and time for scanning the projected database thus the efficiency of algorithm improves.

REFERENCES

[1] R Agrawal and R Srikant, Mining sequential patterns, International Conference on Data Engineering (ICDE'95) Taiwan, 1995, pp.3- 14.

[2] R Agrawal and R Srikant, Fast algorithms for mining association rules, International Conferenceon Very Large Data Bases (VLDB'94) Chile, 1994, pp. 487- 499.

[3] M. Zaki, SPADE: An Efficient Algorithm for Mining Frequent Sequences, Machine Learning, vol. 40, 2001, pp. 31- 60.

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

[5] Jian Pei, Jiawei Han, Behzad Mortazavi, UmeshwarDayal,Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach, IEEE transactions on knowledge and data engineering, Vol. 16, 2004 pp. 1424-1440.

[6] Zhou Zhao, Da Yan and Wilfred Ng,MiningProbabilistically Frequent Sequential Patterns in Large Uncertain Databases,IEEE transactions on knowledge and data engineering, Vol. 26, 2014 pp. 1171-1184.

[7] K. Rana et al,An Effective Approach to Mine Frequent Sequential Pattern over Uncertain dataset, International Journal of Computer Science and Information Technologies, Vol. 6 , 2015 pp. 3242-3244.

[8] Bi-Ru Dai, Hung-Lin Jiang and Chih-Heng Chung, Mining Top-K Sequential Patterns in the Data Stream Environment, International Conference on Technologies and Applications of Artificial Intelligence, 2010, pp.142-149.