A Performance Analysis of Sequential Pattern Mining Algorithms

Karishma B Hathi, Jalpa A Varsur, Sonali P Desai, Sagar R Manvar
PG Student: CSE Department, B.H.Gardi College of Engg&Tech, Rajkot, Gujarat India

Abstract : Sequential pattern mining is a very important mining technique with wide applications. It found very useful in various domains like natural disaster, sales record analysis, marketing strategy, shopping sequences, medical treatment and DNA sequences etc. It discovers the subsequence's and frequent relevant pattern from the given sequences in the database. The first was Apriori algorithm, which was put forward by the founders themselves. Later more scalable algorithms for difficult applications were developed. E.g. GSP, Spade, PrefixSpan etc. In this paper, a survey of the sequential pattern mining algorithms is given.

***Keywords* - Sequential Pattern, Sequence Database, Itemsets, Apriori, Pattern Growth.**

INTRODUCTION

The sequential pattern mining is a very important concept of data mining, a further extension to the concept of association rule mining [1]. That has a huge range of real-life application. This mining algorithm solves the problem of discovering the presence of frequent sequences in the given database [2]. Sequential Pattern Mining finds interesting sequential patterns among the huge database. It discovers frequent subsequences as patterns from a given sequence database. With large amounts of data continuously being collected and stored, various industries are becoming interested in mining sequential patterns from their database. Sequential pattern mining is one of the most well-known methods and has wide applications including web-log analysis, customer purchase behavior analysis and medical record analysis [1]. In the retailing business, sequential patterns can be mined from the transaction records of customers. For example, having bought a camera, a customer comes back to buy a cover and a SD card next time. The retailer can use such information for discovering the behavior of the customers, to understand their interests, to satisfy their demands, and above all, to predict their needs.In the medical field, sequential patterns of symptoms and diseases exhibited by patients discover strong symptom/disease complementary relations that can be a valuable source of information for medical diagnosis and preventive medicine. In the Web log analysis, the exploring behavior of a user can be obtained from member records or log files. For example, having viewed a web page on ―Cards ―Birthday cards. These sequential patterns yield huge benefits, when acted upon, increases customer royalty.

A. *Basic Concepts of Sequential Pattern Mining[9]*

1) Let $I = \{x_1, . . . , x_n\}$ be a set of *items*, each perhaps being associated with a set of *attributes*, such as value, price, profit, calling distance, period, etc. The value on an attribute *A* of item *x* is denoted by *x.A*. An *itemset* is a non-empty subset of items, and an itemset with k iems is called k itemset.

2) A *sequence*=<*X*· · *α* · *X* > is an ordered list of itemsets. An itemset X_i $(1 \leq i \leq l)$ in a sequence is called as a *transaction*, a term emerged from shopping sequences in a transaction database. A transaction *Xi* may have an exceptional attribute, *time-stamp*,denoted by *Xi.time*, which registers the time when the transaction was executed. For a sequence $\alpha= <X_1 \cdot \cdot \cdot X_l>$, we assume that *Xi.time* < *Xj.time* for $1 \leq i < j \leq l$.

3) The number of transaction in a sequence is called as *the length of sequence*. A sequence with length *l* is called *as* an *l-sequence*. For an *l*-sequence *α*, we have *len (α)=l*. Moreover, the *i*-th itemset is denoted by *α*[*i*]. An item can arise at most once in an itemset, but can arise multiple times in various itemsets in a sequence.

4) A sequence $\alpha= <X_1 . . . X_n>$ is called as a *subsequence* of another sequence $\beta= <Y_1 . . .Y_m>$ $(n \leq m)$,), and *β*a *super-sequence* of *α*, if there exist integers $1 \leq i_1 <..< i_n \leq m$ such that $X_1 \subseteq Y_{i1},.., X_n \subseteq Y_{in}$.

5) A *sequence database SDB* is a set of 2-tuples *(sid, α)* where *sid* is a *sequence-id* and *α is* a sequence.
A tuple *(sid, α)* in a sequence database SDB *is said to hold* sequence *γ* if *γ* is a subsequence of *α*. The number of tuples in a sequence database *SDB* holding sequence *γ* is called the *support* of *γ*, denoted by *sup (γ)*. Given a positive integer *min_sup* as the *support threshold*, a sequence γ is a *sequential pattern* in sequence database *SDB* if *sup≥min(γ)sup.* The *sequential pattern mining* problem is to discover the *complete* set of sequential patterns with respect to a given sequence database *SDB* and a support threshold *min_sup.*

CLASSIFICATION OF SEQUENTIAL PATTERN MINING ALGORITHM

In recent years many approaches in sequential pattern mining have been proposed, these studies cover broad portion of issues [11]. In general there are two valuable concerns in sequential pattern mining.

1. The first one and very important one is to increase the performance or efficiency and accuracy in sequential pattern mining process.
2. Expand the mining of sequential pattern to the time related constraint.

Sequential pattern mining is broadly classified into Two groups:
a) Apriori Based.
b) Pattern Growth Based.

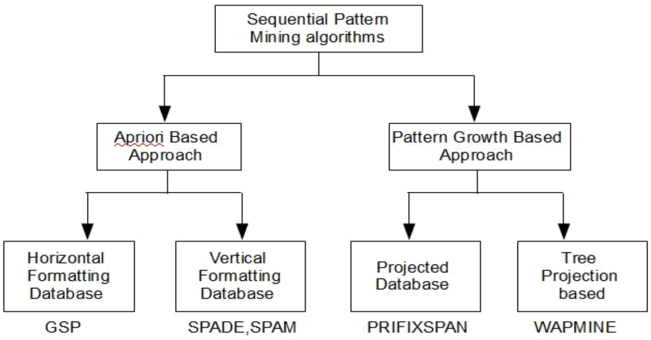

Sequential Pattern Mining algorithms
Apriori Based Approach
Pattern Growth Based Approach
Horizontal Formatting Database
Vertical Formatting Database
Projected Database
Tree Projection based
GSP
SPADE,SPAM
PRIFIXSPAN
WAPMINE

Fig.1 **Classification of Sequential Pattern Mining Algorithm**

a) Apriori Based Algorithms

The Apriori and AprioriAll algorithms set the basic for a set of algorithms that relies largely on the appriori property and use the appriori- generate joint procedure to generate the candidate sequences. As per the apriori statement property all the nonempty subset from the frequent item set much also be the frequent. That is also be described as (downward-closed) in that if a sequence cannot satisfy the minimum support test, than its entire super sequence will also fail the test/condition.

Important terms of the apriori -based algorithms are[3]

1) *Breadth-first search technique used*: Basically the apriori based algorithms are work on this technique. Apriori-based algorithms are described as breath-first (level-wise) search algorithms because they construct all the k-sequences, in kth iteration of the algorithm, as they traverse the search space.[15]
2) *Generate-and-Test*: This kind of feature is used by the very early algorithms from initials research done in sequential pattern mining algorithms which depend on this technique only shows the inefficient pruning method and create large number of candidate sequences and then test each one sequentially for satisfying some user specified constraints consuming huge memory in the early stage of mining.
3) *Multiple scan of the database*: This feature is very unwanted because it requires the lots of processing time and IO cost.

i) **GSP (Generalized Sequential Pattern)**- algorithms is represented by Agrawal and Shrikant [4] makes the multiple passes on the data. This algorithm is much faster than the AprioriAll algorithm. In the GSP algorithm the two steps are involved, first one is candidate generation and candidate pruning method. The algorithm is not a main memory algorithm generates only as many candidates as will suitable in memory and the support of the candidate is find out by scanning the dataset. Frequent Sequences from these candidates are written to disk and the candidates which are without minimum support are deleted [10]. The same step is iterated until every candidate has been counted. The GSP algorithm searches all the length-1 candidates (using one database scan) and orders them by their support value ignoring whose support<min_support. Then for each level (i.e. sequences of length k) the algorithm scans the dataset to gather the support count of the each candidates sequence and generates candidates of length (k+1) sequence from length-K frequent sequences using apriori. This step is continued until there is no frequent sequence or no candidates can be found.

This algorithm has a very effective scale up properties with respect to the number of transaction per data sequence and number of items per transaction. But this algorithm is less than efficient where the mining in large sequencing of databases having countless pattern or long patterns as it cannot generates any more candidates sequence and also several scans of database is needed because the length of each candidates grows by one at each database scan.

ii) **SPIRIT -** The basic concept behind this algorithm is to use the regular expression at flexible tool for the constraint specifications [12]. It gives the generic user specified regular expression constraint on the mined pattern, for giving the more powerful restriction. There are various versions in the algorithm. The selection of the regular expression as a constraint specification tool is considered on the basic of two valuable factors. The first one regular expression is the simple form and natural syntax for specification of families of sequential pattern and second it has the more power for specifying wide range of interesting pattern constraints.

iii) **SPADE -** As like horizontal formulating methods (GSP) the sequential dataset can be converted into a vertical dataset format consisting of item id-lists [5]. The vertical dataset list is the list of (sequential-id, timestamps) pair showing the occurring timestamps of the item in that sequence. The finding in the format of dataset is done by the id-list interaction, this SPADE algorithm complete the mining in total three passes of database scanning. In addition to this the computation time requires to convert in the horizontal dataset to vertical dataset and also require additional storage space several times larger than that of the original sequence database

iv) **SPAM -** SPAM combines the ideas of GSP, SPADE, and FreeSpan [6]. This algorithm uses the vertical bitmap data structure representation of database which is similar to the given id-list of SPADE algorithm. The entire algorithm with its data structure fits in the main memory. For the performance grow the SPAM use the depth-first traversal fashion. SPAM is similar to SPADE, but it uses the bitwise operations on behalf of the regular and temporal join when the comparison of SPAM and SPADE is consider the SPAM outplay more than SPADE, while the SPADE algorithm is more SPACE-efficient than SPAM.

v) **CloSpan-** CloSpan (Closed Sequential Pattern Mining) algorithm mines the frequent closed sub sequences only [6]. That is, those containing no super-sequences with the equal support when mining long frequent sequence. The performance of algorithms decreases dramatically. This algorithm creates the lower number of sequences than the other algorithms.

vi) **CMDS -** (Closed Multidimensional Pattern Mining) is an combines method of closed- item set pattern mining and closed sequential pattern mining [13]. It consist of two steps-

- Fusion of closed sequential pattern mining with closed item set pattern mining.
- Deletion of redundant pattern.

The number of pattern in CMDS is lower than the number of pattern in multidimensional pattern mining. The set of CMDS pattern can cover the set of MDS pattern.

b)Pattern-growth Sequential Pattern Mining Algorithms

The Pattern Growth algorithm comes in the early 2000s, for the answer to the problem of generates and test. The main idea is for to avoid the candidate generation step altogether, and to concentrate the search on a specified portion of the initial database. In this kind of the algorithm the technique of search space partitioning is plays main role in pattern-growth. In this kind of algorithm begins by building a representation of the database to be mined, and after that explains the way to partition the search space and generates the candidates' sequences by growing on the initially mined frequent sequences. The initial algorithm begins by using projected databases, which is free-span, prefix span with latter one being most influential.

i) **PrefixSpan-** The PrefixSpan (Prefix Projected Sequential pattern Mining) algorithms represented by Jian Pei, Jiavei Han and Helen Pinto[7] is the only projection based algorithms from all the sequencing pattern mining algorithms. It performs higher than the algorithm like apriori, SPADE (vertical data format). This algorithm searches the frequent items by scanning the sequence database once. The database is projected into many smaller databases according to the frequent items. By recursively increasing subsequence fragment in every projected database, we got the complete set of sequential pattern. The chief concept behind the prefix span algorithm to successfully discovered patterns is employing the divide-and-conquer strategy. The prefix span algorithm requires high memory space as differentiate to the other algorithms in the sense that it requires creation and processing of large number of projected sub-databases.

ii) **FREESPAN-** The freespan algorithm reduces the cost require to candidate generation and testing of apriori, with satisfying its basic feature [14]. In short, the freespan algorithm uses the frequent items to repetitively project the sequence database into projected database while growing subsequence's frequently in each projected dataset. Every projection divides the database and confines further testing to progressively small-scale and more manageable units. The valuable issue is to considerable amount of sequences can appear in more than single projected database and the size of database decreases with each iteration.

iii) **WAP-MINE-** WAP-MINE is pattern-growth based algorithm with tree-structure mining technique on its WAP-tree data structure[8]. In this algorithm the sequence database is scanned twice to construct the WAP-tree from the frequent sequences by their support values. Here header table is managed first to point that where is first occurrence of the each item in a frequent item set which can be useful to mine the tree for frequent sequences built up on their suffix. It found in the analysis that the WAP-MINE algorithm have more scalability than GSP and perform sharply by marginal points. Although this algorithm scans the database twice only and avoids the problem of generating big amount of candidate as in case of apriori-based approach, the WAP-MINE faces the problem of memory consumption, as it iteratively regenerate n increase automatically.

COMPARATIVE STUDY OF SEQUENTIAL PATTERN MINING ALGORITHM

Comparative analysis of sequential pattern mining algorithm is done on the basis of their different important features. For comparison sequential pattern mining is subdivided into two broad categories, namely, Apriori Based and Pattern Growth Based Algorithms. All the nine features used to categorize these algorithms are discussed first and then comparison is done for the following algorithms

GSP: Generalized Sequential Patterns
SPADE: Sequential Pattern Discovery using Equivalence Classes
SPAM: Sequential PAttern Mining
Prifixspan: Prefix Projected Sequential pattern Mining
WAP-MINE: Web Access Pattern Mining

Characteristics of Sequential Pattern Mining Algorithm are:

Apriori-Based vs. Pattern-Growth-Based

Apriori-based algorithms usually use a candidate generate-and-test‖ type of approach ,which utilize the downward closure property:

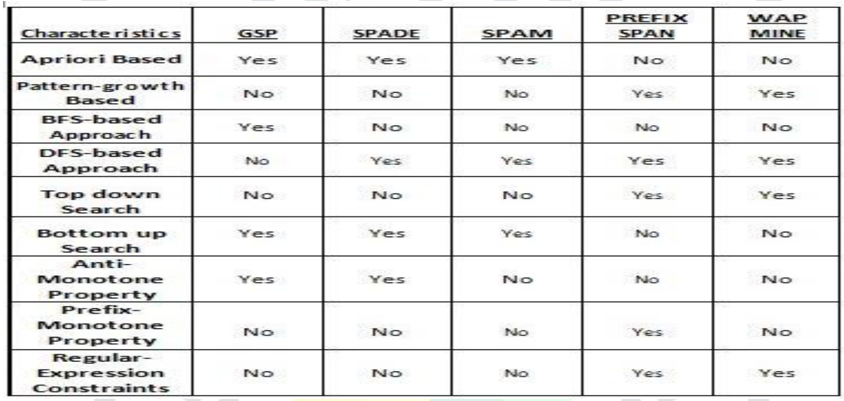
if an itemset α is not frequent, then any superset of α must not be frequent either, Pattern-growth algorithms use a more incremental approach in producing possible frequent sequences, and use what might be called a divide-and-conquer approach. Pattern-growth algorithms make projections of the database in an attempt to decrease the search space.

BFS-Based Approach Vs. DFS-Based Approach In a BFS approach level-by-level search can be conducted to search the complete set of patterns i.e. All the children of a node are processed before proceeding to the next level. On the other hand, when using a depth-first search approach, all sub-arrangements on a path must be traversed before proceeding to the next one. The advantage of DFS over BFS is that DFS can very rapidly reach large frequent arrangements and therefore, some expansions in the other paths in the tree can be neglected.

Top-Down Search Vs. Bottom-Up Search Apriori-based algorithms utilize a bottom-up search, enumerating every single frequent sequence. This implies that in order to produce a frequent sequence of length l, all 2_l subsequences have to be produced. It can be easily concluded that this exponential complexity is limiting all the Apriori-based algorithms to find only short patterns, since they only implement subset infrequency pruning by eliminating any candidate sequence for which there exists a subsequence that does not belong to the set of frequent sequences. In a top-down approach the subsets of sequential patterns can be mined by building the corresponding set of projected databases and mining each recursively from top to bottom.

Anti-Monotone Vs. Prefix-Monotone Property In Anti-Monotone property states that every non-empty sub-sequence of a sequential pattern is a sequential pattern, while Prefix-Monotone property states that if for each α sequence fulfilling the constraint, so does every sequence having α as a prefix also fulfills the constraint.

Table1: Comparative study of sequential pattern mining algorithms



Characteristics	GSP	SPADE	SPAM	PREFIX SPAN	WAP MINE
Apriori Based	Yes	Yes	Yes	No	No
Pattern-growth Based	No	No	No	Yes	Yes
BFS-based Approach	Yes	No	No	No	No
DFS-based Approach	No	Yes	Yes	Yes	Yes
Top down Search	No	No	No	Yes	Yes
Bottom up Search	Yes	Yes	Yes	No	No
Anti-Monotone Property	Yes	Yes	No	No	No
Prefix-Monotone Property	No	No	No	Yes	No
Regular-Expression Constraints	No	No	No	Yes	Yes

Regular Expression Constraint Complexity of regular expression constraints can be approximately measured by the numbers of state changes in their corresponding deterministic finite automata. A regular expression constraint has a good property called growth-based anti-monotonic. A constraint is growth-based anti-monotonic if it has the succeeding property: If a sequence fulfills the constraint must be reachable by growing from any component which matches part of the regular expression. From the comparative study of table 1, it is easy to understand that PrefixSpan algorithm uses depth first search based approach, top down search which are efficient techniques to search frequent subsequences as sequential patterns form the large database. Also PrefixSpan utilize regular expression constraints as well as prefix monotone property, which makes this algorithm a clear choice for applying user defined constraints for mining only some concerned sequential patterns.

CONCLUSION

In this paper, we considered what is a sequential pattern mining and different types of their algorithms. This concept is being introduced in 1995[2] has gone through astonishing advancement in few years only. Initial work on this topic is focused on improvement of the performance of algorithms by using various data structure or various representations. So, on the basis of these problems the sequential pattern mining is divided into two main groups, Apriori approach based algorithms and pattern growth approach based algorithms. From our comparative study and previous some studies by different researchers on sequential pattern mining algorithms it is found that the algorithm which are based on the approach of pattern growth are superior in terms of scalability, time-complexity and space-complexity.

REFERENCES

[1] J. Han and M. Kamber, “Data Mining: Concepts and Techniques”, Morgan Kaufman publishers, 2001.

[2] R. Agrawal and R. Srikant, “Mining Sequential Patterns”, In Proceedings of the 11th International Conference on Data Engineering, pp. 3-14, Taipei, Taiwan, 1995.

[3] NIZAR R. MABROUKEH and C. I. EZEIFE, A Taxonomy of Sequential Pattern Mining Algorithms, ACM Computing Surveys, Vol. 43, No. 1, Article 3, Publication date: November 2010.

[4] Jian Pei, Jiawei Han and Wei Wang, "Constraint-based sequential pattern mining: the pattern-growth methods", Journal of

Intelligent Information Systems, Vol:28, No: 2 ,pp:133-160, 2007.

[5] Mohammad J. Zaki,- SPADE: An Efficient Algorithm for Mining Frequent Sequences, Kluwer Academic Publisher. Machine Learning,42,31-60,2001.

[6] 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.

[7] J. Pei, J. Han, B. Mortazavi-Asi, H. Pino, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix- Projected Pattern Growth", ICDE'01, 2001.

[8] 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.

[9] Chetna Chand, Amit Thakker, Amit Ganatra," Sequential pattern mining survey & current research challenges ", International journal of soft computing and Engineering(IJSCE),ISSN: 2231-2307,Vol.2,Issue-1,March 2012.

[10] Vishal S. Motegaonkar, Prof. Madhav V. Vaidya,”A survey on sequential pattern mining algorithms”,International Journal of Computer Science and Information Technologies (IJCSIT),Vol.5 (2),2014.

[11] J.Pei, J.Han, B.MortazaviAsl, J.Wang, H.Pinto, Q.Chen, U.Dayal and M.-C.Hsu, “Mining sequential patterns by pattern growth: The prefixspan approach”, IEEE Transactions on Knowledge and Data Engineering, Vol.16, no.11, 2004.

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

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

[15] Irfan Khan, Anoop Jain, “A Comprehensive Survey on Sequential Pattern Mining”, International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 4, June – 2012.

Author Profiles:

1)Hathi Karishma is a student of Masters of Engineering in Computer Science and Engineering at B.H.Gardi College of Engineering and Technology, Rajkot, Gujarat, India. She is bachelors in Computer Science and Engineering. Her area of interest are Data Mining, Computer networking and Security. Contact:+91 9429810304

2)Desai Sonali is a student of Masters of Engineering in Computer Science and Engineering at B.H.Gardi College of Engineering and Technology, Rajkot, Gujarat, India. She is bachelors in Computer Science and Engineering. Her area of interest are Data Mining, Computer networking and Security.Contact:+91 9408966536

3)Varsur Jalpa is a student of Masters of Engineering in Computer Science and Engineering at B.H.Gardi College of Engineering and Technology, Rajkot, Gujarat, India. She is bachelors in Computer Science and Engineering. Her area of interest are Data Mining, Software Engineering and Information Security. Contact: +91 9925460983

4)Manvar Sagar is a student of Masters of Engineering in Computer Engineering at B. H. Gardi College of Enggineering and Technology, Rajkot, Gujarat, India. He is bachelors in Information Technology. His area of interest are Data Mining,Computer networking and Security.
Contact:+91 9586507454