Data Mining in Sequential Pattern for Asynchronous Periodic Patterns

Thodeti Srikanth
Research Scholar, Ph.D. (Computer Science),
Dravidian University, Andhra Pradesh, INDIA

Abstract- Data mining is becoming an increasingly important tool to transform enormous data into useful information. Mining periodic patterns in temporal dataset plays an important role in data mining and knowledge discovery tasks. This paper presents, design and development of software for sequential pattern mining for asynchronous periodic patterns in temporal database. Comparative study of various algorithms on sequential pattern mining for asynchronous periodic patterns is also carried out by taking artificial and real life database of glossary shop. The proposed system will be based on optimization of Efficient Mining of Asynchronous Periodic Pattern Algorithm (EMAP), which will be implemented for efficient mining of asynchronous periodic patterns in large temporal database.

Keywords- Sequential patterns, Temporal dataset, Knowledge discovery, Asynchronous Periodic patterns,

I. INTRODUCTION

Pattern mining plays an important role in data mining tasks. Various patterns have been introduced for different applications, e.g., frequent item sets and sequential patterns for transaction databases, frequent episodes in event sequences, and frequent continuities for inter transaction association. Periodic patterns are recurring patterns that have temporal regularities in time-series databases. Periodic patterns exist in many kinds of data. For example, tides, planet trajectories, somite formation, daily traffic patterns, and power consumptions all present certain periodic patterns. There are many emerging applications, including stock market price movement, earthquake prediction, telecommunication network fault analysis, repeat detection in DNA sequences and occurrences of recurrent illnesses, etc. The discovery of patterns with periodicity has been studied in several works. For example, Ozden et al. proposed the mining of cyclic association rules that reoccur in every cycle of the time span of the temporal database. Han et al. considered imperfect periodic patterns that reoccur for at least minconf percent of the cycles. Berberidis et al. further proposed an approximate periodicity detection algorithm. However, these studies considered only synchronous periodic patterns and did not recognize the misaligned presence of patterns due to the intervention of random noise. For example, assume that a temporal database contains a periodic pattern, “burger and maggi,” on Friday nights, from January to March. However, in April, the business has a big promotion for beer every Saturday. Therefore, many customers would buy burger on Saturday instead of Friday because of this promotion[8].

In this case, it would be desirable if the pattern can still be recognized when the disturbance is within some reasonable threshold. Therefore, in , Yang et al. extended the idea to find asynchronous periodic patterns. Yang et al.’s asynchronous periodic pattern problem aims at mining the longest periodic subsequence which may contain a disturbance of length up to a certain threshold. Formally, a valid subsequence with respect to a pattern P in a sequence D is a set of non overlapping valid segments, where a valid segment has at least min rep contiguous matches of P and the distance between any two successive valid segments does not exceed a parameter max dis. A valid subsequence with the most overall repetitions of P is called its longest valid subsequence. However, this model has some problems. . First, this model only focused on mining periodic patterns in temporal sequences of events. However, in real-world applications, we may find multiple events at one time slot in terms of various intervals (e.g., hour, day, week, etc.) as discussed in previous works. We refer to such databases as sequences of event sets. . Second, this model only focused on mining the longest sequence of a pattern, which can only capture part of the system’s behavior. For example, in the case when two successive, non overlapped segments with a disturbance larger than max dis, only the larger segment will be reported[7].

To address these problems, K. Huang proposed a novel SMCA algorithm which requires no candidate pattern generation as compared to previous technique. Their algorithm allows the mining of all asynchronous periodic patterns, not only in a sequence of events, but also in a temporal dataset with multiple event sets. They also proposed a dynamic hash-based validation mechanism which discovers all asynchronous type periodic patterns in a single scan of temporal dataset. Their four phase approach uses a sequence of algorithms to mine singular pattern, multiple pattern, maximal complex pattern and finally asynchronous periodic patterns. Each of these algorithms uses output of the last executed algorithm as their input. The main limitation of their algorithm is that, it not only mines the maximal complex pattern but also its subsets (single event patterns and multiple events patterns) using depth first search enumeration approach, thus wasting a considerable amount of processing time for mining

subsets. For large datasets having *i-patterns* where *i* is too large, a large amount of processing time is waste for mining singular and multi events *1-patterns* that are subsets of *i-patterns*. To increase the efficiency of mining asynchronous periodic patterns on large datasets, we propose a novel efficient algorithm E-MAP. Our propose algorithm finds all maximal complex patterns in a single step algorithm using a single dataset scan without mining single event and multiple events patterns explicitly, while asynchronous periodic patterns are mined using the same depth first search enumeration process as described in . The single dataset scan and single step mining approach makes the E-MAP much faster and efficient as compared to previous technique SMCA. The other feature of E-MAP is that, it requires less storage space as compared to SMCA. To check the effectiveness of our E-MAP approach, we also provide detailed experimental results on real and artificial datasets. Our different experimental results suggest that mining asynchronous periodic patterns using E-MAP is more efficient as compared to SMCA.

Database mining is motivated by the decision support problem faced by most large retail organizations. Development of bar-code technology has made able retail organizations to collect and store massive amounts of sales data, referred to as the basket data. A record in such data typically consists of the transaction date and the items bought in the transaction [1]. Very often, data records also contain customer-id, particularly when the purchase has been made using a credit card or a frequent-buyer card. Catalog companies also collect such data using the orders they receive.

A sequence database consists of sequences of ordered elements or events, recorded with or without a concrete notion of time [1]. There are many applications involving sequence data. Typical examples include customer shopping sequences, Web click streams, biological sequences, sequences of events in science and engineering, and in natural and social developments.

Sequential Pattern Mining

Sequence Pattern Mining is the mining of frequently occurring ordered events or subsequences as patterns [9]. An example of sequential pattern is “Customers who buy a canon digital camera are likely to buy an HP color printer within a month”[1]. For retail data, sequential patterns are useful for shelf placement and promotions. Also telecommunications and other businesses may also use sequential patterns for targeted marketing, customer retention and many other tasks. Other areas in which sequential patterns can be applied include Web access pattern analysis, weather prediction, production processes, and network intrusion detection analysis. Most studies of sequential pattern mining concentrate on categorical patterns [6]. The sequential pattern mining problem was first introduced by Agrawal and Srikant in 1995[1] based on their study of customer purchase sequences, as follows: ”Given a set of sequences, where each sequence consist of a list of events(or element) and each event consists of set of items, and given a user specified minimum support threshold of min_sup, sequential pattern mining finds all the frequent subsequences, that is, the subsequences whose occurrence frequency in the set of sequences is no less than min_sup.”

Definition 1: Let $X = \{x_1, x_2, x_{3,..} x_n\}$ be a set of different items. An element *e*, denoted by $< x_1, x_{2, ...} >$, is a subset of items belonging to X which appear at the same time. A sequence *s*, denoted by $< e_1; e_2; \ldots ; e_m >$, is an ordered list of elements. A sequence database *Db* contains a set of sequences, and | *Db* | represents the number of sequences. in Db. A sequence $\alpha = < a_1; a_2; \ldots ; a_n >$ is a subsequence of another sequence $\beta =< b_1; b_2; \ldots ; b_m >$ if there exist a set of integers, $1 \leq i_1 \leq i_2 \leq i_n \leq m$, such that a_1 is a subset of b_{i1}; a_2 is a subset of b_{i2} ; . . . and *an* is a subset of b_{in} . The sequential pattern mining can be defined as “Given a sequence database *Db* and a user-defined minimum support min sup, find the complete set of subsequences whose occurrence frequencies ≥ min sup *| *Db* |” .

II. RELATED WORK

As mentioned, the sequential pattern mining with a static database and with an incremental database is two special cases of the progressive sequential pattern mining. In the following, we introduce the previous works on the static sequential pattern mining, the incremental sequential pattern mining, and the progressive sequential pattern mining. Previous researchers have developed various methods to find frequent sequential patterns with a static database.

The assumption of having a static database may not hold in many applications. The data in real world usually change on the fly. When we deal with an incremental database, it is not feasible to remine the whole sequential patterns every time when the database increases because the remaining process is costly. To handle the incremental database, Parthasarathy et al. presented the algorithm ISM [2] using a lattice framework to incrementally update the support of each sequential pattern in equivalent classes. Masseglia et al. derived the algorithm ISE [3] to join candidate sequential patterns in original database with the newly increasing database. Cheng et al. introduced algorithm IncSpan [4], which utilized a special data structure named sequential pattern tree to store the projection of database. However, the incremental mining algorithms can only handle the incremental parts of the database. Because of the limitation of data structures maintained in their algorithms, they can only create new candidates but cannot delete the obsolete data in a progressive database. The deletion of an item from the database results in the reconstruction of all candidate item sets, which induces incredible amount of computing.

III. PROBLEM STATEMENT

Design & Develop an Automatic Data Mining System for Asynchronous Periodic Patterns. The proposed system will be based on Modified version of E-MAP algorithm compatible for temporal dataset. The entries for temporal dataset will be taken from a real life database of a grocery shop.
Let E be a set of all events. An event set is a nonempty subset of E. A temporal dataset D is a set of records where each record is a tuple in the form < tid, X > for time instant tid and event set X.

Table 1: A temporal data with 4 events (A, B, C, D) and 7 time instances

Time instance	Event
1	A,C
2	A,B
3	A,C
4	A,B,D
5	D
6	C
7	C

III. FORMAT OF DATABASE

Table 2 : Database

Seq_no	Element	Time_stamp
1	Magi	1
1	Soup	2
1	Soup	4
2	Burger	2
2	Icetea	3
2	Magi	4
2	Icetea	5
3	Soup	1
3	Icetea	2
3	Magi	3
3	Icetea	5
4	Berger	3
4	Icetea	4
4	Burger	5
4	Magi	4
5	Soup	2
5	Burger	5

IV. IMPLEMENTATION

This system tries to find segment of subsequence as a valid pattern depending upon the minimum repetition and threshold value given to it. It provides the graphical user interface to input these values.

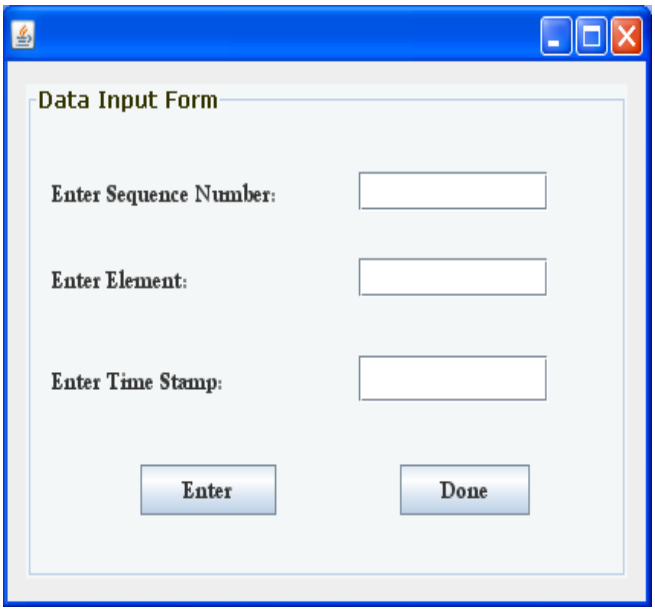

Fig 1 : GUI (Input data form)

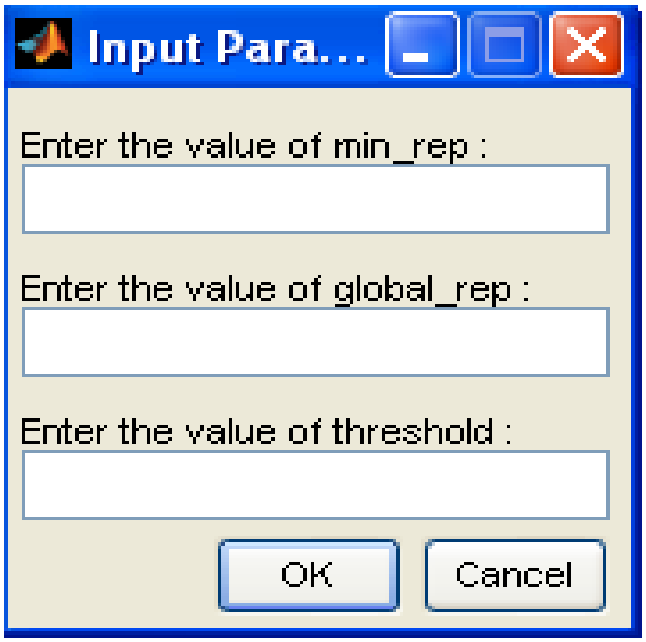

Fig 2 : GUI (Input parameters)

V. RESULTS & DISCUSSION

After implementation of the algorithm, we have obtained the result.
We tried to compare its performance with other mentioned algorithms on the basis of Space and time complexity .

Let $SP_{SMCA}(n)$: Space Complexity of SMCA
$SP_{EMAP}(n)$: Space Complexity of EMAP
$SP_{MEMAP}(n)$: Space Complexity of MEMAP

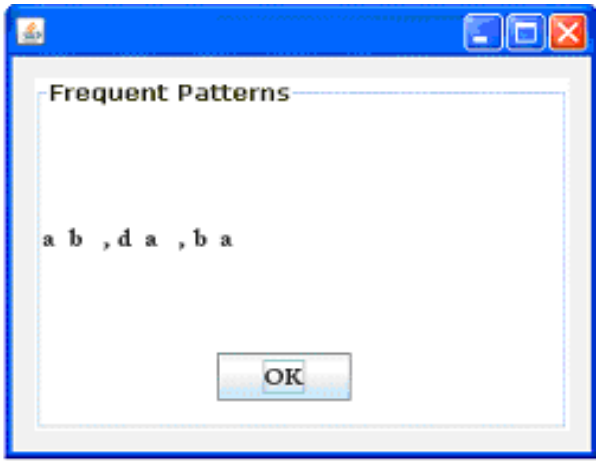

Fig 3 : Frequent patterns

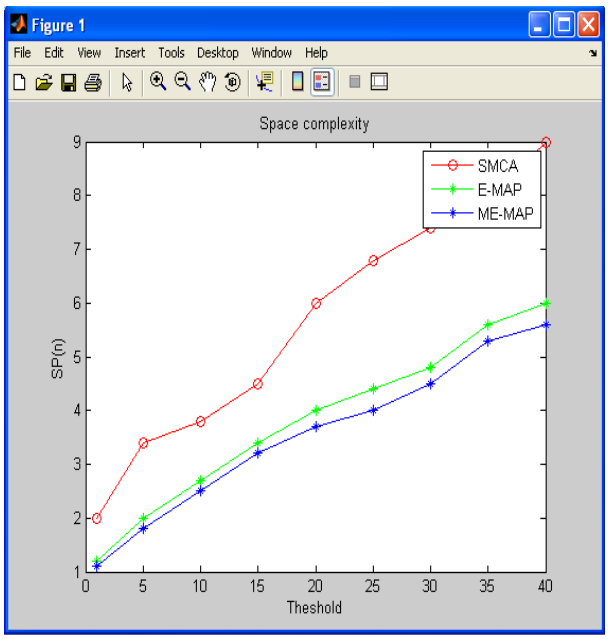

Fig 3 : Space Complexity

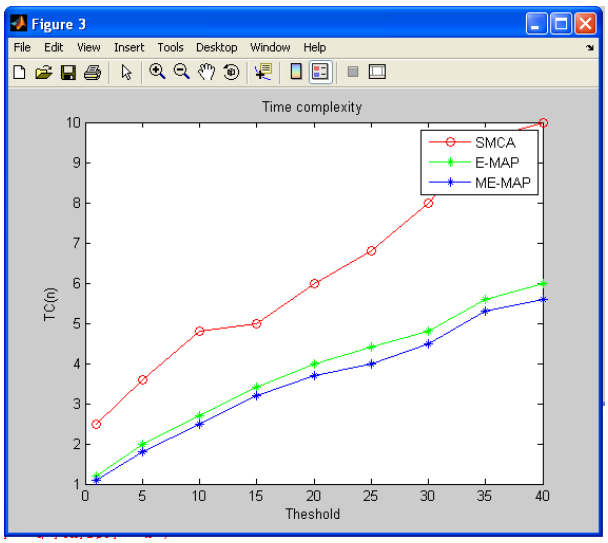

Fig 4 : Time Complexity

VI. CONCLUSION

In this paper we proposed a new algorithm ME-MAP for sequential pattern mining of asynchronous periodic patterns. We have studied the performances of SMCA, E-MAP and Modified E-MAP algorithms. The performance evaluation is done on the basis of time and space complexities of these algorithms. It is found that :

$SP_{SMCA}(n) \leq SP_{E\text{-}MAP}(n) \leq SP_{ME\text{-}MAP}(n)$

$TC_{SMCA}(n) \leq TC_{E\text{-}MAP}(n) \leq TC_{ME\text{-}MAP}(n)$

This shows the effectiveness of our approach.

VII. REFERENCES

[1] Agrawal, R. and Srikant, R. Mining sequential patterns. In Eleventh International Conference on Data Engineering, P. S. Yu and A. S. P. Chen, Eds. IEEE Computer Society Press, Taipei, Taiwan, pp. 3-14, 1995.

[2] S. Parthasarathy, M.J. Zaki, M. Ogihara, and S. Dwarkadas, "Incremental and Interactive Sequence Mining," Proc. 8th ACM Int'l Conf. Information and Knowledge Management (CIKM '99), pp. 251-258, 1999.

[3] F. Masseglia, P. Poncelet, and M. Teisseire, "Incremental Mining of Sequential Patterns in Large Databases," Data and Knowledge Eng., vol. 46, pp. 97-121, 2003.

[4] H. Cheng, X. Yan, and J. Han, "INCSPAN: Incremental Mining of Sequential Patterns in Large Database," Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '04), pp. 527- 532, 2004.

[5] A. Balachandran, G.M. Voelker, P. Bahl, and P.V. Rangan, "Characterizing User Behavior and Network Performance in a Public Wireless LAN," Proc. ACM SIGMETRICS Int'l Conf. Measurement and Modeling of Computer Systems (SIGMETRICS '02), pp.195-205, June 2002.

[6] Jen-Wei Huang, Chi-Yao Tseng, Jian-Chih Ou, and Ming-Syan Chen. "A General Model for Sequential Pattern Mining with a Progressive Database", Knowledge And Data Engineering, vol. 20, no. 9, pp. 1153-1167, September 2008.

Thodeti Srikanth received his Master of Computer Applications degree from Kakatiya University, Andhra Pradesh, INDIA in 2004. He is pursuing Ph.D. (Computer Science) from Dravidian University, Andhra Pradesh, INDIA.