

# A survey on Temporal High Utility Itemsets on data streams

Thalla Shankar

Assistant Professor, Department of CSE,  
Kshatriya College of Engineering, Armoor  
Nizamabad, Andhra Pradesh, India, 503224  
[shankar.thalla@yahoo.com](mailto:shankar.thalla@yahoo.com)

Sadak Murali

Assistant Professor, Department of CSE,  
Kshatriya College of Engineering, Armoor  
Nizamabad, Andhra Pradesh, India, 503224  
[muralisadak@gmail.com](mailto:muralisadak@gmail.com)

**Abstract--** An important research issue extended from the association rules mining is the discovery of temporal association patterns in data streams due to the wide applications on various domains. The definition of temporal data mining can be said as the activity of looking for interesting correlations or patterns in large sets of temporal data accumulated for other purposes. For time variant data streams to develop an efficient and effective method to mine various temporal patterns there is a strong demand. However, most methods designed for the traditional databases cannot be directly applied for mining temporal patterns in data streams because of the high complexity. However, these techniques cannot discover all the possible relations and thus there is need for a different approach that will provide a deeper kind of analysis. We know that temporal high utility item sets are the itemsets with support larger than a pre-specified threshold in current time window of DataStream. Discover of temporal high utility item sets is an important process of mining interesting patterns like association rules from data streams. For temporal association mining with utility approach the novel algorithm is proposed. The temporal high utility itemset which can generate less candidate item sets can be found. Although there existed numerous studies on high utility itemsets mining and data stream analysis as described above, there is no algorithm proposed for finding temporal high utility itemsets in data streams.

**Keywords--** Data mining, utility mining, temporal data, temporal high utility itemsets, data streams.

## I. INTRODUCTION

*Data mining* refers to extracting or mining knowledge from large amounts of data. Thus, data mining should have been more appropriately named “knowledge mining from data”. Finding of frequent patterns task in data mining in large databases is very important use full in many applications over the past few years. The primary goal is to discover unexpected trends, hidden patterns in the data. Data mining is concerned with analysis of large volumes of data to automatically discover interesting regularities or relationships which in turn leads to better understanding of the underlying processes. Data mining activities uses combination of techniques from database artificial

intelligence, statistics, technologies machine learning. This includes bioinformatics, genetics, medicine, clinical research, education, retail and marketing research.

*Utility mining* is one of the most challenging data mining tasks is the mining of high utility itemsets efficiently. Identification of the itemsets with high utilities is called as Utility Mining. The utility can be measured as per the user preferences utility can be measured in terms of cost, profit or other expressions. The limitations of frequent or rare itemset mining motivated researchers to conceive a utility based mining approach, which allows a user to conveniently express his or her perspectives concerning the usefulness of itemsets as utility values and then find itemsets with high utility values higher than a threshold. In utility based mining the term utility refers to the quantitative representation of user preference i.e. according to an itemsets utility value is the measurement of the importance of that itemset in the user’s perspective.

*Mining high utility itemsets* from databases refers to finding the itemsets with high profits. Here, the itemset utility meaning means profitability of an item to users or importance, interestingness. High utility itemsets mining has become one of the most interesting data mining tasks with broad applications and it identifies itemsets whose utility satisfies a given threshold. By using different values it allows users to quantify the usefulness or preferences of items using different values. A high utility itemset is defined as: A group of items in a transaction database is called itemset. In a transaction database consists of two aspects in this itemset: First one is itemset in a single transaction is called internal utility and second one is itemset in different transaction database is called external utility.

*Temporal data mining* is to discover hidden relations between sequences and subsequences of events. The techniques and algorithms are applied on data collected over time is a research field of growing interest. The activity of looking for interesting correlations or patterns in large sets of temporal data accumulated for other purposes is the definition of temporal data mining. To discover hidden

relations between sequences and subsequences of events is the ultimate goal of temporal data mining. There are mainly three steps involved in the discovery of relations between sequences of events: the representation and modeling of the data sequence in a suitable form; the definition of similarity measures between sequences; and the application of models and representations to the actual mining. In the process of Knowledge Discovery in Temporal Databases that enumerates structures (temporal patterns or models) over the temporal data temporal data mining is a single step. Temporal data mining tasks are examples of clustering of time series and classification, trends or discovery of temporal patterns in the data, similarity-based time series retrieval, association of events over time, segmentation and time series indexing. Temporal data mining could indeed play an essential role in the stock market domain.

*Temporal high utility itemsets* for mining temporal high utility itemsets from data streams efficiently and effectively. The itemsets with support larger than a pre-specified threshold in current time window of data stream are the temporal high utility itemsets. It can discover temporal high utility itemsets by generating fewer candidate itemsets such that the execution time can be reduced substantially in mining all high utility itemsets in data streams from the data streams efficiently and effectively. The process of discovering all temporal high utility itemsets under all time windows of data streams can be achieved in this way effectively with less memory space and execution time. By generating less candidate itemsets discover the temporal high utility itemsets with higher performance.

## II. LITERATURE SURVEY

The rationale behind mining frequent itemsets is that only itemsets with high frequency are of interest to users. By the significance of the discovered itemsets, however the practical usefulness of frequent itemsets is limited. The statistical correlation between items is only reflected by the frequent itemset but not reflect the semantic significance of the items. In this paper, we discussed earlier researches different algorithms and method.

R. Agrawal et al in [2] proposed Apriori algorithm, it is used to obtain frequent itemsets from the database. One of the most common approaches to mining frequent patterns is the Apriori method and when a transactional database represented as a set of sequences of transactions performed by one entity is used, temporal sequences manipulation. In mining the association rules we have the problem to generate all association rules that have support and confidence greater than the user specified minimum support and minimum confidence respectively. The first pass of the

algorithm simply counts item occurrences to determine the large 1-itemsets. First it generates the candidate sequences and then it chooses the large sequences from the candidate ones. Next, the database is scanned and the support of candidates is counted. The second step involves generating association rules from frequent itemsets. Apriori is a classic algorithm for frequent itemset mining and association rule learning over transactional databases. After identifying the large itemsets, only those itemsets are allowed which have the support greater than the minimum support allowed. The frequent item sets determined by Apriori can be used to determine association rules which highlight general trends in the database. Apriori Algorithm generates lot of candidate item sets and scans database every time. When a new transaction is added to the database then it should rescan the entire database again.

Liu et al in [9] proposed a Two-phase algorithm for finding high utility itemsets. The utility mining is to identify high utility itemsets that drive a large portion of the total utility. Utility mining is to find all the itemsets whose utility values are beyond a user specified threshold. Two-Phase algorithm, it efficiently prunes down the number of candidates and obtains the complete set of high utility itemsets. We explain transaction weighted utilization in Phase I, only the combinations of high transaction weighted utilization itemsets are added into the candidate set at each level during the level-wise search. In phase II, only one extra database scan is performed to filter the overestimated itemsets. Two-phase requires fewer database scans, less memory space and less computational cost. It performs very efficiently in terms of speed and memory cost both on synthetic and real databases, even on large databases. In Two-phase, it is just only focused on traditional databases and is not suited for data streams. Two-phase was not proposed for finding temporal high utility itemsets in data streams. However, this must rescan the whole database when added new transactions from data streams.

V.S. Tseng et al in [13] proposed a novel method THUI (Temporal High Utility Itemsets)-Mine for mining temporal high utility itemset mining. The temporal high utility itemsets are effectively identified by the novel contribution of THUI-Mine by generating fewer temporal high transaction weighted utilization 2-itemsets such that the time of the execution will be reduced substantially in mining all high utility itemsets in data streams. To generate a progressive set of itemsets THUI-Mine employs a filtering threshold in each partition. The temporal mining problem can be decomposed into two procedures: 1. Preprocessing procedure: This procedure deals with mining on the original transaction database. 2. Incremental procedure: The procedure deals with the update of the high utility itemsets from data streams. In this way, discovering the process of all

temporal high utility itemsets under all time windows of data streams can be achieved effectively. The temporal high utility itemsets with less candidate itemsets and higher performance can be discovered by THUI- mine. From these candidate *k*-itemsets to find a set of high utility itemsets finally, it needs one more scan over the database. Huge memory requirement and lot of false candidate itemsets are the two problems of THUI- Mine algorithm.

Tarek F et al in [12] proposed a novel method ITARM (Incremental Temporal Association Rules Mining) for updating temporal association rules in the transaction database. To deal with changing data temporal data mining has become a core technical data processing technique. The discovered rules need to be updated so that the Temporal databases are continually updated; by discarding the rules that become insignificant and including new valid ones previously discovered rules have to be maintained. In order to solve the problem of handling time series by including time expressions into association rules Temporal association rule (TAR) has been introduced. To maintain temporal frequent itemsets after the temporal transaction database has been updated Incremental Temporal Association Rules Mining (ITARM) is used. ITARM deals with objects in the case of numerical attributes. Each object has a set of numerical and a unique ID. The algorithm employs the skeleton of the incremental procedure of the Sliding-Window Filtering algorithm (SWF). The time needed for generating new candidates is reduced by the proposed algorithm by storing candidate 2-itemsets) It presents a technique to update the previously generated candidates instead of re-generating them from scratch. Re-running the temporal mining algorithm every time is ineffective since it neglects the previously discovered rules and incremental mining techniques cannot deal with temporal association rules are the problems of ITRAM.

Bai-En S et al [11] proposed a novel algorithm *GUIDE* (Generation of temporal maximal Utility Itemsets from Data streams) which can we find efficiently mining temporal maximal utility itemsets from data streams. They proposed a novel data structure *TMUI-tree* (Temporal Maximal Utility Itemset tree), *is also proposed for efficiently capturing* the utility of each itemset with one-time scanning and for storing information in the processes of mining utility patterns from data streams. To pick up the essential information about the utilities of the appeared itemsets and store them into a compact data structure, namely *TMUI-tree* effectively is the basic idea of *GUIDE*. To mining temporal maximal utility itemsets from data streams *GUIDE* is the first one-pass utility-based algorithm.

There is no need to generate candidate itemsets during the mining processes in the *GUIDE*. *TMUI tree* can help *GUIDE* finding *TMUIs* efficiently and it is easy to maintain. *TMUI tree*, facilitates *GUIDE* for finding temporal maximal utility itemsets efficiently and it is easy to maintain for storing essential information in the mining processes. *TMUI-tree* is storing the utilities and the potential *TMUIs* in a new transaction. Thereafter, the actual *TMUIs* are found and generated in the data structure simultaneously. In *GUIDE*, lot of maximal utility itemsets and in *TMUI-tree* it is hard to maintain the utility patterns.

Authors	Algorithm	Features	Problem
R.Agrawal et. al	Apriori	Frequent itemsets and candidate generation	Rescan database every time and lot of candidates generated.
Liu et. al	Two-Phase	Finds High utility itemset in traditional database and generates less candidate	No temporal itemsets found and rescan database every time
Tseng et. al	THUI-Mine	Finds temporal data in data stream and generates few candidate	Lot of false candidate are found and huge memory
Tarek F et al	ITARM	Temporal frequent itemsets	Re-running and neglects the previously discovered
Bai-En S et al	<i>GUIDE</i> and <i>TMUI-tree</i>	Temporal maximal utility itemsets and storing utility patterns from data streams	Lot of maximal utility itemsets and hard to maintain the utility patterns

### III. CONCLUSION

This paper presents a survey on various Temporal High Utility Itemsets algorithms that were proposed by earlier researches for the better development in the field of Data Mining. Various algorithms and methods discussed above will help in developing Temporal High utility itemsets for data stream. In the future scope, we will be presenting a comparative study of various algorithms for temporal mining high utility itemset.

### REFERENCES

- [1] R. Agrawal , T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", in *proceedings of the ACM SIGMOD International Conference on Management of data*, pp. 207-216, 1993
- [2] R. Agrawal and R. Srikant. "Fast algorithms for mining association rules," in *proceedings of the 20th VLDB Conf.*, pp. 487-499, 1994.
- [3] Ayn, N.F., Tansel, A.U., and Arun, E. "An efficient algorithm to update large itemsets with early pruning" *proceedings of the Fifth ACM*

- SIGKDD International conference on Knowledge Discovery and Data Mining in Diego*, August 1999.
- [4] Bettini, C., Wang, X.S., Jajodia, S., 1996. Testing complex temporal relationships involving multiple granularities and its application to data mining. In: Proceedings of the 15th ACM Symposium on Principles of Database Systems, Montreal, Canada, 1996, pp. 68–78.
- [5] R. Chan, Q. Yang and Y. Shen. “Mining high utility itemsets,” in *Proc. of Third IEEE Int’l Conf. on Data Mining*, pp. 19-26, Nov., 2003.
- [6] Hui Ning, Haifeng Yuan, Shugang Chen, “Temporal association rules in mining method”, in *Proceedings of the 1st International Multi-Symposiums on Computer and Computational Sciences (IMSCCS’06)*, Zhejiang, China, vol. 2, June 2006, pp. 739–742.
- [7] Chang-Hung Lee, Ming-Syan Chen, Cheng-Ru Lin, “Progressive partition miner: an efficient algorithm for mining general temporal association rules”, *In the IEEE Transaction on Knowledge and Data Engineering* 15 (4) (July/August 2003) 1004–1017.
- [8] Chi, Y., Wang, H., Yu, P. S., and Richard, R. Muntz. “Moment: Maintaining Closed Frequent Itemsets over a Stream Sliding Window”, in *Proceedings of IEEE International Conference on Data Mining (ICDM’04)*, 2004
- [9] Y. Liu, W. Liao and A. Choudhary, “A fast high utility itemsets mining algorithm”, in *proceedings. of the Utility-Based Data Mining Workshop*, 2005.
- [10] Claudia m M. Antunes and Arlindo L. Oliveira, “Temporal Data Mining: an overview”. *Lecture Notes in Computer Science*.
- [11] Bai-En. S, V. S. Tseng and P. S. Yu, “Online mining of temporal maximal utility itemsets from data streams,” in *Proc. of the 25th Annual ACM Symposium on Applied Computing*, Switzerland, Mar., 2010.
- [12] Tarek F. Gharib, Hamed Nassar, Mohamed Taha, Ajith Abraham. “An efficient algorithm for incremental mining of temporal association rules”, in *proceedings*
- [13] V. S. Tseng, C. J. Chu and T. Liang, “Efficient Mining of Temporal High Utility Itemsets from Data streams,” in *Proc. of ACM KDD Workshop on Utility-Based Data Mining Workshop (UBDM’06)*, USA, Aug., 2006.