A Recent Review on Itemset Tree Mining: MEIT Technique

Tanvi P. Patel
Department of Computer Science and Engineering
Parul Institute of Technology
Waghodia, Vadodara – 391760, India

Warish D. Patel
Department of Computer Science and Engineering
Parul Institute of Technology
Waghodia, Vadodara – 391760, India

ABSTRACT
Association rule mining forms the core of data mining and it is termed as one of the well-researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Hence, Association rule mining is imperative to mine patterns and then generate rules from these obtained patterns. This paper provides the preliminaries of basic concepts about Itemset mining and survey the list of existing tree structure algorithms. These algorithms include various tasks such as fast query processing, optimizing memory space and reducing tree construction time. For mining maximal frequent pattern various algorithms used which optimization the search space for pruning.

Keywords
Association rule mining, Itemset mining, Itemset tree, MEIT, maximal frequent pattern

1. INTRODUCTION
Association rule mining, one of the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. Rule support and confidence are two measures of rule interestingness.[24]

$$support(A \Rightarrow B) = P(A \cup B) \quad (24)$$

$$confidence(A \Rightarrow B) = P(A|B) \quad (24)$$

$$confidence(A \Rightarrow B) = P(A|B) \quad (24)$$

$$= \frac{support\ (A \cup B)}{support\ (A)}$$

$$= \frac{support\_count\ (A \cup B)}{support\_count\ (A)}$$

In general, association rule mining can be viewed as a two-step process[24]:

i. Find all frequent item-sets, each one of these item-sets will occur at least as frequently as a determined minimum support count, min sup.

ii. Generate strong association rules from the frequent item-sets; these rules must satisfy minimum support and minimum confidence.

Frequent pattern mining searches for recurring relationships in a given data set. Extracting all probable association rules from a database, on the other hand, is a computationally intractable problem, in consequence of the operation of explosion in the number of sets of attributes for which frequency counts must be computed, it may involve multiple passes of the database. So by using a single database pass to execute an incomplete computation of the totals mandatory, storing all these in the form of tree.

Here to recover the requirement of an incremental data mining approach relay on data structure called the itemset tree. This approach is efficient for solving problems related to efficiency of managing data updates, accurateness of data mining results, handling input transactions, and responding user queries. There are several capable algorithms to insert transactions into the item-set tree and to count frequencies of item-sets for queries about power of association among items.[2]

One of the efficient tree structures is memory efficient itemset tree. An efficient data structure for performing targeted queries for item-set mining and association rule mining is the MEIT. During transaction insertion, it employs an effective node compression mechanism for reducing the size of tree nodes. Furthermore, during transaction insertion or query processing, it relies on an on-the-fly node decompression mechanism for restoring node content.[1]

In this paper we aim to minimize number of patterns generated, which leads us minimizing computation time and main memory consumption. This may result in less number of rule generations. This tree structure will give fast traversal as well as fast access for generating rules. So mining time also will be reduced and main memory overhead also gets reduced.

2. RELATED WORK
R. Agrawal, T. Imielinski and A. Swami proposed[4] technique for Mining Association Rules between Sets of Items in Large Databases. In this, all significant association rules generated between items in the Database. It is reducing the number of item sets by generating, closed, maximal, optimal item sets. Several algorithms to reduce the number of rules using, (1)Non-redundant rules (2)Pruning techniques. The Demerits of system are Usefulness of association rules is strongly, limited by the huge amount of delivered rules, It is crucial to help the decision-maker with an efficient technique for reducing the number of rules. The Merits of System are Reduce the number of item sets by generating closed, maximal optimal item sets, and several algorithms to reduce the number of rules, using non-redundant rules, and pruning techniques.

Fournier-Viger, P., Wu, C.-W., Tseng, V.S. introduce[5] Mining Top-K Association Rules. They proposed an algorithm to mine the top-k association rules, where k is the number of association rules to be found and is set by the user. Top-K Rules takes as input a transaction database, a number k of rules that the user wants to discover and the minconf

threshold. It sets an internal minsup variable to 0. Then, it starts searching for rules. As soon as a rule is found, it is added to a list of rules L ordered by the support. The algorithm continues searching for more rules until no rule are found, which means that it has found the top-k rules. It is to mine the top-k rules with the highest support that meet a desired confidence. Here rule expansions method is used. They expanding rules in Top-K Rules left expansion and right expansion.

P. Fournier-Viger, V.S. Tseng introduces[6] Mining Top-K Non-Redundant Association Rules. In this there is an approximate algorithm named TNR using for mining the top-k non-redundant association rules. The algorithm is said to be approximate because it is guaranteed to find non-redundant rules. But the rules set up may not be the top-k non redundant rules. It was derived from the approach for generating association rules that is known as "rule expansions", and includes strategies to ignore generating redundant rules. An evaluation of the TNR has excellent performance and scalability.

S. Dandu, B.L. Deekshatulu proposed[7] Improved Algorithm for Frequent Item sets Mining Based on Apriori and FP-Tree. In this, they introduce APFT (combination of apriori and FP-tree). It includes correlated items & trims the non-correlated Item-sets. The advantage of APFT is that it doesn't generate conditional & sub conditional patterns of the tree recursively and the results of the experiment show that it works fasts than apriori and almost as fast as FP-growth. They have proposed to go one step further & modify the APFT to include correlated items & trim the non-correlated itemsets. This additional feature optimizes the FP-tree & removes loosely associated items from the frequent itemsets.

C.K. Leung, Q.I. Khan, Z. Li, T. Hoque introduces[8] CanTree which is a canonical-order tree for incremental frequent-pattern mining. It takes the content of the transaction database and arranging tree nodes with reference towards some canonical order, which can be determined by the user prior to the mining process or at runtime during the mining process. Hence, the building of the tree has need of only one database scan. It significantly reduces computation and time, because they easily find join paths and require only upward path traversals. It provides users with efficient incremental mining. CanTrees can be used for (i) constrained mining, (ii) incremental constrained mining, (iii) interactive mining, (iv) incremental interactive mining.

D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, T. Yiu introduces MAFIA: A Maximal Frequent Itemset Algorithm [9], it is used for mining maximal frequent itemset from a transactional database; MAFIA uses a vertical bitmap representation for the transactional database. MAFIA performs best on dense datasets where large subtrees can be removed from the search space. It integrates a depth-first traversal of the itemset lattice with effective pruning mechanisms which increase performance. MAFIA is highly optimized for mining long itemsets on dense data. It includes search space pruning techniques and adaptive compression. Mafia uses two pruning strategies to remove non-maximal sets. The first is the look-ahead pruning first used in MaxMiner. The second is to check if a new set is subsumed by an existing maximalset.

K. Gouda, M.J. Zaki introduces[10] GenMax. It is an efficient Algorithm for Mining Maximal Frequent Itemsets which uses backtrack search based algorithm. It uses a novel technique named "progressive focusing" for checking maximality. It is widely used for mining the exact set of maximal patterns efficiently and to exclude non-maximal itemsets, and uses "diffset propagation" for fast frequency checking. (i) Superset checking optimization, (ii) Frequency testing optimization, (iii) Diffsets propagation. Mafia[9] mines a superset of the MFI, and requires a post- pruning step to eliminate non-maximal patterns. In contrast GenMax integrates pruning with mining and returns the exact MFI.

A. Hafez, J. Deogun, V. V. Raghavan proposed[3] the A Item-Set Tree: Data Structure for Data Mining. It is complete transaction lattice. Each node on the lattice represents a possible large item-set. A count is attached to each node to reflect the frequency of item-sets. It is an enhancement in data capturing technology which leads to exponential growth in amounts of data being stored in information systems. This growth in turn has forced researchers to seek new techniques for extraction of knowledge implicit or hidden in the data.

M. Kubat, A. Hafez, V.V. Raghavan, J.R. Lekkala, W.K. Chen introduce[2] concept of item-set Trees for Targeted Association Querying. In this paper we can have faster detection of item-sets and association rules. Here it establishes support, search for items, and generation of rules. Here it includes the concept of an item-set tree and presents an algorithm that generates data structure from a set of market baskets. It explains how to use it to process three different query types then reports experiments illustrate the technique's performance in terms of the amount of computation needed to response a query. It also investigates the costs of building an item-set tree from data. It establishes the existence of a one-to-one mapping between the space of item-set trees and the space of market basket databases. This mapping guarantees that the resulting item-set tree does not depend on the order in which the market baskets have been presented. As shown in figure 1, itemset tree can constructing by following.

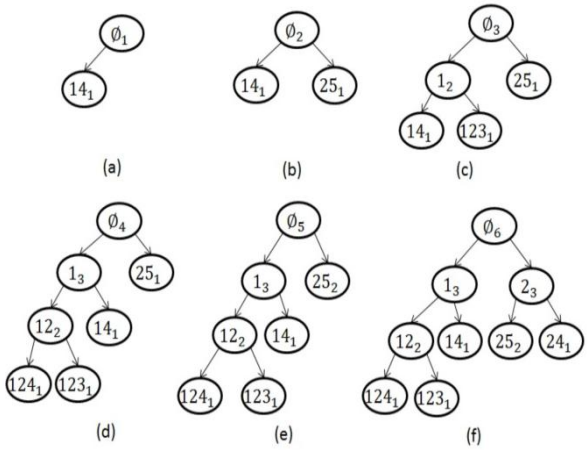

Fig 1: Itemset Tree Construction[2]

P. Fournier-Viger, E. Mwamikazi, T. Gueniche, U. Faghihi proposed[1] Memory Efficient Item-set Tree (MEIT) for Targeted Association Rule Mining. It is incrementally updatable by putting in new transactions. An effective node compression mechanism is used for reducing the size of tree nodes. An on-the-fly node decompression mechanism is used for restoring node content. They have designed the MEIT based on three observations that are formalized by the following three properties[1].

i. In an IT, transactions are insert by traversing branches in a top-to-bottom manner. It inserted both, by creating

new node or by incrementing the support of an existing node.

ii. Queries on an IT are always processed by traversing tree branches in a top-to-bottom manner.

iii. Let k be an IT node and parent(k) be its parent. The relationship $i(parent) \subset i(k)$ holds between k and its parents. More generally, this property is transitive. Therefore, it can be said that for any ancestor x of k, $i(x) \subset i(k)$.

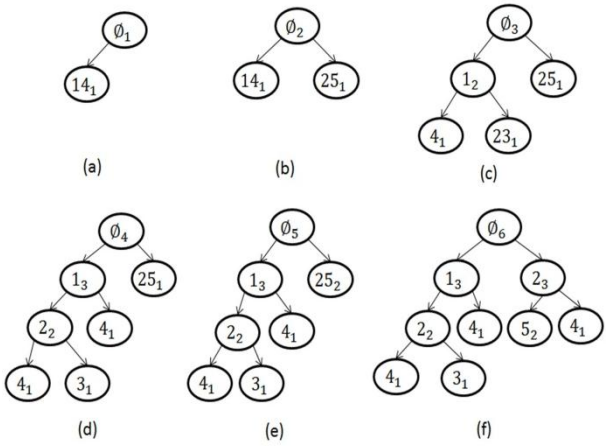

Fig 2: Memory Efficient Itemset Tree Construction[1]

Fig 2 shows the construction of memory efficient itemset tree. From both Fig 1 and Fig 2 we can conclude that, IT is larger in amount of memory up to 60% than MEIT. But, MEIT takes almost twofold time for tree construction than IT.[1]

3. CONCLUSION

From the literature survey it is concluded that itemset mining gives idea for improved data structure for Itemset tree construction. MEIT approach used for incremental mining, consumes less memory as compared to the traditional tree data structures. Here, focus is led on generation of an itemset tree structure in a more efficient manner, to perform rapid association rule mining and frequent itemset mining. Further, to obtain maximal frequent items, unnecessary items which are less frequent are pruned out. As a result, mining time as well as memory consumption is reduced.

4. REFERENCES

[1] P. Fournier-Viger, E. Mwamikazi, T. Gueniche, U. Faghihi, "MEIT: Memory Efficient Item-set Tree for Targeted Association Rule Mining", 9th International Conference, ADMA 2013, Part II, Volume 8347 - Springer, Heidelberg, pp. 95-106, 2013.

[2] M. Kubat, A. Hafez, V.V. Raghavan, J.R. Lekkala, W.K. Chen, "Item-set trees for targeted association querying", Knowledge and Data Engineering, IEEE Transactions on Volume 15(6), pp.1522 – 1534, 2003.

[3] A. Hafez, J. Deogun, V. V. Raghavan, "The Item-Set Tree: A Data Structure for Data Mining", First International Conference, DaWaK'99 Florence, Italy, August 30 – September - Springer, pp.183-192, 1999.

[4] R. Agrawal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM - SIGMOD, pp. 207-216, 1993.

[5] P. Fournier-Viger, C.W. Wu, V.S. Tseng, "Mining Top-K Association Rules", L.Kosseim, D. Inkpen, Canadian AI 2012. LNCS, vol. 7310, Springer- Heidelberg, pp. 61–73, 2012.

[6] P. Fournier-Viger, V.S. Tseng, "Mining Top-K Non-Redundant Association Rules" Chen, L., Felfernig, A., Liu, J., Ras, Z.W. (eds.) ISMIS 2012. LNCS, vol. 7661 - Springer, Heidelberg, Pp. 31–40, 2012.

[7] S. Dandu, B.L. Deekshatulu, "Improved Algorithm for Frequent Item sets Mining Based on Apriori and FP-Tree", Global Journal of Computer Science and Technology, Global Journal of Computer Science and Technology, Volume 13, 2013.

[8] C.K. Leung, Q.I. Khan, Z. Li, T. Hoque "CanTree: a canonical-order tree for incremental frequent-pattern mining", Knowledge and Information Systems – Springer, pp. 287-311, April 2007.

[9] D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, T. Yiu, "MAFIA: A Maximal Frequent Itemset Algorithm", IEEE transactions on knowledge and data engineering, vol. 17, no. 11, November 2005.

[10] K. Gouda, M.J. Zaki, "GenMax: An Efficient Algorithm for Mining Maximal Frequent Itemsets", Data Mining and Knowledge Discovery - Springer, Volume 11, Issue 3, pp. 223-242, November 2005.

[11] C.I. Ezeife, Y. Su, "Mining incremental association rules with generalized FP-tree", 15th Conference of the Canadian Society for Computational Studies of Intelligence, Volume 2338 - Springer, Heidelberg, pp 147-160, 2002.

[12] M.J. Zaki, K. Gouda, "Fast vertical mining using diffsets", Proc. of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - ACM Press, Pp. 326–335, 2003.

[13] J. Pei, J. Han, H. Lu, S. Nishio, S. Tang, D. Yang, "H-Mine: Fast and space-preserving frequent pattern mining in large databases", IIE Transactions, Volume 39(6), Pp. 593-605, 2007.

[14] J.H. Chang, W.S. Lee, "Finding Recent Frequent Itemsets Adaptively over Online Data Streams", Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, SIGKDD, Pp. 487-492, August 24-27, 2003.

[15] S. Kotsiantis, D. Kanellopoulos, "Association Rules Mining: A Recent Overview", GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), pp. 71-82, 2006.

[16] F.M. Christian, N.C. Chauhan, N.B. Prajapati, "A Comparative Study of Frequent Pattern Recognization Techniques from Stream Data", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2(1), January 2014.

[17] P. Fournier-Viger, U. Faghihi, R. Nkambou, E.M. Nguifo "CMRules: Mining Sequential Rules Common to Several Sequences", Volume 25(1), Pp.63–76 , Elsevier - February 2012.

[18] K. Lai, N. Cerpa, "Support vs Confidence in Association Rule Algorithms", Proceedings of the OPTIMA Conference, Curico, 2001.

[19] F. Coenen, G. Goulbourne, P. Leng, "Tree Structures for Mining Association Rules", Data Mining and Knowledge Discovery, Kluwer Academic Publishers, volume 8, Pp.25–51, 2004.

[20] Y.H. Hua, Y.L. Chen, "Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism", Volume 42(1), Pp. 1–24, October 2006.

[21] T. Gueniche, P. Fournier-Viger, V. Tseng, "Compact Prediction Tree: A Lossless Model for Accurate Sequence Prediction", 9th International Conference, ADMA, Part II, vol. 8347, Springer, Heidelberg, pp. 177–188, 2013.

[22] P. Fournier-Viger, A. Gomariz, T. Gueniche, E. Mwamikazi, R. Thomas, "TKS: Efficient Mining of Top-K Sequential Patterns", 9th International Conference, China, Part I, Volume 8346, Springer Heidelberg, pp. 109-120, December 14-16, 2013.

[23] K. Gouda, M.J. Zaki, "Efficiently Mining Maximal Frequent Itemsets," Proc. First IEEE International Conference of Data Mining, Pp.163 – 170, November 2001.

[24] J. Han, M. Kamber, J. Pei, "Data Mining: Concepts and Techniques", 2nd edition, Morgan Kaufmann, San Francisco, 2006.