ISSN: 2321-7782 (Online)
Volume 2, Issue 11, November 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Recent study of exploring Association Rules from Dynamic Datasets

J.K.Kavitha[1]
Research Scholar
Department of Computer Science and Engineering
Anna University
Chennai – India

D.Manjula[2]
Associate Professor
Department of Computer Science and Engineering
Anna University
Chennai – India

Abstract: Association rule mining is one of the most challenging areas of data mining. It aims at finding interesting patterns among the databases. In a dynamic database where the new datasets are inserted into the database, keeping patterns up-to-date and discovering new pattern are challenging problems. This may introduce new association rules and some existing association rules would become invalid. This may cause iterative database scans and high computational costs. Thus, the maintenance of association rules for dynamic datasets is an important problem. Hence it is important in developing techniques in such a way that interesting rules are mined effectively from dynamic datasets. This paper provides an overview of techniques that are used to improvise the efficiency of Association Rule mining from dynamic datasets.

Keywords: Association rule Mining; Incremental Rule Mining; Data Mining.

I. INTRODUCTION

Data mining and knowledge discovery is the process of discovering and extracting information or pattern from the relevant sets of data in database. Among discovering many kinds of knowledge in database, Association rules mining was a form of data mining to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the databases. This is due to its wide applicability in many areas, including decision support, market strategy and financial forecast.

Association rule mining is one of the most challenging areas of data mining which was introduced in Agrawal et al., (1993) to discover the associations or co-occurrence among the different attributes of the dataset. Several algorithms like Apriori(Agrawal et. al., 1993), SETM (Houtsma and Swami, 1993), AprioriTID (Agrawal and Srikant, 1994), DIC (Brin et al., 1997), partition algorithm (Savasere et al.,1995), Pincer search (Lin and Kedem, 1998), FP-tree (Han et al., 2000) etc. have been developed to meet the requirements of this problem.Mining association rules can be decomposed into two steps: the first is generating frequent itemsets. The second is generating association rules. The main challenge in association rule is to identify frequent itemsets. Finding frequent itemset is one important step in association rule mining. Since the solution of second subproblem was straightforward, most of the researchers had focus on how to generate frequent itemsets. However, the rules discovered from a database only reflect the current state of the database.

But in a dynamic database where the new datasets are inserted into the database, keeping patterns up-to-date and discovering new pattern are challenging problems. The problem of discovering the frequent itemsets becomes more time consuming if the dataset is incremental in nature. In the incremental dataset, new records are added time to time. Generally the size of the increments, i.e. the number of transactions added to the dataset, is very small in comparison to the whole dataset. But due to the addition of these new transactions, set-up of the rules in the updated dataset may be changed. Some of the new itemsets may become frequent, while some previously derived frequent set may become infrequent. For example, say A, B and C are three distinct itemsets of a dataset having 1000 records with support count 210, 190 and 220 respectively. Subject to the minimum support of 20%, A and C are frequent itemsets but B is not. Now, another 100 records, where 5, 50 and 2 records

contain A, B and C are added to the dataset. In the updated dataset, the support of the itemset A and C falls below the threshold but the support of B goes above the threshold. In other words A and C becomes infrequent but B becomes frequent in updated dataset.

Due to these changes of frequent itemsets, some earlier derived rules may be dropped and some new rules may come up. For the upto-date rules over the total dataset, if the association mining technique redo the rule generation process for the whole dataset, based on the frequent itemsets, simply by discarding the earlier computed results, it will inefficient. It is mostly due to the multiple scanning over the older dataset. If the results of the older dataset are reused for updating the frequent itemsets, then a significant amount of time may be saved. FUP (Cheung et al., 1996), FUP2 (Cheung et al., 1997), MAAP (Ezeife and Su, 2002), Borders (Feldman et al., 1999), Modified borders (Das and Bhattacharyya, 2005), DHP (Park et al., 1995), UWEP (Ayan et al., 1999) etc. are some of the existing algorithms which attempt to discover the frequent itemsets with minimum number of scanning over the old dataset. Several other works are also found in Hong et al. (2008), Huang et al. (2007), Ou et al. (2008), Kao et al. (2005), Li et al. (2004), Tseng et al. (2008) that has given some attention to the incremental rule mining problem.

To re-mining the frequent itemsets of the whole updated database is clearly inefficient, because all computations done in the previous mining of original database are not reused. The main idea of the association rule mining in dynamic database refers to optimizations that can be done across mining computations over updated dataset based on previously stored knowledge. The solution is to survey different aspects which are discussed in the several research papers and after analyzing those research papers, conclude a solution which is best in efficiency and performance. This paper provides an overview of some techniques that are used to improvise the efficiency of Incremental Association Rule Mining (IARM) with their strength and weakness.

II. Improving the Efficiency of IARM

The main goal of Incremental Association Rule mining (IARM) is to solve the updating problem of association rules after a number of new records have been added to a database. The factors to be considered while improving the efficiency of Incremental Association Rule mining are as follows:

- Reducing the number of scans in the original database.
- Reducing the unnecessary intermediate results to be stored to minimize memory utilization.
- Reducing the total execution and computation time.
- Reducing the resource utilization.
- Increase the performance by exploring fine rules.

III. IARM Approach

Association mining over dynamic dataset is a challenging area of research for the data mining researchers. Several recent works can be found in the survey to meet this challenge. Some of them are: Pre-FUFP (Chun-Wei Lin et al., 2009), False Positive Item set **(**Amornchewin et al., 2009), Probability-based (Amornchewin et al., 2009) Genetic (B. Nath et.al, 2010)**.**

A. *Pre-FUFP*

In the precedent, Han, Pei, and Yin (2000) proposed the Frequent-Pattern tree (FP-tree) structure for efficiently mining association rules without generation of candidate itemsets. The FP-tree (Han et al., 2000) was used to compress a database into a tree structure which stored only large items. They must process all the transactions in a batch way. In real-world applications, new transactions are usually inserted into databases incrementally. Later Hong, Lin, and Wu (2006) modified the FP-tree structure and designed the fast updated frequent pattern trees (FUFP-trees) to efficiently handle newly inserted transactions based on the FUP (Fast-Updated Algorithm which was proposed by Cheung et al. (1996)) concept. The FUFP-tree structure was similar to the FP-tree structure except that the links between parent nodes and their child nodes were bi-directional. Besides, the

counts of the sorted frequent items were also kept in the Header_Table of the FP-tree algorithm. Experimental results showed that the FUFP-tree maintenance algorithm could achieve a good performance for handling the new inserted transactions.

The Pre-FUFP (Chun-Wei Lin et al., 2009) algorithm is the modification of the FUFP-tree algorithm for incremental mining based on the pre-large concept (Hong et al., 2001). Based on two support thresholds, this approach can effectively handle cases in which itemsets are small in an original database but large in newly inserted transactions. Using two user-specified upper and lower support thresholds, the pre-large itemsets act as a gap to avoid small itemsets becoming large in the updated database when transactions are inserted. When new transactions are added, the proposed Pre-FUFP maintenance algorithm processes them to maintain the FUFP tree. It first partitions items of new transactions into three parts according to whether they are large, pre-large or small in the original database. Each part is then processed in its own way. The Header_Table and the FUFP-tree are correspondingly updated whenever necessary.

The Pre-FUFP algorithm does not require rescanning the original databases to construct the FUFP tree until a number of new transactions have been processed. The number is determined from the two support thresholds and the size of the database. Experimental results also show that the Pre FUFP-tree maintenance algorithm has a good performance for incrementally handling new transactions.

Strengths:

- The Pre FUFP-tree structure is used efficiently and effectively to handle new transactions.
- It reduces the rescanning process to construct the FUFP tree based on two user-specified upper and lower support thresholds.
- The pre-large itemsets act as a gap to avoid small itemsets becoming large in the updated database when transactions are inserted.

Weakness:

- The algorithm requires scanning of the original databases when a large number of new transactions have to be processed.
- The two support thresholds and the size of the database determine the rescanning of original database.

B. *False Positive Item set*

The maintenance of association rules for dynamic database is an important problem. In the past, Borders (Feldman et al., 1999) approach maintains both frequent itemsets and border itemsets. The border itemset is not a frequent itemset but all its proper subsets are frequent itemsets. The approach need to keep a large number of border itemsets in order to reduce scanning times of an original database. Basically, the border-based algorithms start by scanning a new database. Then, the border-based algorithms update support counts of all frequent sets and border sets. Most updated frequent itemsets can be found not only from frequent itemsets but also from border itemsets. This can reduce scanning times of an original database. However, when new frequent itemsets are introduced as updated frequent itemsets, several database scanning is required to obtain support counts of the new frequent itemsets and their subsets. Adnan et al. (2005) shows that the execute time of the border-based algorithms can severely slower than that of Apriori when new frequent itemsets are introduced as updated frequent itemsets.

The False Positive Itemset **(**Amornchewin et al., 2009) algorithm, which is an incremental algorithm, to deal with this problem. The main goal of this algorithm is to solve the updating problem of association rules after a number of new records have been added to a database. The algorithm uses maximum support count of 1-itemsets obtained from previous mining to estimate infrequent itemsets, called false positive itemsets, of an original database. False positive itemsets will capable of being

frequent itemsets when new transactions are inserted into an original database.The False Positive Itemset Algorithm provides the new idea to avoid scanning the original database. It computes not only frequent itemset but also itemset that may be potentially large in an incremental database called "False positive Itemset". The algorithm finds all possible k itemset of false positive itemset in original database. If member of frequent for each iteration is more than or equal to k-itemset. This idea is guarantee that false positive itemset algorithm are cover all frequent itemset that occur in updated database.In this approach, an original database is firstly mined and all frequent itemsets and false positive itemset. Secondly each incremental dataset in mined and updated to frequent and false positive itemset. The result of updating, some infrequent itemsets or new itemsets may be changed into frequent itemset.

Strengths:

- Updations of the new transactions are quickly because it can use the information from the existing original database.
- The False positive itemset algorithm has much better running time that of FUP, Border and Pre-large algorithm based on experimental results.

Weakness:

- Assumption that a minimum support factor and a confidence factor do not change, false positive itemset algorithm can guarantee to discover frequent itemsets.
- Rescanning has to be performed when a huge number of datasets inserted at a time.

C. *Probability-Based*

An incremental association rule discovery can create an intelligent environment such that new information or knowledge such as changing customer preferences or new seasonal trends can be discovered in a dynamic environment. The rules discovered from a database only reflect the current state of the database. However, in a dynamic database where new transactions are inserted frequently, association rules discovered in the previous database possibly no longer valid and interesting rules in the updated database. As a result, new business information such as changing customer preferences or new seasonal trends may not be discovered. To create an intelligent environment such that new business information can be discovered in a dynamic database, association rules algorithms should be capable of mining a dynamic database incrementally. The probability-based incremental association rule discovery (Amornchewin et al., 2009) algorithm is to deal with this problem. The algorithm uses the principle of Bernoulli trials to find expected frequent itemsets. It is capable of dynamically discovering new association rules when a number of new records have been added to a database. The probability is used to predict infrequent itemsets that are capable of being frequent itemsets after a number of new records have been added to a database. That infrequent itemsets is called expected frequent itemsets. The algorithm can reduce a number of times to scan an original database.

When a dynamic database is inserted new transactions, not only some existing association rules may be invalidated but also some new association rules may be discovered. This is the case because frequent itemsets can be changed after inserting new transactions into a dynamic database.An original database, which is a database before being inserted new transactions, is firstly mined to find all frequent itemsets that satisfy a minimum support count. The probability-based algorithm predicts and keeps expected frequent itemsets that may become frequent itemsets if new transactions are inserted into the original database.The process of inserting m transactions into an original database of n transaction can be considered as (m+n) Bernoulli trials, which are (m+n) sequence of identical trials. Each itemset has its probability of appearing in a transaction, i.e. the probability of success. The probability-based algorithm generates candidate next k-itemsets by using both frequent k-itemsets and expected frequent k-itemsets. Firstly, new itemsets is obtained by union of frequent k-itemsets and expected frequent k-itemsets. Then,

the candidate (k+1)- itemsets is obtained by self joining the new itemset together. Both frequent (k+1)-itemsets and expected frequent (k+1)-itemsets can be found similar to that of k-itemsets.

Strengths:

- The probability based algorithm can reduce a number of times to scan an original database.
- New pruning And Updating methodology followed to handle incremental datasets.
- The probability based algorithm has much better running time than FUP, Borders and Pre-large algorithm based on experimental results..

Weakness:

- Assumption that the minimum support and confidence do not change, the algorithm can maintain association rules for a dynamic database.
- An expected frequent itemset need to be obtained from an original database before an increment database is available.
- The approximation of the probability of success of an itemset determines the expected frequent itemset. word “data” is plural, not singular.

D. *Genetic Algorithm*

The problem of discovering the frequent itemsets becomes more time consuming if the dataset is incremental in nature. It has been observed that most of existing techniques suffer from the following disadvantages:

- A two phase association mining often can be found to be time and resource consuming in case of larger incremental datasets.
- Due to conversion of the real-life data into market-basket domain, information loss occurs.
- Single objective function (i.e. based on only frequency of occurrence) based rules generation often can be found to be non-interesting.

To address these issues, a single phase incremental association mining technique based on genetic algorithm, which can extract reduced set of interesting rules from real-life datasets without transforming it into the market basket domain. This technique found to be significant in view of the following points:

- During extraction of the rules, it evaluates the rules based not only on the *support count*, but also on the measures *comprehensibility* and *interestingness.*
- It does not require transforming the dataset into market basket domain.
- It avoids the frequent itemset generation phase, rather it generates the rules directly.

In this method, the association rules are extracted from a continuous valued dataset, using genetic algorithm. This algorithm is free from the above mentioned difficulties. During extraction of the rules, generally predictive accuracy or confidence of a rule is used to evaluate the rules. And for this, the support of the different sets of items is needed. The above mentioned algorithms (Pre-FUFP (Chun-Wei Lin et al., 2009), False Positive Item set **(**Amornchewin et al., 2009), Probability-based (Amornchewin et al., 2009)) concentrate on the efficient extraction of the itemsets that meets the minimum threshold requirements. However, the Genetic approach evaluates the rules based on three different measures, namely *support count, comprehensibility* and *interestingness*. If a huge number of attributes are involved in a rule then the rule may be found difficult and hence useless from the point of understandability. Involvement of fewer attributes makes a rule more understandable. *Comprehensibility* is a measure which can be estimated by the number of attributes involved in the rule and tries to quantify the

understandability of the rule. Since association rule mining is a part of data mining process that extracts some hidden information, it should extract those rules that have comparatively less occurrence in the entire database. *Interestingness* is a measure, which can be estimated by checking how surprising the rule is. Such a surprising rule may be more interesting to the users.

This approach attempts to solve the association rule-mining problem with a Pareto based Genetic (B. Nath et.al, 2010) algorithm. Here, the first task is to represent the possible rules as *chromosomes*, for which a suitable *encoding/ decoding* scheme is required. Each *chromosome* represents a separate rule containing both the *antecedent* and *consequent* parts. After getting the chromosomes, various genetic operators can be applied on it. Presence of large number of attributes in the records results in large chromosomes, thereby needing multi-point crossover. Since it may not be possible to load the complete dataset to memory, only a partition of the dataset is loaded to the memory and is used for mining the rules. After the specified number of generations is completed by the Pareto based genetic algorithm, the extracted rules are stored. And extraction of rules is done in the next partition. This process continues till all the partitions are used for mining. Then the whole dataset is used to find the different measures of the rules from all partitions.

Strengths:

- The algorithm do not need the market basket encoded data, it can work on the original continuous valued dataset.
- No separate frequent itemset generation phase is needed; it can produce the rules directly.
- User parameters like minimum support and minimum confidence are not required here and hence they cannot affect the execution time of the algorithm.
- Needs only one scanning of the whole dataset to produce the correct rules.
- Small number of rules will be generated and is controlled by the population size of the genetic algorithm.

Weakness:

- Execution time of this algorithm is higher, since it may not be possible to load the complete dataset to memory, only a partition of the dataset is loaded to the memory and is used for mining the rules.

IV. Conclusion and Future Work

From the above discussion and analysis of the existing techniques of incremental Association Rule Mining, it can be observed that each algorithm can extract the association rules from an incremental dataset with their pros and cons. The best technique for updation problem of incremental mining depends on factors like reducing the number of scans on original database, improving performance etc,. But all the existing Incremental algorithms works well only when the number of new records to be inserted was limited. Those techniques failed to handle huge number of new records. The existing techniques also lack to solve the record modification (updation and deletion of existing records) problem. Thus an appropriate technique has to be employed to handle both huge incremental datasets and record modifications efficiently. This research is prepared to propose a new technique to solve this issue.

References

1. Agrawal R, Imielinski T and Swami A, “Mining Association Rules between Sets of Items in Large Databases”, ACM SIGMOD Int’l Conference on Management of Data,1993.
2. Houtsma M and Swami A., “Set Oriented Mining for Association Rules in Relational Databases”, Technical Report, RJ 9567, IBM Almaden Research Center, San Jose, California, October 1993.
3. Agrawal R and Srikant R, “Fast Algorithms for Mining Association Rules in Large Databases”, Proceedings of 20th Int’l Conference on VLDB, August-September, Chile, 1994.
4. Brin S, Motwani R, Tsur D and Ullman J, “Dynamic Itemset Counting and Implication Rules for Market-Basket Data”, Proc. of 1997 SIGMOD, Montreal, June 1997.

5. Savasere A, Omiecinski E and Navathe S, “An Efficient Algorithm For Mining Association Rules in Large Databases”, Proceedings of 21st Conference on Very Large Databases, Zurich, September,1995.

6. Lin D-I., Kedem Z. M. , "Pincer Search: A New Algorithm for Discovering Maximum Frequent Set”, Sixth International Conference on Extending Database Technology, March 1998.

7. Han J, Pei J and Yin Y , “Mining Frequent Patterns without Candidate Generation”, Proceedings of the 2000 ACM-SIGMOD International Conference on Management of Data, Dallas, Texas, USA, 2000.

8. Cheung D W, Han J, Ng V T and Wong C Y “Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique”, 12th International Conference on Data Engineering, New Orleans,Louisiana, 1996.

9. Cheung D W, Lee D W, and Kao S D, “A General Incremental Technique for Maintaining Discovered Association Rules”, In Proceedings of the Fifth International Conference on Database System for Advanced Applications, Melbourn, Australia,1997.

10. Ezeife C I and Su Y , “Mining Incremental Association Rules with Generalized FP Tree”, Proceedings of 15th Canadian Conference on Artificial Intelligence, AI2002, Calgary, Canada, May 2002.

11. Ayan N. F., Tansel A. U., Arkun M.E. , “An Efficient Algorithm to Update Large Itemsets with Early Pruning", In Knowledge Discovery and Data Mining, pp 287-291,1999.

12. Park J. S., Chen M. S., Yu P.S., “An Efficient Hash Based Algorithm for Mining Association Rules”, Proceedings of ACM SIGMOD’95, pp 175-186, May, 1995.

13. Feldman R, Aumann Y and Lipshtat O, “Borders : An Efficient Algorithm for Association Generation in Dynamic Databases”, Journal of Intelligent Information System, Pages 61–73,1999.

14. Das A and Bhattacharyya D. K. , “Rule Mining for Dynamic Databases”, AJIS, 13, No.1, pp 19-39,2005.

15. Tseng M-C., Lin W-Y and Jeng R., "Incremental Maintenance of Generalized Association Rules Under Taxonomy Evolution", Journal of Information Science, 34, No. 2, pp 174-195, April,2008.

16. Ou J-C., Lee C-H and Chen M-S., "Efficient Algorithms for Incremental Web Log Mining with Dynamic Thresholds", The International Journal on Very Large Data Bases, 17, No. 4, pp 827-845, July 2008.

17. Hong T-P., Lin C-W and Wu Y-L, "Incrementally Fast Updated Frequent Pattern Trees", Expert Systems with Applications: An Int’l Journal, 34, no. 4, pp 2424-2435, May 2008.

18. Huang J-P., Chen S-J and Kuo H-C, "An Efficient Incremental Mining Algorithm-QSD", Intelligent Data Analysis, 11, No. 3, pp 265-278, August 2007.

19. Kao B., Zhang M., Yip C-L., Cheung D. W and Fayyad U, "Efficient Algorithms for Mining and Incremental Update of Maximal Frequent Sequences", Data Mining and Knowledge Discovery, 10, No. 2, pp 87-116, March 2005.

20. Li J., Manoukian T., Dong G and Ramamohanarao K, "Incremental Maintenance on the Border of the Space of Emerging Patterns", Data Mining and Knowledge Discovery, 9, No. 1, pp 89-116, July 2004.

21. Chun-Wei Lin, Tzung-Pei Hong and Wen-Hsiang Lu,” The Pre-FUFP algorithm for incremental mining”, Expert Systems with Applications , 36, pp 9498–9505,2009.

22. Chun-Wei Lin , Tzung-Pei Hong , Wen-Hsiang Lu ,Han, J., Pei, J., and Yin, Y,” Mining frequent patterns without candidate generation “, ACM SIGMOD international conference on management of data ,(pp. 1–12),2000.

23. Hong, T.P., Lin, J.W., & Wu, Y.L ,” A fast updated frequent pattern tree”, IEEE International conference on systems, man, and cybernetics, (pp. 2167–2172), 2006.

24. Cheung, D. W., Han, J., Ng, V. T., and Wong, C. Y., “ Maintenance of discovered association rules in large databases: an incremental updating approach.”, IEEE international conference on data engineering, (pp. 106–114), 1996.

25. Hong, T. P., Wang, C. Y., and Tao, Y. H. ,” A new incremental data mining algorithm using pre-large itemsets”, Intelligent Data Analysis, 5(2), pp 111–129,2001.

26. Ratchadaporn Amornchewin1 and Worapoj Kreesuradej2 ,”False Positive Item set Algorithm for Incremental Association Rule Discovery”, International Journal of Multimedia and Ubiquitous Engineering ,Vol. 4, No. 2, April, 2009.

27. Feldman, R., Aumann,Y., and Lipshtat, O., “Borders: An efficient algorithm for association generation in dynamic databases”, Journal of Intelligent Information System, pp 61-73,1999.

28. Adnan, M., Alhajj, R., and Barker, K., “Performance Analysis of Incremental Update of Association Rules Mining Approaches”,IEEE International Conference on Intellgent Engineering System ,pp 129-134, Sept, 2005.

29. Ratchadaporn Amornchewin and Worapoj Kreesuradej,” Mining Dynamic Databases using Probability-Based Incremental Association Rule Discovery Algorithm”,Journal of Universal Computer Science, vol. 15, no. 12, pp 2409-2428,2009.

30. B. Nath1, D K Bhattacharyya2 and A Ghos,”Discovering Association Rules from Incremental Datasets”, International Journal of Computer Science & Communication,Vol. 1, No. 2, pp. 433-441,2010.

AUTHOR(S) PROFILE

J.K.Kavitha, Assistant Professor. She received the M.Tech. Degree from the Department of Computer Science and Engineering at Anna University, Chennai, in 2009. She is now a Ph.D. candidate in the Department of Computer Science and Engineering at Anna University, Chennai. Her research interests include Data mining, Image Processing, Cloud Computing.