Volume 5, Issue 1, January 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Compact Table Based Time Efficient Technique for Mining Frequent Items from a Transactional Data Base

Nilesh Singh Lodhi, Asst. Professor Jitendra Dangra, Professor Dr. M.K. Rawat

M.tech, Department of Computer Science,

LNCT, Indore, (m.p) . India

Abstract: Frequent item set mining is a heart favorite topic of research for many researchers over the years. It is the basis for association rule mining. Association rule mining is used in many applications like: market basket analysis, intrusion detection, privacy preserving, etc. In this thesis, we have developed a method to discover large item sets from the transaction database. The proposed method is fast in comparison to older algorithms. Also it takes les main memory space for computation purpose. Experimental results have proved that the proposed scheme is time and memory efficient.

Keywords: KDD, TIARM, FUFP, CDB

I. INTRODUCTION

With the increase in Information Technology, the size of the databases created by the organizations due to the availability of low-cost storage and the evolution in the datacapturing technologies is also increasing,. These organization sectors include retail, petroleum, telecommunications, utilities, manufacturing, transportation, credit cards, insurance, banking and many others, extracting the valuable data, it necessary to explore the databases completely and efficiently. Knowledge discovery in databases (KDD) helps to identifying precious information in such huge databases. This valuable information can help the decision maker to make accurate future decisions. KDD applications deliver measurable benefits, including reduced cost of doing business, enhanced profitability, and improved quality of service. Therefore Knowledge Discovery in Databases has become one of the most active and exciting research areas in the database community.

In recent years the size of database has increased rapidly. This has led to a growing interest in the development of tools capable in the automatic extraction of knowledge from data. The term data mining or knowledge discovery in database has been adopted for a field of research dealing with the automatic discovery of implicit information or knowledge within the databases. The implicit information within databases, mainly the interesting association relationships among sets of objects that lead to association rules may disclose useful patterns for decision support, financial forecast, marketing policies, even medical diagnosis and many other applications.

Data mining generally involves four classes of task; classification, clustering, regression, and association rule learning. Data mining refers to discover knowledge in huge amounts of data. It is a scientific discipline that is concerned with analyzing observational data sets with the objective of finding unsuspected relationships and produces a summary of the data in novel ways that the owner can understand and use.

II. RELATED WORK

Apriori Algorithm [3] [5] is one of simplest approach to generate frequent pattern. This algorithm is recursive in nature, so processing is iterative (brute force approach). In first iteration candidate-set of size-1 (C1) is generated, and then whole database scanning is done. The items having support greater than user defined minimum support are used as frequent items (L1) of size-1. This process continuously till Ci or Li becomes empty. It is basically candidate-set generation and test approach. Disadvantages of this is that large number of candidate generation and time consuming as it required multiple passes for processing.

FP-tree [6] [7] [8] is one of best approach to discover frequent pattern to overcome the drawback of the apriori algorithm. It requires only two passes of processing. One pass is required for ordering and structuring frequent items other pass is for inserting those frequent items in the tree. FP-tree as better performance than Apriori as reduce database scan. Since even if small insertion is done, restructuring of item is required again to arrange in descending order. FP-growth [8] [9] algorithm is applied on FP-tree to discover frequent pattern. It is based on divideconquer approach to discover frequent pattern of various sizes.

G. Pradeepini and S. Jyothi [10] have proposed algorithm called Tree-based incremental Association rule mining (TIARM) algorithm. This algorithm has two different mechanisms. First, is to generate INC-tree which is more enhanced than FP-tree to make tree more compact in nature. Second, TIARM is applied on INC-tree to discover frequent patterns of different sizes.

The process of generating INCtree is same as that of the FP-tree with single pass processing. By using conditional pattern base and FP-tree, frequent patterns are generated without candidate itemset.

Liu Jian-ping et al [11] present an algorithm called FUFPtree based incremental association rule mining algorithm (Pre-FP). It is based FUFP [12] [14] (Fast Updated Frequent Pattern) concept. The major idea of FUFP is re-use of previously mine frequent items to update with incremental database. It reduces number of candidate set in updating process. All the links in FUFP are bidirectional where in FP-tree links are only in single direction. Advantage of bidirectional link is easy to add remove child node without much reconstruction. This FUFP structure is used as input to the Pre-large, which gives positive count difference whenever small amount of data is added to original database. It deals with change in database in case of inserting new transaction. The algorithm classify items into the three categories: Frequent, infrequent and pre-large. PreLarge [13] itemsets has two support threshold values i.e. upper and lower threshold. These support thresholds are helpful for maintaining cost while insertion and deletion of items into original dataset. These items are differentiated between nine cases in first pass. Each case is handled efficiently to discover frequent pattern in second pass. Such type of characteristics is useful for real-world applications such web mining.

Chowdhury Farhan Ahmed et al. [18] have proposed two Single-pass incremental and interactive frequent itemsets mining algorithms with single database scan. One is weight in ascending order (i.e. IWFPwa) in which each item is having specific weight (different degree of importance). In this algorithm the given weight of items are used to calculate support of items in the database. Those weights are sorted in ascending order with highest weight in bottom this leads to database size reduction. This compressed structure is used to build FP-tree and then FP-growth algorithm is applied to discover frequent pattern. Another algorithm is based on frequency by arranging it in descending order (i.e. IWFPfd). The main advantage of this algorithm is prefix sharing of node [19] with compact structure of the tree. Numbers of nodes are less as compared to the previous method which saves memory space.

Siqing Shan et al. [15] have presented Incremental Association Rules Mining method based on Continuous Incremental Updating Technique. Transaction Amalgation Algorithm is used to merge the transaction in transaction database based on quantity present in transaction in descending order. That reduces the overall size of the database drastically saves memory space. T-tree algorithm is applied on these database which works as FP-tree. Finally T-tree is given as input to the FP-growth algorithm to discover frequent pattern. Each pattern in overall database (original+ new) is applied to candidate pattern pool, where it is classified in four cases:

i. Pattern may frequent in old database and not frequent loser in increment to database
ii. Frequent in both old database as well as increment to it
iii. Not frequent in both old database as well as increment to it
iv. Frequent in increment to database and not frequent in old database

D. Kerana Hanirex and Dr. M. A. Dorai Rangaswamy [20] have proposed clustering based incremental algorithm to discover Frequent Patterns. The partitioning algorithm has proposed to generate cluster. Then Improved Apriori Algorithm [21] is applied to generate frequent patterns. If pattern is frequent then it is present in any of the cluster. Whenever new transaction is added to the database it treated as new cluster. Again Improved Apriori algorithm is applied to discover newly frequent pattern in incremental database. This algorithm has better efficiency than previous Apriori algorithm by reducing memory space and number of passes.

Liu Han-bing, Zhang Ya-juan, Zheng Quan-lu and Ye Mao-gong [22] has proposed Incremental Frequent Pattern mining algorithm based on AprioriTidList Algorithm [23]. This algorithm also improves Apriori performance by pruning transaction. It requires only one database scan which make it more efficient. It scans a database and creates a Tid List .It does not uses whole database to count support value instead it consider particular large item in transaction with identifier TID. If transaction does not contain that large item then that transaction is deleted which reduces database size drastically. Tid list of

Item „I‟ contain list of all the transaction in which I is present. Tid list of Item „J‟ contains list of all the transactions in which J is present. Intersection of both the list gives the list of transaction in which both I and J are present. When new data is added it discover frequent pattern using old frequent pattern.

Shih-Sheng Chen et al. [16] have proposed a method for discovery of frequent periodic pattern using multiple minimum supports. This very efficient approach to find frequent pattern because it is based on multiple minimum support based on real time event. All the items in the transactions are arranged according to their MIS (Minimum Item Support). It does not hold downward closure property instead it uses sorted closure property based on ascending order. Then it uses PFP [17] (Periodic Frequent Pattern) whose construction is same as that of the FP-tree. Finally, PFP-growth algorithm is applied which is same as that FPgrowth and conditional pattern base is used to discover frequent pattern. This algorithm is more efficient in terms of memory space and database scan by reducing number of candidate set.

III. PROPOSED ALGORITHM

Input:

- *A Transaction Database D*
- *MST – Minimum support Threshold*

Step1: First of all, scan the transaction database D to find out the number of occurrences of all size 1itemsets. Then the support of each item is computed & stored in a data structure. This data structure has two parts: the head part & the body part. The body of this data structure contains all single items with their support.

Step 2: In this step all the size 1 items are arranged in the decreasing order of there support count. This is the candidate set of all size 1 items.

Step 3: In this step, the support of each size 1 item is compared with the minimum threshold known as the MST (Minimum Support Threshold). Eliminate all those size-1 itemsets of step 1 whose support is less than the MST. It will result in a compressed table, which consists of all the frequent items of size1. It is known as compressed data base (CDB)

Step 4: Now sort all the itemsets of last step in descending order of their itemcount (frequency)

Step 5: Create a 2 dimensional data structure (Table) and store the transaction and the correspondent frequency in that table.

Step 6: Then scan the data structure created in step 5 to locate all the K size itemsets. Select only those item sets whose support is greater than the minimum threshold (MST).

Step 7: If the support count of the k size item sets is less than the MST then take K size itemsets and k – 1 size itemsets together to generate a k - 1 size item set. Continue this step until no item set having support greater than the MST is found.

Step 8: All the largest possible size item sets are found in step 7 then by applying the downward closure property all the subsets are also frequent.

Step 9: Now scan the compressed table of step 5. It may contain some frequent item sets of smaller size which have yet not been included in the list of frequent item sets. Now reduce the data base of step 5 by considering only those transactions which contain frequent 1-itemset element but not contain the maximal frequent transaction

Step 10**:** If no such transaction exists in table of step 5 then go to step 11 otherwise repeat step 5 to 10.

Step 11: exit.

Output : *All the frequent item sets*

IV. CONCLUSION

In this thesis, we surveyed the list of existing frequent item set mining techniques. We restricted ourselves to the classic frequent item set mining problem. It is the generation of all frequent item sets that exists in market basket like data with respect to minimal thresholds for support & confidence.we presented a novel algorithm for mining frequent item sets. Frequent item set mining is crucial for association rule mining. We have evaluated the performance of our proposed algorithm. It is fast. Also it is taking less main memory for computation in comparison to previous algorithm.

V. FUTURE WORK

- More compact data structure can be proposed to reduce space consumption
- Our proposed algorithm works for the normal data set. The same algorithm can be extended to work for the uncertain data set.
- One limitation though data mining can help reveal patterns and relationships, it does not tell the user the value or significance of these patterns.It does not tell the users which patterns are sensitive and which are not.
- It can be said that software privacy failures can be direct result of one or more of the following points that are taken from risk management:

REFERENCES

[1] A. Savasere, E. Omiecinski, and S. Navathe. "An efficient algorithm for mining association rules in large databases". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1995, pages 432–443.

[2] Aggrawal.R, Imielinski.t, Swami.A. "Mining Association Rules between Sets of Items in Large Databases". In Proc. Int'l Conf. of the 1993 ACM SIGMOD Conference Washington DC, USA.

[3] Agrawal.R and Srikant.R. "Fast algorithms for mining association rules". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1994, pages 487–499.

[4] Brin.S, Motwani. R, Ullman. J.D, and S. Tsur. "Dynamic itemset counting and implication rules for market basket analysis". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), May 1997, pages 255–264.

[5] C. Borgelt. "An Implementation of the FP- growth Algorithm". Proc. Workshop Open Software for Data Mining, 1–5.ACMPress, New York, NY, USA 2005.

[6] Han.J, Pei.J, and Yin. Y. "Mining frequent patterns without candidate generation". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), 2000

[7] Park. J. S, M.S. Chen, P.S. Yu. "An effective hash-based algorithm for mining association rules". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), San Jose, CA, May 1995, pages 175–186.

[8] Pei.J, Han.J, Lu.H, Nishio.S. Tang. S. and Yang. D. "H-mine: Hyper-structure mining of frequent patterns in large databases". In Proc. Int'l Conf. Data Mining (ICDM), November 2001.

[9] C.Borgelt. "Efficient Implementations of Apriori and Eclat". In Proc. 1st IEEE ICDM Workshop on Frequent Item Set Mining Implementations, CEUR Workshop Proceedings 90, Aachen, Germany 2003

[10] Toivonen.H. "Sampling large databases for association rules". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1996, Bombay, India, pages 134–145.

[11] Nizar R.Mabrouken, C.I.Ezeife. Taxonomy of Sequential Pattern Mining Algorithm". In Proc. in ACM Computing Surveys, Vol 43, No 1, Article 3, November 2010.

[12] Yiwu Xie, Yutong Li, Chunli Wang, Mingyu Lu. "The Optimization and Improvement of the Apriori Algorithm". In Proc. Int'l Workshop on Education Technology and Training & International Workshop on Geoscience and Remote Sensing 2008.

[13] "Data mining Concepts and Techniques" by By Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, 2006.

[14] S.P Latha, DR. N.Ramaraj. "Algorithm for Efficient Data Mining". In Proc. Int'l Conf. on IEEE International Computational Intelligence and Multimedia Applications, 2007, pp. 66-70.

[15] Dongme Sun, Shaohua Teng, Wei Zhang, Haibin Zhu. "An Algorithm to Improve the Effectiveness of Apriori". In Proc. Int'l Conf. on 6th IEEE Int. Conf. on Cognitive Informatics (ICCI'07), 2007.

[16] Q.Lan, D.Zhang, B.Wu. "A New Algorithm For Frequent Itemsets Mining Based On Apriori And FP-Tree". In Proc. Int'l Conf. on Global Congress on Intelligent System, 2009, pp.360-364.

[17] W.LIU, J.CHEn, S.Qu, W.Wan. "An Improved Apriori Algorithm. In Proc. IEEE International Conference, 2008, pp.221-224".

[18] S.P Latha, DR. N.Ramaraj. "Agorithm for Efficient Data Mining". In Proc. Int'l Conf. IEEE International Computational Intelligence and Multimedia Aplications, 2007, pp. 66-70.

[19] M. El-Hajj and O. R. Zaiane. "Inverted matrix: Efficient discovery of frequent items in large datasets in the context of interactive mining". In Proc. Int'l Conf. on Data Mining and Knowledge Discovery (ACM SIGKDD), August 2003.

[20] M. El-Hajj and O. R. Zaiane. "COFI-tree Mining:A New Approach to Pattern Growth with Reduced Candidacy Generation". Proceedings of the ICDM 2003 Workshop on Frequent Itemset Mining Implementations, Melbourne, Florida, USA, CEUR Workshop Proceedings, vol. 90, pp. 112-119, 2003.

[21] Y. G. Sucahyo and R. P. Gopalan. "CT-ITL: Efficient Frequent Item Set Mining Using a Compressed Prefix Tree with Pattern Growth". Proceedings of the 14th Australasian Database Conference, Adelaide, Australia, 2003.

[22] Y. G. Sucahyo and R. P. Gopalan. "CT-PRO: A Bottom Up Non Recursive Frequent Itemset Mining Algorithm Using Compressed FP-Tre Data Structure". In proc Paper presented at the IEEE ICDM Workshop on Frequent Itemset Mining Implementation (FIMI), Brighton UK, 2004.

