Global Journal of Computer Science and Technology
Volume 11 Issue 14 Version 1.0 August 2011
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

A Study on Efficient Data Mining Approach on Compressed Transaction

By V.Vidhya Rani,Dr. I. Elizabeth Shanthi
Avinashilingam Deemed University for Women, Coimbatore

Abstract - Data mining can be viewed as a result of the natural evolution of information technology. The spread of computing has led to an explosion in the volume of data to be stored on hard disks and sent over the Internet. This growth has led to a need for data compression, that is, the ability to reduce the amount of storage or Internet bandwidth required to handle the data. This paper analysis the various data mining approaches which is used to compress the original database into a smaller one and perform the data mining process for compressed transaction such as M2TQT,PINCER-SEARCH algorithm, APRIORI & ID3 algorithm, TM algorithm, AIS & SETM, CT-Apriori algorithm, CBMine, CT-ITL algorithm, FIUT-Tree. Among the various techniques M2TQT uses the relationship of transactions to merge related transactions and builds a quantification table to prune the candidate item sets which are impossible to become frequent in order to improve the performance of mining association rules. Thus M2TQT is observed to perform better than existing approaches.

Keywords : Association rule, merged transaction, quantification table.

GJCST Classification : H.2.8

A Study on Efficient Data Mining Approach on Compressed Transaction

V.Vidhya Rani$^{\alpha}$,Dr. I. Elizabeth Shanthi$^{\Omega}$

Abstract - Data mining can be viewed as a result of the natural evolution of information technology. The spread of computing has led to an explosion in the volume of data to be stored on hard disks and sent over the Internet. This growth has led to a need for *data compression*, that is, the ability to reduce the amount of storage or Internet bandwidth required to handle the data. This paper analysis the various data mining approaches which is used to compress the original database into a smaller one and perform the data mining process for compressed transaction such as M^{2}TQT,PINCER-SEARCH algorithm, APRIORI & ID3 algorithm, TM algorithm, AIS & SETM, CT-Apriori algorithm, CBMine, CT-ITL algorithm, FIUT-Tree. Among the various techniques M^{2}TQT uses the relationship of transactions to merge related transactions and builds a quantification table to prune the candidate item sets which are impossible to become frequent in order to improve the performance of mining association rules. Thus M^{2}TQT is observed to perform better than existing approaches.

Keywords : *Association rule, merged transaction, quantification table.*

I. Introduction

DATA mining is used to help users discover interesting and useful knowledge more easily. It is more and more popular to apply the association rule mining in recent years because of its wide applications in many fields such as stock analysis, web log mining, medical diagnosis, customer market analysis and bioinformatics. The main focus is on association mining and data pre-process with data compression.

The process of mining association rules consists of two main steps:

- Find the frequent item sets or large item sets with a minimum support
- Use the large item sets to generate association rules that meet a confidence threshold.

Finding frequent item sets is more expensive since the number of item sets grows exponentially with the number of items. A large number of increasingly efficient algorithms to mine frequent item sets have been developed over the years .

For example, association rules were first defined for transaction databases [3]. An association rule *R* is an implication of the form $X \Rightarrow Y$, where *X* and *Y* are set of items and $X \cap Y = \emptyset$. The support of a rule $X \Rightarrow Y$ is the fraction of transactions in the database which contain $X \cup Y$. The confidence of a rule $X \Rightarrow Y$ is the fraction of transactions containing *X* which also contain *Y*. An association rule can be considered interesting if it satisfies the minimum support threshold and minimum confidence threshold, which are specified by domain experts. The most common approach to mining association rules consists of two separate tasks: in the first phase, all frequent item sets that satisfy the user specified minimum support are generated; the second phase uses these frequent item sets in order to discover all the association rules that meet a confidence threshold. Since the first problem is more computationally expensive and straight forward, a large number of algorithms have been developed over years.

A transaction database is a set of records representing transactions, where each record consists of a number of items that occurs together in a transaction. The most famous example of transaction data is market basket data, in which each transaction corresponds to the set of items bought by a customer during a single visit to a store. Transaction databases have important role in data mining. Association rules identify relationships among sets of items in a transaction database. Ever since its introduction (Agrawal, Imielinski and Swami 1993), association rule discovery has been an active research area.

This paper is a study of existing data mining techniques for pattern discovery and technique that produce compression form .The rest of the paper is organized as follows: section 2 discusses frequent pattern mining techniques, section 3 reports compact representation of transaction database and mining, section 4 discusses concludes the paper.

II. Frequent Pattern Mining Technique

a) Apriori Algorithm

In computer science and data mining, **Apriori** is a classic algorithm for learning association rules. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). The algorithm attempts to find subsets which are common to at least a minimum number C (the cutoff, or confidence

Author $^{\alpha}$: M.Phil Scholar, Department of computer science Avinashilingam Deemed University for Women, Coimbatore-43.
E-mail : vidhya.rani2k@gmail.com.
Author $^{\Omega}$: Associate Professor Department of computer science,Avinashilingam Deemed University for Women, Coimbatore-43. E-mail :shanthianto@yahoo.com.

threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as *candidate generation*, and groups of candidates are tested against the data.

b) *Pincer-Search Algorithm*

This is an efficient algorithm [14] for discovering maximum frequent set. pincer-search approach combines both top-down searches and bottom –up searches to prune candidates. Bottom–up approach is used in the case ,where all maximal frequent item sets are short whereas top-down approach is used when all maximal frequent item sets are long .It reduces number of candidate and number of passes in frequent set discovery process.

Paper [15], RP-global method as theoretical bound, and works well on small collections of frequent patterns. The Rplocal method is quite efficient and preserves reasonable compression quality. Two algorithms AIS and SETM are combined into a hybrid algorithm called Apriori Hybrid scales linear [16],[17] with the number of transactions has been discussed. It is used for discovering all significant association rules between items in a large database of transaction.

The data mining techniques[6] for database compression such as Apriori algorithm and ID3 algorithm(Goh et al.,2001),Decision Tree(Babu et al.,2001), and FUP&FUP2 algorithm (Lee & Tang,2004) .An efficient approach named as Adjust FTCP-split algorithm for incremental mining to solve the problem of data maintenance when compressed database occur variation.

The data mining in particular focuses on discovering the data usage pattern of compressed transaction from the database.

A sample based method for mining frequent pattern s from database [22] consists of three phases. In the first phase, a small sample of data to estimate the set of frequent patterns. The second phase computes the actual support of the pattern and identifies a subset of pattern that needs to be further examines in the next phase. The third phase explores the set and finds all missing frequent pattern.

A new algorithm[13] suitable for mining association rules in databases is designated as CBMine(Compressed Binary Mine).This algorithm Show better performance the several algorithm like Bandon's Apriori algorithms and in sparse databases .

Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach is a novel data structure[26], *frequent pattern tree* (FP-tree), for storing compressed ,crucial information about frequent patterns, and developed a pattern growth method, *FP-growth*, for efficient mining of frequent patterns in large databases. They have implemented the *FP-growth* method, studied its performance in comparison with several influential frequent pattern mining algorithms in large databases. Our performance study shows that the method mines both short and long patterns efficiently in large databases, outperforming the current candidate pattern generation-based algorithms.

Efficient mining of association rules by reducing the number of passes over the database[32] discusses one algorithm called And, which is usually used in logical algorithms, And Code (AC) algorithm can discover frequent itemsets without producing candidates and setting thresholds for candidates. Frequent itemsets can be fast discovered by corresponding codes which are cited by this paper to describe different itemsets for AC algorithm. The support count of frequent itemsets can be computed before the process of scanning during processes of AC algorithm.

III. Compact Representation of Transaction Database and Mining

a) *M²TQT Approach*

In the research [1], a more efficient approach, called Mining Merged Transactions with the Quantification Table (M²TQT) is proposed, which can compress the original database into a smaller one and perform the data mining process without the problems such as:

- ➢ The compressed database is not reversible after the original database is transformed by the data pre-process step.
- ➢ It is very difficult to maintain this database in the future.
- ➢ Although some rules can be mined from the new transactions, it still needs to scan the database again to verify the result. This is because the data mining step produces potentially ambiguous results.
- ➢ It is a serious problem to scan the database multiple times because of the high cost of re-checking the frequent item sets

M²TQT has the following characteristics:

- ✓ The compressed database can be decompressed to the original form.
- ✓ Reduce the process time of association rule mining by using a quantification table.
- ✓ Reduce I/O time by using only the compressed database to do data mining.
- ✓ Allow incremental data mining.

The paper focuses on compressed transaction, a technology that both reduces the effective price of logical data storage capacity, and improves query performance.

b) *Merge-Mining Algorithm*

This algorithm [1] is used to find frequent itemsets from the new transaction. There are two phases in this algorithm is used.

1. Finding frequent itemsets
2. To prune redundancy.

M^{2}TQT uses transaction relation distance to merge the relevant transactions. Based on the distance relation between transactions. It can be merged with closer relationship to generate a better compressed database. Then we create a quantification table to reduce number of candidate's itemsets to be generated. It helps to prune non-frequent itemsets.

Another algorithm called partition [4] reads the database at most two times to generate all significant association rules. This algorithm is especially suitable for very large size databases .For very large databases, the CPU overhead was reduced by as much as a factor of four and i/o was reduced by order of magnitude.

MAFIA (A maximal frequent item set algorithm)[18] is an algorithm for finding maximal frequent Itemset .The breakdown of algorithmic components and dynamic reordering were quite beneficial in reducing the search space while relative compression of vertical bitmap representation of database with efficient bitmap compression schema.

HI-mine algorithm is effective and efficient and improves the performance of indirect association mining significantly. It consist of a strategy that compresses a transaction database into a super compact transaction database, which dramatically reduces not only the number of transaction in original database, but also memory requirement for storing frequent item and the runtime for mining indirect associations.

Paper [23] reports FPTree algorithm as Horizontal and vertical Compact Frequent Itemset Pattern Mining Tree (HVCFPMINETREE). HVCFPMineTree combines all the maximum occurrence of frequent itemsets before converting into the tree structure a new algorithm HVCFP Mine Tree in both horizontal and vertical layouts. It leads to a compressed FPTree structure in an efficient manner

A new algorithm TM using the vertical database representation has been discussed in paper [24]. Transaction ids of each itemsets are transformed and compressed to continuous transaction interval lists in a different space using the transaction tree and frequent itemsets are found by transaction intervals intersection along a lexicographic tree in depth first order. This compression greatly saves the intersection time. Through experiments, TM algorithm has been shown to gain significant performance improvement over FP-growth and dEclat on datasets with short frequent patterns, and also some improvement on datasets with long frequent patterns. We have also performed the compression and time analysis of transaction mapping using the transaction tree and proved that transaction mapping can greatly compress the transaction ids into continuous transaction intervals especially when the minimum support is high. Although FP-growth is faster than TM in this experiment, the comparison is unfair. In our future work we plan to improve the implementation of the TM algorithm and make a fair comparison with FP-growth.

Compact tree structure called CT-tree[2],to compress the original transactional data. This allows the CT-Apriori algorithm, which is revised from the classical Apriori algorithm, to generate frequent patterns by skipping the initial database scan and reduce great amount of I/O time per database scan's tree data structure will fit into main memory. It will not apply for very large databases.

FIUT-Tree [20],Frequent items ultrametric tree ,this algorithm is used for mining frequent itemsets. It is classified into candidate itemsets approach such as Apriori called Apriori-Like; a method without candidate itemsets-generation approach such as FP-growth algorithm called FP-growth-like. It is an improved method to partition a database by clustering the transactions and significantly reducing the search space. FIUT-tree is also used for storing compressed databases.

CT-ITL algorithm [21] for mining complete set of frequent Itemset. It uses Item-Trans Link(ITL) data structure that combines both horizontal and vertical layout for association rule mining.

A Compress-Based Association Mining Algorithm for Large Dataset [25], is efficient when the size of dataset is huge that cannot be load in the main memory. Outcome of our algorithm is to reduce the search space by eliminating *1-itemsets* from the transactions after the first pass. This feature may prove useful for finding frequent itemset from many real life dense dataset.

A General Incremental Technique for Maintaining Discovered Association Rules [31] is developed for maintaining the association rules discovered in a database in the cases including insertion, deletion, and modification of transactions in the database. The algorithm FUP2 makes use of the previous mining result to cut down the cost of finding the new rules in an updated database. FUP2 can efficiently update the discovered rules when new transactions are added to a transaction database, and obsolete transactions are removed from it. This algorithm has been implemented and its performance is studied and compared with the best algorithms for mining.

IV. Other Related Works

Mining Multiple-Level Association Rules in Large Databases [27] compares the performance of the proposed algorithms of all four algorithms DML T2L1; ML T1LA; ML TML1; and ML T2LA . They are implemented and tested on a Sun/Sparcstation20 with 32 MB of main memory running Solaris 2.5.

Mining important association rules based on the RFMD (Recency, Frequency, Monetary value and

Duration)technique[28] is a novel method that combines RFMD analysis with the association rule mining technique to extract effective rules from distributed databases.

Fuzzy Association Rule Mining Algorithm for Fast and Efficient Performance on Very Large Datasets[33],a new fuzzy Association rule mining (ARM) algorithm meant for fast and efficient performance on very large datasets. As compared to fuzzy Apriori, our algorithm is *8-19 times faster* for the very large standard real-life dataset we have used for testing with various mining workloads, both typical and extreme ones. A novel combination of features like two-phased multiple partition tidlist-style processing, byte-vector representation of tidlists, and fast compression of tidlists contributes a lot to the efficiency in performance. In addition, unlike most two phased ARM algorithms, the second phase is totally different from the first one in the method of processing (individual itemset processing as opposed to simultaneous itemset processing at each *k*-level), and is also many times faster. Our algorithm also includes an effective preprocessing technique for converting a crisp dataset to a fuzzy dataset.

An efficient algorithm for mining association rules in large databases[30],is an efficient algorithm for mining association rules that is fundamentally different from known algorithms. Compared to previous algorithms, our algorithm not only reduces the I/O overhead significantly but also has lower CPU overhead for most cases. It was found that for large databases, the CPU overhead was reduced by as much as a factor of four and I/O was reduced by almost an order of magnitude. Hence this algorithm is especially suitable for very large size databases.

Fast Algorithms for Mining Association Rules[29] proposes an algorithm called AprioriHybrid. Scale-up experiments show that AprioriHybrid scales linearly with the number of transactions. AprioriHybrid also has excellent scale-up properties with respect to the transaction size and the number of items in the database.

V. Conclusion And Future Work

In this paper, we have analyzed the various data mining techniques and algorithms used to compress data such as M^{2}TQT, PINCER-SEARCH algorithm, APRIORI & ID3 algorithm, TM algorithm, AIS & SETM, CT-Apriori algorithm, CBMine, CT-ITL algorithm and FIUT-Tree. These techniques were analyzed. Among them algorithm M^{2}TQT performs in a better way by reducing the processing time and I/O time and by decompression of compressed database to original database and by scanning the transaction database only once. The M^{2}TQT approach utilizes the compressed transactions to mining association rule efficiently with a quantification table.

We have also given the overview of frequent pattern mining, several algorithms and techniques and approaches used for data compression and algorithms for database compresses transaction our next focus is to improve the compression rate by incorporating FP-Tree in M^{2}TQT.

References Références Referencias

1. Jia-Yu Dai, Don-Lin Yang, Jungpin Wu, and Ming-Chuan Hung “An Efficient Data Mining Approach on Compressed Transactions”, International Journal of Electrical and Computer Engineering 3:2 2008
2. Qigarwal,T.an Wan and Aijun An” Compact Transaction Database for Efficient Frequent Pattern Mining”.
3. R.Agarwal, T.Imielinski and A.Swami.Minign, ”Association rules between sets of items in large databases. In proceeding ACM SIGMOD International conference on management of data, pages 207-216, Washington, D.C., USA, May 1993.
4. Savasere, E. Omiecinski, and S. Navathe, "An Efficient Algorithm for Mining Association Rules in Large Databases," in *Proceedings of the 21st International Conference on Very Large Data Bases*, pp. 432-444, 1995.
5. Santhosh Kumar, “Implementation of Web Usage Mining Using APRIORI and FP Growth Algorithms”. Int. J. of Advanced Networking and Applications 400 Volume:01, Issue:06, Pages: 400-404 (2010)
6. Bratko ANDREJ.BRA et al,”Spam Filtering Using Statistical Data Compression Models”, Journal of Machine Learning Research 7 (2006) 2673-2698 Submitted 03/06; Revised 09/06; Published 12/06
7. Chin-Feng Lee, Chia-Hsing Tsai ,” Efficient Associating Mining Approaches for Compressing Incrementally Updatable Native XML Databases “.IAENG International Journal of Computer Science, 33:1, IJCS_33_1_14
8. G. Grahne and J. Zhu, "Fast algorithms for frequent item set mining using FP-trees," *IEEE Transactions on Knowledge and Data Engineering,* Vol. 17, pp. 1347-1362, 2005.
9. J. Arokia Renjit Dr.K.L.Shunmuganathan ,”Mining The Data From Distributed Database Using An Improved Mining Algorithm” (IJCSIS) International Journal of Computer Science and Information Security,Vol. 7, No. 3, March 2010
10. M.H.Margahny and A.A.Mitwaly, “Fast Algorithm for Mining Association Rules”, AIML 05 Conference, 19-21 December 2005, CICC, Cairo, Egypt.
11. M.K. Sharma*, Jyotsana Sah**Applications of Data Compression Approach In Data Warehouse Design, Proceedings of 2nd National Conference on Challenges & Opportunities in Information Technology (COIT-2008) RIMT-IET, Mandi Gobindgarh. March 29, 2008 *Management of Data*,

pp. 1-12, 2000.

12. Mohammed Al-laham1 & Ibrahiem M. M. El Emary," Comparative Study Between Various Algorithms of Data Compression Techniques"
13. Jose Hernandez palancar et al,"A Compressed vertical Binary Algorithm for Mining Frequent Pattern"
14. D. I. Lin and Z. M. Kedem, "Pincer-search: an efficient algorithm for discovering the maximum frequent set," *IEEE Transactions on Knowledge and Data Engineering,* Vol. 14, pp. 553-566, 2002.
15. D. Xin, J. Han, X. Yan, and H. Cheng, "Mining Compressed Frequent-Pattern Sets," in *Proceedings of the 31st international conference on Very Large Data Bases*, pp. 709-720, 2005.
16. R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487-499, 1994
17. Rakesh Agarwal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A.Inkeri Verkamo,"Fast Discovery of Association Rules",inthe proceeding of IEEE.
18. D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, and T. Yiu, "MAFIA: A maximal frequent item set algorithm," *IEEE Transactions on Knowledge and Data Engineering,* Vol. 17, pp. 1490-1504, 2005.
19. Qian Wan and Aijun An,"Efficient Indirect Association Discovery using compact Transaction Database"
20. Yuh-Jiuan Tsay, Tain-Jung Hsu,Jing-Rung Yu," FIUT:A new method for mining frequent itemset",in the proceeding of information science,2009
21. Yudho Giri Sucahyo, Raj P.Gopalan ,"CT-ITL: Efficient Frequent Item Set Mining Using A Compressed Prefix Tree With Pattern Growth"
22. Yang-Liang Chen and Chin-Yuan Ho,"A Sampling Based Method for Mining Frequent Patterns From Database". in the proceedings of Fuzzy systems and knowledge discovery in second international Dr.K.Alagarsamy and A.Meenakshi," A Novelty Approach for Finding Frequent Itemsets in Horizontal and Vertical Layout- HVCFPMINETREE", International Journal of Computer Applications (0975 – 8887) Volume 10– No.5, November 2010.
23. Mingjun Song, and Sanguthevar Rajasekaran," A Transaction Mapping Algorithm for Frequent Itemsets Mining" ,in the proceeding of IEEE transactions on knowledge and data engineering.
24. M. Z. Ashrafi, D. Taniar, and K. Smith, "A Compress-Based Association Mining Algorithm for Large Dataset," in *Proceedings of International Conference on Computational Science*, pp. 978-987, 2003.
25. Jiawei Han, Jian Pei, Yiwen Yin, Runying Mao, "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach", *Microsoft Corporation* Data Mining and Knowledge Discovery, 8, 53–87, 2004.
26. Jiawei Han, Yongjian Fu, "Mining Multiple-Level Association Rules in Large Databases" IEEE transactions on knowledge and data engineering, vol. 11, no. 5, september/october 1999 .
27. Yoones Asgharzadeh Sekhavat*, M. Fathian, M.R. Gholamian and S. Alizadeh*," Mining important association rules based on the RFMD technique " *Int. J. Data Analysis Techniques and Strategies, Vol. 2, No. 1, 2010.*
28. R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487-499, 1994.
29. Savasere, E. Omiecinski, and S. Navathe, "An Efficient Algorithm for Mining Association Rules in Large Databases," in *Proceedings of the 21st International Conference on Very Large Data Bases*, pp. 432-444, 1995.
30. D. W. L. Cheung, S. D. Lee, and B. Kao, "A general incremental technique for maintaining discovered association rules," in Proceedings of the 15th International Conference on Database Systems for Advanced Applications, pp. 185-194, 1997.
31. LI Qingzhong, WANG Haiyang, YAN Zhongmin, MA Shaohan," .Efficient Mining of Association Rules by Reducing the Number of Passes over the Database[J] Journal of Computer Science and Technology, 2001,V16(2): 0-0.
32. Ashish Mangalampalli, Vikram Pudi," Fuzzy Association Rule Mining Algorithm for Fast and Efficient Performance on Very Large Datasets",In *IEEE International Conference on Fuzzy Systems*(*FUZZ-IEEE*) Jeju Island, Korea Report No: IIIT/TR/2009/173.

This page is intentionally left blank