A NOVEL TRANSACTION REDUCTION & DATA ELIMINATION BASED TECHNIQUE FOR MINING FREQUENT ITEM SETS FROM A TRANSACTION DATA BASE

[1]Rupesh Panwar, [2]Prof. Abhishek Raghuvanshi

Abstract: Data mining automates the process of finding predictive information in large databases. Questions that traditionally required extensive hands-on analysis can now be answered directly from the data — quickly. A typical example of a predictive problem is targeted marketing. In this paper, we have developed a method to discover large item sets from the transaction database. The proposed methodology uses an elimination based technique for frequent item set mining. The proposed method reduces the transaction which does not contain any frequent item in the initial steps of the frequent item set mining process. The proposed method eliminates the infrequent item sets from the transaction data base to convert it in to a reduced transaction data base. 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.

1. Introduction:

Database has been used in business management, government administration, scientific and engineering data management and many other important applications. The newly extracted information or knowledge may be applied to information management, query processing, process control, decision making and many other useful applications. With the explosive growth of data, mining information and knowledge from large databases has become one of the major challenges for data management and mining community.

KDD process

General steps involved in knowledge discovery:

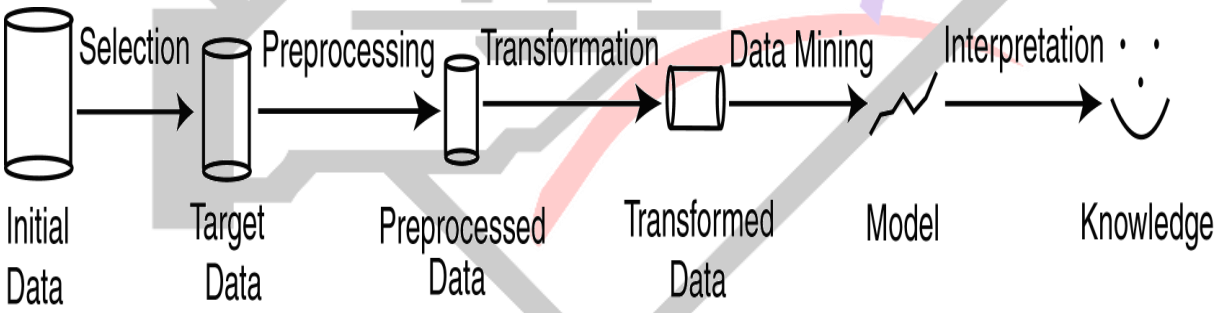

Figure 1 KDD Process

- **Selection**: Obtain data from various sources.
- **Preprocessing**: Cleanse data.
- **Transformation**: Convert to common format. Transform to new format.
- **Data Mining**: Obtain desired results.
- **Interpretation/Evaluation**: Present results to user in meaningful manner
- **Data visualization**: Generating graphs and charts for knowledge that is discovered.

The frequent itemset mining is motivated by problems such as market basket analysis [3]. A tuple in a market basket database is a set of items purchased by customer in a transaction. An association rule mined from market basket database states that if some items are purchased in transaction, then it is likely that some other items are purchased as well. Finding all such rules is valuable for guiding future sales promotions and store layout.

This is where the classic beer/diapers bought together analysis came from. It finds groupings. Basically, this technique finds relationships in product or customer or wherever you want to find associations in data. The process of grouping a set of physical or abstract objects into classes of similar object is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and dissimilar to the objects in other clusters. A cluster of data objects can be treated collectively as one group in many applications. Association analysis is the discovery of association rule showing

attribute-value conditions that occur frequently together in a given set of data. Association analysis is widely used for market based or transaction data analysis. Association rules identify collections of data attributes that are statistically related in the underlying data. An association rule is of the form X => Y where X and Y are disjoint conjunctions of attribute-value pairs. The confidence of the rule is the conditional probability of Y given X, Pr(Y|X), and the support of the rule is the prior probability of X and Y, Pr(X and Y). Here probability is taken to be the observed frequency in the data set**.**

2. Related Work:

This algorithm [1] also used to reduce the number of database scan. It is based upon the downward disclosure property in which adds the candidate itemsets at different point of time during the scan. In this dynamic blocks are formed from the database marked by start points and unlike the previous techniques of Apriori it dynamically changes the sets of candidates during the database scan. But it generates the large number of candidates and computing their frequencies are the bottleneck of performance while the database scans only take a small part of runtime assumption [4, 5].

It was absorbed in [7] [5] that the improved algorithm is based on the combination of forward scan and reverse scan of a given database. If certain conditions are satisfied, the improved algorithm can greatly reduce the iteration, scanning times required for the discovery of candidate itemsets.

COFI tree [8] generation is depends upon the FP-tree however the only difference is that in COFI tree the links in FP-tree is bidirectional that allow bottom up scanning as well [2,9]. The relatively small tree for each frequent item in the header table of FP-tree is built known as COFI trees [9]. Then after pruning mine the each small tree independently which minimise the candidacy generation and no need to build he conditional sub-trees recursively. At any time only one COFI tree is present in the main memory thus in this way it overcome the limitations of classic FP-tree which can not fit into main memory and has memory problem.

COFI tree is based upon the new anti-monotone property called global frequent/local non frequent property [10].

Compress tree structure is also the prefix tree in which all the items are stored in the descending order of the frequency with the field index, frequency, pointer, item-id [10]. The CT-PRO uses the compact data structure known as CFP-tree i e. compact frequent pattern tree so that all the items of the transactions can be represented in the main memory [10].

H-mine [3] algorithm is the improvement over FP-tree algorithm as in H-mine projected database is created using in-memory pointers. H-mine uses an H-struct new data structure for mining purpose known as hyperlinked structure. For the large databases, first in partition the database then mine each partition in main memory using H-struct then consolidating global frequent pattern [3]. If the database is dense then it integrates with FP-Growth dynamically by detecting the swapping condition and constructing the FPtree. This working ensures that it is scalable for both large and medium size databases and for both sparse and dense datasets [6].

3. Problem Specification

The concept of frequent itemset mining was first introduced for mining transaction databases. Let I = {I1, I2, . . . , In} be a set of all items. Also A k-itemset α which consists of k items from I is frequent if α occurs in a transaction database D no lower than θ |D| times where θ is a user-specified minimum support threshold (called min_sup) and |D| is the total number of transactions in D.

4. Proposed Algorithm:

STEP 1: START

STEP 2: INPUT TRANSACTION DATA SET & MINIMUM SUPPORT THRESHOLD

STEP 3: FIRST THE PROPOSED ALGORITHM SCANS THE TRANSACTION DATA BASE AND CALULATES THE SUPPORT OF EACH SINGLE SIZE ITEM.

STEP 4: IN THIS STEP A LIST OF FREQUENT ITEM AND INFREQUENT ITEM IS PREPARED ON THE BASIS OF MINIMUM SUPPORT THRESHOLD.

IF AN ITEM IS HAVING SUPPORT GREATER THAN THE MINIMUM SUPPORT THRESHOLD THEN ITEM IS PLACED IN FREQUENT ITEM LIST AND ALSO IN EXPANSION LIST. OTHERWISE IT IS PLACED IN INFREQUENT ITEM LIST

STEP 5: REMOVE THE TRANSACTION WHICH DOES NOT CONTAIN ANY FREQUENT ITEM

STEP 6: IN THIS STEP, ALL THE MEMBERS OF THE INFREQUENT ITEM LIST ARE REMOVED FROM THE TRANSACTION DATA BASE BECAUSE THEY WILL NOT APPEAR IN ANY FREQUENT ITEM SET. IN THIS WAY, THE ORIGINAL TRANSACTION DATA BASE IS CONVERTED INTO REDUCED SIZE DATA BASE. NOW THIS REDUCED DATA BASE WILL BE USED IN THE CALCULATION OF LARGER SIZE FREQUENT ITEM SETS.

STEP 7: WHILE EXPANSION LIST IS NOT EMPTY

- **PERFORM LEFT EXPANSION OF SMALLER SIZE ITEMS TO GENERATE HIGHER SIZE ITEMS AND THEN REPEAT STEP 4 FOR THEM**
- **PERFORM RIGHT EXPANSION OF ELEMENTS AND THEN REPEAT STEP 4 FOR THEM.**

STEP 8: WRITE THE LIST OF FREQUENT ITEM SETS

STEP 9: STOP

Conclusion:

Data warehousing represents an ideal vision of maintaining a central repository of all organizational data. Centralization of data is needed to maximize user access and analysis. Dramatic technological advances are making this vision a reality for many companies and equally dramatic advances in data analysis software are allowing users to access this data freely. The data analysis software is what supports data mining. This paper presents a novel reduction and elimination based frequent item set mining technique. This proposed method is time and memory efficient.

References:

[1] 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.
[2] C. Borgelt. "An Implementation of the FP- growth Algorithm". Proc. Workshop Open Software for Data Mining, 1–5.ACMPress, New York, NY, USA 2005.
[3] 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.
[4] 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.
[5] "Data mining Concepts and Techniques" by By Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, 2006.
[6] 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.
[7] 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.
[8] 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.
[9] 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.
[10] 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.