Improved Positive and Negative Quantitative Association Rule Mining using Sam

P Sundari[1] K Arulkumar[2]
[1]Assistant Professor [2]Research Scholar
[1,2]Department of Computer Science
[1,2]Govt. Arts College, Coimbatore, India

Abstract— Mining and discovering association from huge dataset is one of the common data mining technique, which help to extract interesting knowledge and find dependencies between items in the dataset. Several techniques and algorithms have been proposed to find dependencies between items with positive dependencies and those techniques don't concentrate on negative dependency calculation. Certain algorithms initiated the findings of negative association rules, even though the techniques are effective, that is only considered the quality in rules. So there is a need of finding both positive and negative quantitative association rules from the dataset. This paper proposes a new technique named as Improved Positive and Negative Quantitative Association Rule Mining using SaM(Split and Merge). It is a new multi-objective based algorithm, which helps to mine a decreased set of positive and negative quantitative association rules rapidly. In addition, this proposal maximizes the following objectives such as improving precision, interestingness, performance and reducing the storage overhead. This also includes the split and merge algorithm for fast data management. in order to obtain set of rules which are interesting, easy to understand, suitable for decision making and provide good coverage of the dataset. The effectiveness of the proposed approach is validated over several real-world datasets.

Key words: Data Mining, Association Rule Mining, Positive and Negative Rules, Frequent and Infrequent Itemsets

I. INTRODUCTION

Association rule mining is a method to identify the hidden facts in large instances database and draw interferences on how subsets of items influence the existence of other subsets. Association rule mining aims to discover strong or interesting rule and week rule from the transactional database. Association rule mining is the process of finding the association rules that satisfy the predefined minimum support and confidence from a given database [1][4]. Association rule mining finds an interesting relations and connections along with large set of data items. Association rules show attribute value conditions that occur frequently together in a given dataset with help of candidate generation. A typical and widely-used example of association rule mining is Market Basket Analysis.

A. Item set

Item set is a non-empty set of elements. The item set ranges from one to any positive number. Each transaction record contains an item set of size x. A collection of transaction records is a dataset. An item set that can be observed repeatedly in a set of transaction records is typically called a frequent item set. If a threshold is set to differential the frequency of occurrences, then item sets that are observed below the given threshold are called infrequent item sets. Both frequent and infrequent item sets can be observed over a set of transaction records, they indicate the presence of item sets.

B. Association Rule Definition

The basic definition of association rule states that Let I={i1,i2,i3,.....,in} be a set of items and T is the transactional database where t is the set of items of each transaction, An association rule is a rule relating an antecedent item set X to another consequence item set Y such that $X \Rightarrow Y$, where $X \subseteq I$, $Y \subseteq I$ and $X \cap Y = \emptyset$[2]. Assuming that the user has defined a minimum support, ms, for $X \Rightarrow Y$, all association rules found will have support values of at least ms.

C. Support and Confidence Framework

A support and confidence framework introduced by Agrawal, utilizes the measure of interestingness support and confidence to define and identify strong association rules from a dataset[2]. The algorithm discovers all association rules that have strength values with at least a minimum support and a minimum confidence threshold[8].

$$Supp(X \Rightarrow Y) = \frac{num\,of\ tuples\ containing\ \ both\ X\ and\ Y}{total\ number\ of\ transaction}$$

$$Conf(X \Rightarrow Y) = \frac{num\,of\ tuples\ containing\ \ X\ and\ Y}{num\,of\ tuples\ containing\ \ X}$$

D. Association Rules: Positive and Negative Association Rules

An association rule can be further distinguished as a type of positive or negative association rule. All association rules are typically positive association rules as defined in the beginning of this section. They relate the items that can be observed in a dataset [10]. On the other hand, any association rule involving the absence of an item set X or Y is called a negative association rule. For example, $X \Rightarrow \neg Y$, $\neg X \Rightarrow Y$, and $\neg X \Rightarrow \neg Y$, where the symbol '¬' denotes the absence of an item set in a set of transaction records [7].

II. RELATED WORK

The PNAR [Positive Negative Association Rule Mining] is for mining both positive and negative association rules in databases. The algorithm extends traditional association rules to include quantity of these items. When mining negative association rules, author use same minimum support threshold to mine frequent negative itemsets. Traditional algorithms for mining association rules are built on the binary attributes databases. The aim of paper [8] is to develop a new model for mining interesting positive and negative association rules out of a transactional data set. The proposed model is Improved Positive and Negative Association Rule Mining using SaM [3], for mining both

negative and positive association rules from the interesting frequent and infrequent itemsets.

III. PROBLEM STATEMENT

Owing the continuous development in the usage of computers in all places, several databases are being constantly generated. However, there is no effective technique to utilize these databases efficiently and to find the valuable associations in between them. Association rule mining finds interesting association or correlation relationships among a large amount of data items. With huge amounts of data constantly being collected and accumulated, many industries and supermarkets are showing interest in mining association from this large collection of business transaction records, as it can assist in many business decision making processes, such as catalog design, cross marketing and others. In the processing, the split step stays the same, but now it only yields an intermediate database with all transactions (or suffixes) that actually contain the split item under consideration. In order to form the full conditional database, we have to add those transactions that do not contain the split item, but can be made to contain it by inserting it[9]. The pervious split and merge algorithm do not generate the positive and negative rule. The proposed split and merge algorithm can easily generate these rules. In this paper, both frequent and infrequent itemsets are used to mine both the positive and negative association rules from frequent and infrequent itemsets. It minimizes I/O overhead by scanning the database only once [7].

IV. PROPOSED WORK

Consider a transactional database which consists of set of transactions with their transaction id and list of items in the transaction. Then scan the entire database. Collect the count of the items present in the database. Then sort the items in decreasing order based on their frequencies.

A. Transactional Database:

The first step is the process of generating transactional dataset. The following is the transactional list taken for the experiments.

ID	Transactions-items
1	A,B
2	A,B,C
3	B,C
4	B,C,D
5	B,C,D
6	B,C,D
7	B,C,D
8	B,D
9	H
10	A,H
11	A
12	A
13	A
14	A
15	A,B,C
16	A,C
17	A,C,D
18	G
19	I

Fig. 1: transactional database sample

B. Results of SaM in Each Step for Frequent Items:

ID	Transactions-items
1	A,B
2	A,B,C
3	B,C
4	B,C,D
5	B,C,D
6	B,C,D
7	B,C,D
8	B,D
9	H
10	A,H
11	A
12	A
13	A
14	A
15	A,B,C
16	A,C
17	A,C,D
18	G
19	I

(a)

Item	Frequency
A	10
B	9
C	9
D	6
H	2
I	1
G	1

(b)

SaM: Process in original dataset

Transaction_items
A
A
A
A
A,B
A,B,C
A,B,C
A,C
A,C,D
A,H
B,C
B,C,D
B,C,D
B,C,D
B,C,D
B,D

(c)

Transaction_items
A
A
A
A
A,B
A,B,C
A,B,C
A,C
A,C,D
B,C
B,C,D
B,C,D
B,C,D
B,C,D
B,D

(d)

Fig 2: (a) SaM: Steps (a): original dataset (b) items and its frequency displayed by descending order (c) sorted items lexicographically (d) sorted items based on the threshold

C. Frequency Calculation:

After the successful data collection, the system performs the frequency calculation process, which identifies and shows the number of occurrences of each item and each transaction.

items	frequency
A	4
A,B	1
A,B,C	2
A,C	1
A,C,D	1
A,H	1
B,C	1
B,C,D	4
B,D	1
G	1
H	1
I	1

Item	Frequency
A	10
B	9
C	9
D	6
H	2
I	1
G	1

Fig. 3 Frequency calculation for item and itemset

The above fig 3 shows the items and its frequency from transactional database. Additionally the system uses SaM algorithm based on this frequency values. This is not filtered based on the threshold. It displays all items and its frequency. The SaM algorithm finds the most appropriate

item in the transaction database system. The steps are illustrated in Figure 1 for a simple example transaction database.

- Step 1: Shows the transaction database in its original form.
- Step 2: The frequencies of individual items are determined from this input in order to be able to store infrequent items into the hash table immediately. Assume a minimum support of transactions values.
- Step 3: The (frequent and infrequent) items in each transaction are sorted according to their frequency in the transaction database, since it is well known that processing the items in the order of increasing frequency usually leads to the shortest execution times. For frequency calculation the algorithm reads the specific index and for infrequent it reads negative index.
- Step 4: The transactions are sorted lexicographically into ascending order initially for frequent items and descending for infrequent items, with item comparisons again being decided by the item frequencies.

1) The Positive and negative itemset mining is explained in following steps.

Algorithm: Positive and Negative Association Rules
Input: TDB-Transactional Database
MS-Minimum Support
MC-Minimum Confidence
N-number of transactions
F1-frequent item sets
F-frequency
I - Itemsets
Output: Positive and Negative Association Rules
Method:
1. P← Φ , N←Φ
2. Find F1← Set of frequent 1- itemsets
3. for (k=2;Fk-1!=Φ; k++)
4. {
5. Ck= Fk-1 join Fk-1
6. // Prune using FP Property
7. for each i ε Ck, any subset of i is not in Fk-1 then Ck = CK - { i }
8. for each i ε Ck find Support(i)
{
9. for each A,B (A U B= i)
10. {
11. QA,B= Association(A,B);
15. if Q<0
16. {
17. if(supp(A←¬ B)≥MS&&conf(A←¬ B)≥MC) then
N←N U { A← ¬ B}
19. if(supp(¬ A⟶B)≥MS&&conf(¬ A⟶B)≥MC) then
20. N←N U { ¬ A← B}
21. }
23. }
24. AR← P U N

V. Experimental Results

We conduct experiments on a different transaction size and differing number of transaction in a database to compare our approach with the PNARM. The execution time with different minimum support for the dataset is shown in the figure 4. The execution time with different dataset sizes is for the fixed minimum support. It can be observed that both the methods generate equal number of positive and negative association rules, but the proposed approach reduce the execution time over the existing method.

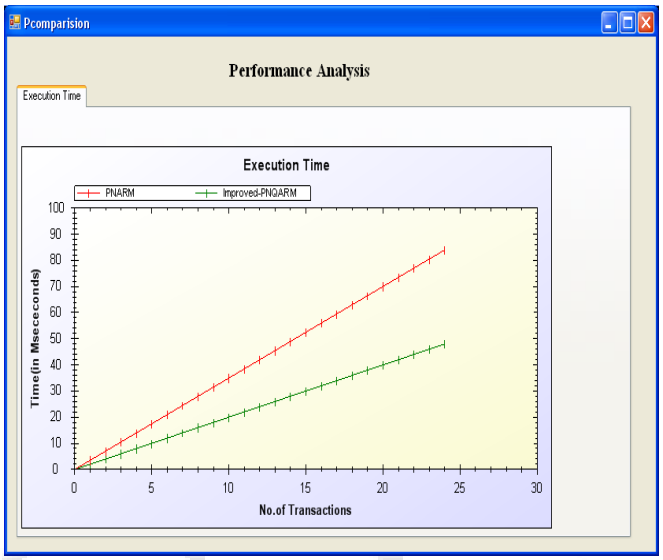

Fig 4: Execution time

VI. Conclusion

The system proposed frequent data set and infrequent data set and its associations in the high dimensional transactional dataset. Positive and negative rule mining from a transactional database helps to discover the items with high utility as well as low utility based on different parameter have been proposed. The new algorithm named Improved PNQARM includes one algorithm SaM structure handling queries. This improves the performance of existing FP growth with the SaM implementation. The proposed algorithm reduces the number of candidates and database scans effectively. The experiments' and results shows the proposed system performs better than existing system.

Reference

[1] Agrawal R, Imielinski T, Swami A, "Mining Association Rules between sets of items in large databases". In Proc of the 1993ACM on Management of Data. Washington, D C, May 1993, 207-216.
[2] Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules in large databases. Proceedings of the 20th International Conference on Very Large Data Bases. pp. 487-499.
[3] Christian Borgelt and Xiaomeng Wang "SaM: A Split and Merge Algorithm for Fuzzy Frequent Item Set Mining"Otto-von-Guericke-University of Magdeburg Universit¨atsplatz 2, 39106 Magdeburg, Germany.
[4] Dhanabhakyam. M and Punithavalli.M. A Survey on Data Mining Algorithm for Market Basket Analysis. Global Journal of Computer Science and Technology, Vol. 11 issue 11, version 1.0, 2011.

[5] Idheba Mohamad Ali O,Swesi ,Azuraliza Abu Bakar, Anis Suhailis Abdul Kadir "Mining Possitive and Negative Association Rules from Interesting Frequent and Infrequent Itemsets"978-1-4673-0024/10/@26 @2012.
[6] Jiawei Han, Micheline Kamber, Jain Pei, Han Kamber Pei, "Data Mining Concepts and Techniques" Murgan Kaufmann Publishers an imprint of Elsevier © 2012.
[7] Karthikeyan. K 1 and N. Ravikumar2, "A Survey on Association Rule Mining", Associate Professor, Department of Computer Science, PSG College of Arts and Science, Coimbatore, India. IJARCCE Vol. 3, Issue 1, January 2014.
[8] Rezbaul Islam. A.B.M., Tae-Sun Chung" An Improved Frequent Pattern Tree Based Association Rule Mining Technique" 978-1-4244-9224-4/11/$26.00 ©IEEE 2011.
[9] Swesi, Idheba Mohamad Ali O., Afarulrazi Abu Bakar, and Anis Suhailis Abdul Kadir. "Mining positive and negative association rules from interesting frequent and infrequent itemsets." Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on. IEEE, 2012.
[10] Wu, Xindong, Chengqi Zhang, and Shichao Zhang. "Efficient mining of both positive and negative association rules." ACM Transactions on Information Systems (TOIS) 22.3 (2004): 381-405.

