Association Rule Mining with Hybrid-Dimension Datasets

Priyanka Pawar*
Sachin Deshpande*
Vipul Dalal*

Abstract

Hybrid dimension association rules mining algorithm satisfies the definite condition on the basis of multidimensional transaction database. Boolean Matrix based approach has been employed to generate frequent item sets in multidimensional transaction databases. When using this algorithm first time, it scans the database once and will generate the association rules. Apriori property is used in algorithm to prune the item sets. It is not necessary to scan the database again; it uses Boolean logical operations to generate the association rules.

Keywords: Association Rule, Hybrid Dimensional Association Rule, Relational Calculus, Multidimensional Transaction Database

1. Introduction

For mining association rule in transactional or relational database in data mining till now we have used different approaches. Apriori algorithm is costly to handle a huge number of candidate sets and it requires multiple scans for the database which is a tedious job. However, in situations with a large number of frequent patterns, long patterns, or quite low minimum support thresholds, an Apriori-like algorithm may suffer from some above problems and it is used for only single dimensional mining. Although an FP-tree is rather compact, its construction needs two scans of a transaction database, which may represent a nontrivial overhead [3].

Finding frequent patterns plays an important role in data mining and knowledge discovery techniques. Association rule describes correlation between data items in large databases or datasets. The first and foremost algorithm to find frequent pattern was presented by R. Agrawal et al. in 1993. Presented frequent pattern tree approach, for mining association rules without candidate generation. The candidate generation and test methodology called Apriori techniques was the first technique to compute frequent patterns based on the Apriori principle and anti-monotone property. The Apriori technique finds the

Computer Department, Vidyalankar Institute of Technology, Wadala, Mumbai, India

frequent pattern of length k from the set of already generated candidate patterns of length k-1. This algorithm requires multiple database scans and large amount of memory to handle the candidate patterns when the number of potential frequent pattern is reasonably large. In the past two decades, large numbers of research studies have been published presenting new algorithms or extending existing algorithms to solve frequent pattern mining problem more effectively and efficiently. But all the above-mentioned studies are well suitable for single-dimensional transactional databases.

2. Association Rules

Definition 1: Let I = {i1, i2, i3,……. in} be a set of items. D is a database of transactions. Each transaction T is a set of items and has an identifier called TID. Each $T \subseteq I$.

Definition 2: Association rule is the implication of the form $A \Rightarrow B$, where A and B are item sets which satisfies $A \subseteq I$, $B \subseteq I$ and $A \cap B = \phi$.

Definition 3: The strength of an association rule can be measured in terms of its Support and Confidence.

The support supp(X) of an item set X is defined as the proportion of transactions in the data set which contain the item set.

The confidence of a rule is defined

$$Conf(X \Rightarrow Y) = supp(X \cap Y)/supp(X)$$

Definition 4: Boolean Matrix: is a matrix with element '0' or '1'.

Definition 5: The Boolean AND operation is defined as follows: 0.0=0 0.1=0 1.0=0 1.1=1. Where logical implication is denoted by '.' or AND

There are different methods for generating frequent item sets and association rule mining.

Some of them are as follows:-

A. Apriori Algorithm

The classical Apriori algorithm employs an iterative method to find all the frequent item-sets. First, the frequent 1- item sets L_1 is found according to the user-specified minimum support threshold and then the L_1 is used to find frequent 2-iemsets L_2, and so on, until there is no new frequent item sets could be found. After finding all the frequent item sets using Apriori, we could generate the corresponding association rules [5]. Apriori employs an iterative approach known as a level-wise search, where k-item sets are used to explore (*k*+1)-item sets. Apriori principle: If an item set is frequent, then all of its subsets must also be frequent. It works in two steps-Join Step: Ck is generated by joining L*k-1*with itself. Prune Step: Any (k-1)-item set that is not frequent cannot be a subset of a frequent k-item set.

Apriori Algorithm is the simple Single-dimensional mining algorithm.

B. Sampling Algorithm

The main idea for the sampling algorithm is to select small sample one that fits in the main memory of the database of transactions and to determine the frequent item sets from that sample. If those frequent item sets form a superset of frequent item sets for the entire database, then we can determine the real frequent item sets by scanning the remainder of the database in order to compute exact support values for the superset item sets. A superset of frequent item sets can usually be found from by using for eg.Apriori algorithm with a lowered minimum support.

C. Partition Algorithm

In this algorithm if we are given a database with a small number of potential large item sets say a few thousands, then support for them can be tested in one scan by using a partitioning technique. Partitioning divides the database into non-overlapping subsets; these are individually considered as separate databases and all large item sets for that partition called local frequent item sets, are generated in one pass. The Apriori algorithm can then be used efficiently on each partition if it fits entirely in main memory. Partitions are chosen in such a way that each partition can be accommodated in main memory.

D. FP-growth algorithm

FP-growth algorithm is an efficient method of mining all frequent item sets without candidate's generation. The algorithm mine the frequent item sets by using a divide-and-conquer strategy as follows: FP-growth first compresses the database representing frequent item set into a frequent-pattern tree, or FP-tree, which retains the item set association information as well. The next step is to divide a compressed database into set of conditional databases (a special kind of projected database), each associated with one frequent item. Finally, mine each such database separately. Particularly, the construction of FP-tree and the mining of FP-tree are the main steps in FP-growth algorithm.

In reality, for example, along with items purchased in sales transactional databases, other related information like quantity purchased, price, branch location etc are stored. Additional related information regarding the customers who purchased the items, such as customer age, occupation, credit rating, income, and address also stored in the database. Frequent item sets along with other relevant information will be helpful in high-level decision-making. This leads to the challenging mining task of multilevel and multidimensional association rule mining. In recent years, there has been lot of interest in mining databases with multidimensional data values.

3. Conditional - Hybrid Dimensional Association Rule Mining

Thus here I present mining conditional hybrid-dimensional association rules. Based on these marking, either it does intra-dimensional join or inter-dimensional join.

To solve these problems for founding frequent item sets we have proposed this algorithm. It mines hybrid dimension Association rules not only from single-dimensional as well as multidimensional database. It meets the definite condition to generate conditional hybrid dimensional association rules, from multidimensional transactional database. It scans database only once which makes easy to find large frequent patterns. It does not generate the candidate item sets as we generate in Apriori

algorithm, rather it uses Boolean Vector "relational calculus" to generate frequent item sets. I take multidimensional datasets with five attribute as input and apply on Hybrid-Dimensional Association Algorithm Rule to generate association rule using Boolean Matrix. I use backend Sql Server and front end jdk1.5 version.

Methodology used in this project:-

- Transforming the multidimensional transaction database into two Boolean matrices one for subordinate attributes (Am*p) and one for main attribute (Am*q).
- Generating the set of frequent 1-itemset L_{A1} (from the subordinate attributes matrix) and L_{B1} (from the main attribute matrix).
- Pruning the Boolean matrices.
- Perform AND operations to generate 2-itemsets: $L_{A1} \bowtie L_{B1}$ and $L_{A1} \bowtie L_{A1}$ for inter-dimension join and $L_{B1} \bowtie L_{B1}$ for intra-dimension join.
- Repeat the process to generate (k+1)-item sets from L_k.

Transforming the multidimensional transaction database into Boolean matrix
Generating the frequent 1-itemset L_1
Pruning the Boolean matrix
Generating the set of frequent k-item sets L_k

The generation of frequent item sets is the core of all the association rules mining algorithms. Previous studies on mining multi-dimensional association rules we focused on finding non-repetitive predicate multi-dimensional rules. We integrate the single-dimensional mining and no repetitive predicate multi-dimensional mining, and present a method for mining hybrid-dimensional association rules using Boolean Matrix.

A. The Join Process

There are two steps in generation of the frequent item sets and frequent predicate sets. The two steps are *joining* and *pruning*.

- The join generating candidate 2-itemsets C2; we find frequent 1-itemsest based on each attribute, at the same time we mark items belong to every main attribute. So it will be clear that the marked items are the items of main attribute and unmarked items are the subordinate items. When we search for C2, if both of the two joining items are marked items, we call the function for intra-dimensional join between the items as well as inter-dimensional join, but only proceed with inter-dimensional join on the other occasions.
- The join on other occasions when we generate frequent item sets directly according to the join mode of the Apriori, it would occur intradimensional join as well as inter-dimensional join. But there are some restrictions to the generation of intradimensional join and inter-dimensional join. Therefore we make the following modifications to the joining step of the Apriori. We assume that items within transaction and item-set are sorted in lexicographic order. We could take two steps to find L_k.
- Distinguish the intra-dimensional join and inter-dimensional join; If all the items within the two (k-1) item-sets belong to the main attribute we proceed with intra-dimensional join and proceed with inter-dimensional join on other occasions.
- Implement join $L_{k-1} \bowtie L_{k-1}$, and choose the corresponding joining condition according to the characteristic of the join (intra-dimensional join or inter-dimensional join)

B. The Conditional restriction in hybrid-dimension

Association rules

First the frequent item-sets are obtained, and then we generate the hybrid-dimension association rules from the frequent item-sets. In the process of generating frequent item-sets, we make both intra-dimensional join and inter-dimensional join as well as the conditional restrictions while proceeding with join, all of the frequent item-sets have such a character: the values within main attribute field occur many times while the values within subordinate attribute fields occur only once. Thus, the rules generated by the algorithm may include many predicates or include the same predicate. So the hybrid dimension association rules are formed [1].

4. Algorithm

The algorithm consists of following steps:

1. Transforming the multidimensional transaction database into two Boolean matrices one for subordinate attributes (Am*p) and one for main attribute (Am*q).
2. Generating the set of frequent 1-itemset L_{A1} (from the subordinate attributes matrix) and L_{B1} (from the main attribute matrix).
3. Pruning the Boolean matrices.
4. Perform AND operations to generate 2-itemsets: L_{A1} join L_{B1} and L_{A1} join L_{A1} for inter-dimension join And L_{B1} join L_{B1} for intra-dimension join.
5. Repeat the process to generate (k+1)-item-sets from L_k
   - Transforming the multidimensional transaction database into Boolean matrix
   - Generating the frequent 1-itemset L_1
   - Pruning the Boolean matrix
   - Generating the set of frequent k-item sets L_k

We integrate the single-dimensional mining and no repetitive predicate multi-dimensional mining and present a method for mining hybrid-dimensional association rules using Boolean Matrix. Let a multi-dimensional transaction database order, which includes two subordinate attributes Age and Income and one main attribute Ordered_items as given in table I. In order to simplify the implement process, we pre-processed some attributes before algorithm executes shown below in table II and table III.

The multidimensional transaction table order is transformed into two Boolean Matrices: Am*p as subordinate attributes matrix and Bm*q as main attribute matrix which are as given below: Let the minimum support is 0.4; m=10 is the number of transactions.

Table I: Order

ID	Age	Income	Ordered_items
1	31..40	6780	I1, I2, I5
2	31..40	7800	I1, I2
3	31..40	9500	I1,I2,I5
4	21..30	4850	I2, I4
5	41..50	7700	I1, I3
6	31..40	8650	I1, I2, I4
7	31..50	3500	I1, I3, I5
8	21..30	4600	I2, I5
9	21..30	3950	I1, I2, I3
10	21..40	5400	I3, I4

Table II: Mapping Age

Interval	Name
21..30	y
31..40	m
41..50	s

Table III: Mapping Income

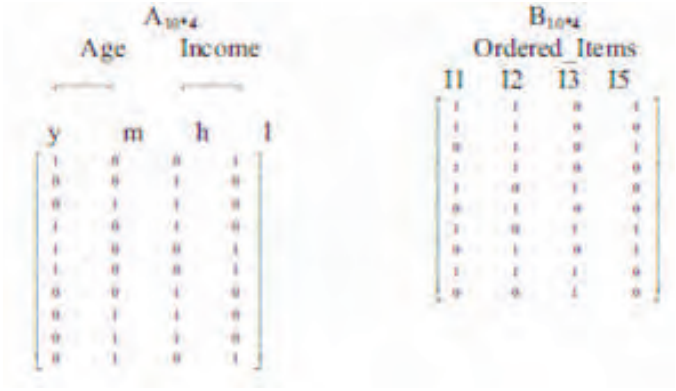
Interval	Name
4000-6000	l
6000-10,000	h

Table IV: Order Sets

ID	Age	Income	Ordered_items
1	M	H	I1, I2, I5
2	M	H	I1, I2
3	M	H	I1,I2,I5
4	Y	L	I2, I4
5	S	H	I1, I3
6	M	H	I1, I2, I4
7	M	L	I1, I3, I5
8	Y	L	I2, I5
9	Y	L	I1, I2, I3
10	Y	L	I3, I4

Therefore min_sup_num=10. We compute the sum of the elements value of each column in the Boolean matrix A_{10*5} and B_{10*5} set of frequent 1-itemset is:

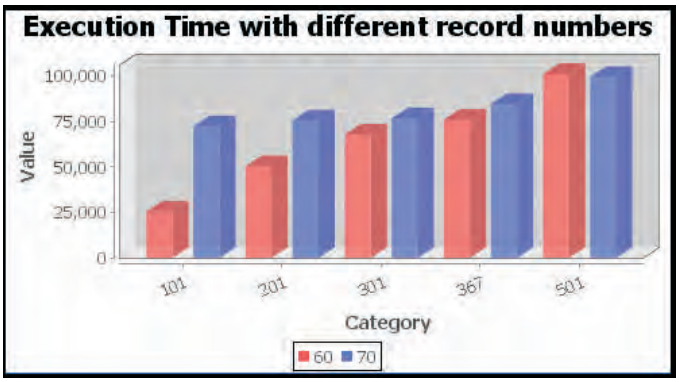
L_{A1} = {{y},{m},{h},{l}}, L_{B1} = {{I1},{I2},{I3},{I5}} smaller than the minimum support number [7]. Now we perform the 'AND' operation to join L_{A1} and L_{B1} (according to the type of join) to generate L_2. The possible 2-itemsets are: Inter-dimensional join ($L_{A1} \bowtie | L_{B1}$ and $L_{A1} \bowtie | L_{A1}$): It is performed by AND operation among the columns of Matrix A_{m*p} AND B_{m*q} and A_{m*p} AND A_{m*p}. Intra-dimensional join ($L_{B1} \bowtie L_{B1}$): It is performed by AND operation among the columns of Matrix B_{m*p} AND B_{m*q}. The possible 2-item sets from L_{A1} and L_{B1} are: (y,l), (m,h), (h,1), (h,2), (h,3), (h,5), (l,1), (l,2), (l,3), (l,5), (y,1), (y,2), (y,3), (y,5), (m,1), (m,2), (m,3), (m,5), (I1,I2), (I1,I3), (I1,I5), (I2,I3), (I2,I5), (I3,I5). After performing 'AND' operation to get the support numbers of these mentioned item sets the Boolean matrices A_{10*18} and B_{10*6} are generated. Now again we compute the sum of the columns of matrices A_{10*18} and B_{10*6}. And prune the columns of the 2-itemsets those are not frequent. Same process will be repeated till for next higher item sets.



We can generate such a hybrid-dimension association rule: $m \cap h \cap I1 \Rightarrow I\,2$ (Support=40% and Confidence=100%)

5. Experiment

To test whether the proposed method is fast, expansible and effective our experiments are made on machine with Intel (R) Core 2Duo, 1.5GHz and 1GB memory. The Operating system is Windows XP. We use a database that has 500 records and 13 attributes, which have 2~8 different value. Time value for execution is given in millisecond.



6. Result and Discussion

The confidence of association rules has a specific meaning: When the antecedent of the rule is satisfied, the consequent of the rule will have *c*% (here c refers to the confidence of the rule) possibility of being satisfied. In association rules, only in the antecedent part of multidimensional association rules include several predictions at the same time. We can say that the result of prediction on multidimensional association rules is better and more precise than on Single Dimensional Association Rules.

For example, Table I presents a multidimensional transaction database ***Order***. If we make a single dimensional association analysis on the predicate ***Ordered_items***, which presents itemsets. A in transaction, the result of analysis will only include the relevance of Order itemsets A. But, if we make a hybrid dimension association analysis, the result of analyzing not only includes the relevance of Order itemsets A, but also includes the relevance of customers' information, e.g.: ***Age, Income***. Thus when we proceed with predictions on the product order of customers' by means of the result of association analysis, obviously the conditions included in the antecedent of multidimensional association rules is more abundant, and will bring better prediction result.

7. Conclusion

The proposed algorithm uses input datasets and meets the definite condition to generate conditional-hybrid dimensional association rules, from multidimensional transactional database. The main features are: it scans the database only once, it does not generate the candidate item sets and it uses the "relational calculus" to generate frequent item sets. It stores data in the form of bits, so it needs less memory space and can be applied to large relational databases.

8. References

1. *R.Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," Proc. Int'l Conf. Very Large Data Base s, pp. 487-499, Sept. 1994.*
2. *S. Bashir, A. Rauf Baig, "Hybrid Miner: Mining Maximal Frequent Itemsets.A using Hybrid Database Representation Approach", In Proc. of 10th IEEE-INMIC conference, Karachi, P Pakistan, 2005.*
3. *D. Burdick, M. Calimlim, and J. Gehrke, "Mafia: A maximal frequent itemsets.A algorithm for transactional databases", In Proc. of ICDE Conf, pp. 443-452, 2001.*
4. *Proc. IEEE ICDM Workshop Frequent Item set Mining Implementations, B. Goethals and M.J. Zaki, eds., CEUR Workshop Proc., vol. 80, Nov. 2003.*
5. *K. Gouda and M. J. Zaki, "Efficiently mining maximal frequent itemsets.A", In ICDM, pp. 163–170, 2001.*
6. *Hunbing Liu and Baishen wang, "An association Rule Mining Algorithm Based On a Boolean Matrix" , Data Science Journal, Volume 6, Supplement 9, S559-563, September 2007.*
7. *Jurgen M. Jams Fakultat fur Wirtschafts- irnd, "An Enhanced Apriori Algorithm for Mining Multidimensional Association Rules, 25th Int. Conf. Information Technology interfaces ITI Cavtat, Croatia (1994).*
8. *R.Agrawal, H.Mannila, R.Srikant, H.Toivone and A.I.Verkamo. Fast discovery of association rules. In U.M. Fayyed, G.Piatetsky-Sharpiro, P.Smyth, and R.Uthurusamy, editors, Advances in knowledge Discovery and Data Mining, pages 307-328.AAAI/MIT press, 1996.*
9. *H. Mannila, H. Toivonen, and A. Verkamo. "Efficient algorithm for discovering association rules". AAA1 Workshop on Knowledge Discovery in Databases.*
10. *Jiawei Han, Micheline Kamber, "Data Mining Concepts and Techniques". Higher Education Press, 2001.*