A Recent Overview: Rare Association Rule Mining

Urvi Y. Bhatt and Pratik A. Patel
Department of Computer Science and Engineering,
Parul Institute of Technology, Waghodia, Vadodara, 391760,
India.

***Abstract*- Rare association rule mining is extract useful information for large database. Traditional association mining techniques generate frequent rules based on frequent item sets with reference to two user defined threshold minsup and minconf. It called as support-confidence framework. As many of generated rules are of no use, further analysis is essential to find interesting Rules. Rare association rule is the type of association rule that contains Rare Items. Rare Association Rules Represent unpredictable or unknown association, so it is more interesting than frequent association rule mining. The main goal of rare association rule mining is to discover relationships among sets of items in a database that occurs uncommonly. This paper presents a survey on the current trends and approaches in the area of rare association rule mining.**

I. INTRODUCTION

Data mining is the process of finding correlations, patterns, trends, or relationships by from a large amount of data stored in repositories, corporate databases, and data warehouses. It represents techniques for discovering knowledge patterns hidden in large databases. For extract interesting knowledge, several data mining techniques are being used. Like association rule mining techniques find out association between the entities, clustering techniques group the unlabeled data into clusters such that there exists Maximum inter similarity and minimum intra similarity between the clusters, classification techniques to identify the different classes existing in categorical labeled data. Mining of frequent itemsets from database using Association Rule mining is the major task of Data Mining field. Major Researcher focused on extraction of frequent patterns in association rule mining. The requirements of reliable rules that do not frequently appear are taking an increasing interest in a great number of areas [1].

Association rule mining is an essential data mining technique to find interesting associations between the entities (or items) in a database. It is used to extract useful information from database. This technique is proposed by Agrawal et.al. In 1993. Association rules are an important category of consistency and predictability that exist in a dataset. Association rules provide a well-situated and efficient approach to recognize and characterize certain dependencies between various attributes in a database.

An association rule is best expressed by means of the expression $X \rightarrow Y$ such that $X \cup Y \subseteq I and X \cap Y = \emptyset$. Where I is the set of items. It means that for any occurrence of item X present in the database there is relatively high probability of occurring the item Y. here X is called as antecedent and Y is called as the consequent. The strength of such rule can only be calculated by means of its support and confidence [2].

1.1Support

The support of an item set XY is defined as the proportion of transactions in the data set which contain the item set. Support is percentage or fraction of records that contain X Y to the total number of transactions in the database [23]. The formula for calculating Support(s) is:

$$\text{Support (XY)} = \frac{\text{Support count of XY}}{\text{Total number of transaction in D}}$$

Support is used to find the strongest association rules in the item sets.

1.2Confidence

Confidence is another approach for finding the association rules. Confidence of an association rule is defined as the percentage/fraction of the number of transactions that contain X Y to the total number of records that contain X, where if the percentage exceeds the threshold of confidence an interesting association rule X Y can be generated [23].

$$\text{Confidence (X|Y)} = \frac{\text{Support (XY)}}{\text{Support(X)}}$$

Using this support and confidence, the set of association rule can be extracted from database. Both frequent and rare association rules present different information about the database. Frequent rules focus on patterns that occur frequently, while rare rules focus on patterns that occur infrequently. In many applications frequently occur event is less interesting than rarely occur event. However, frequent patterns signify the known and expected while rare patterns may signify unexpected or previously unknown associations, which is valuable to domain experts.

A rare itemset is one consisting of rare items. It may be found by setting a low support threshold but leads to combinatorial explosion problem. But it is difficult to mine rare association rules using single support threshold based approaches like Apriori and Frequent Pattern-Growth (FP-Growth). The problem of specifying an appropriate support threshold causes rare item problem. Using single minsup constraint to mine frequent patterns consisting of both frequent and rare items raises the dilemma, called rare item problem. This problem is as follows [21].

1. If minsup is set very high, we fail to spot the frequent patterns consisting of rare items because rare items do not satisfy high minsup.

2. In order to discover frequent patterns consisting of both frequent and rare items, minsup should be set very low. However, it may generate too many frequent patterns. That frequent items will be related in multiple ways but most of them are not useful to the user.

1.3Rare Itemsets

An itemsets is said to be rare itemset if it satisfy less than the minimum frequent support threshold (minFreqSup) but above or equal to the minimum rare support threshold (minRareSup). There are three possible types of rare itemsets: first, itemsets which composed of rare items only; second, itemsets which composed of both rare and frequent items; and third, itemsets which composed of only frequent items which fall below the minimum support threshold. The first and second types of itemsets are referred as rare-item itemsets. Rare-item itemsets are generally more interesting than itemsets of the third type, which we call non-rare-item itemsets. This is because frequent items occur commonly in the database, and there may be many non-rare-item itemsets that do not represent any interesting Relations between items since the items only generated together by chance. Experimental results are also available for the claim that rare-item itemsets are more useful.

Formally, an itemset X is a *rare itemset* iff

$\text{support}(X) < \text{minFreqSup}$, $\text{support}(X) \geq \text{minRareSup}$

An itemset X is a *non-rare-item itemset* iff

$\forall x \in X, \text{support}(x) \geq \text{minFreqSup}$, $\text{support}(X) < \text{minFreqSup}$

An itemset X is a *rare-item itemset* iff

$\exists x \in X$, $\text{support}(x) < \text{minFreqSup}$, $\text{support}(X) < \text{minFreqSup}$

II. LITERATURE REVIEW

Most of the algorithm focuses on finding the frequent itemset, but several algorithms are available for finding rare items efficiently. Working of that paper, motivation for proposal of that algorithm, advantage and their limitations are briefly describe here. They are as follows:

2.1 Apriori Algorithm

Apriori algorithm has been proposed by Agrawal et al., 1993; Agrawal and Srikanth, 1994 for finding frequent itemsets. This algorithm is employs iterative level-wise search for frequent itemset generation it uses a single minsup value at all levels to finding frequent itemsets. Before generating frequent itemsets, algorithm generates all candidate k- itemsets having „k" number of items from that level. A candidate k- itemset is said to be frequent if the support of the subset of candidate k-itemsets is greater than or equal to the user-specified minsup threshold. This algorithm is more useful for finding the frequent itemsets and not the rare itemsets except the value of minsup is set at a low value. This algorithm inherits the drawback of explosion of frequent itemset generation and also takes too many time, space and memory for candidate generation process. It is bottom-up approach [3].

2.2 MS-Apriori Algorithm

An extension of Apriori algorithm called MS-Apriori algorithm has been proposed by Liu et al. is try to mine frequent itemsets involving rare items. It assigns a minsup (MIS) value to each item and the items having MIS value higher than lowest MIS value are used for generating frequent itemset. Based on item support percentage MIS value is derived. Frequent items having higher MIS value whereas rare items having a lower MIS value. In that way this algorithm tries to overcome the rare itemset problem and more efficient than single minsup based algorithm. Rule which having Low support and high confidence are not identified by this algorithm. The rules which have higher MIS value is removed, is the reason for inefficiency of this algorithm [4].

2.3 Relative Support Apriori Algorithm

Yun et al. proposed Relative Support Apriori Algorithm (RSAA) for generating rare association rules without specifying the user defined support threshold value. It gives higher support threshold value for items having low frequency and lower support threshold value for items having higher frequency. Thus it includes rules having less confidence [5].

2.4 Apriori Inverse Algorithm

Apriori-Inverse proposed by Kiran et al. is to mine perfectly rare itemsets. Except that at initialization, this algorithm is similar to the Apriori. Only 1-itemsets that fall below minsup are used to generating two itemsets. Apriori-Inverse inverts the downward closure property of Apriori and itemsets must also meet an absolute minimum support [6].

2.4 Rarity Algorithm

Troiano et al. analyze the problem of bottom up approach algorithms that is it searches through many levels. For reducing the number of searches Troiano et al. proposed the Rarity algorithm that starts with identification of longest transaction from database and search rare itemsets in top-down approach from that. It avoids lower layers which contains frequent itemsets. Candidates (Rare itemsets) are pruned in two different ways. One is all k-itemset candidates that are the subset of any of the frequent k + 1-itemsets are eliminated as a candidate, because they must be frequent according to the downward closure property. Another is the residual candidates have their calculated supports, and for generating the k − 1- candidates, we are used only those that have a support below the threshold. The candidates with supports above the threshold are used to prune k − 1- candidates in the next level [7].

2.5 AfRIM Algorithm

AfRIM Algorithms is proposed by Adda et al. It also uses top-down approach similar to the Rarity Algorithm. Searches for rare items starts with the itemset having all items found in database. Candidate generation is done by finding

common k-itemset subsets among all combinations of rare k+1-itemset pairs in the previous level. Pruning of candidates are same as Rarity Algorithm. It examines itemsets that have zero support, which is major drawback of this algorithm [8].

2.6 MRG-Exp Algorithm

MRG-Exp Algorithm is proposed by Szathmary et al. He Defines three types of itemsets: minimal generators (MG), which are itemsets with a lower support than its subsets; minimal rare generators (MRG), which are itemsets with non-zero support and whose subsets are all frequent; and minimal zero generators (MZG), which are itemsets with zero support and whose subsets all have non-zero support. This algorithms uses MRG for generates candidates in bottom-up fashion with use of all MGs. The MRGs represent a boundary that separates the frequent and rare itemsets. Above this boundary must be rare as per the antimonotonic property [9].

2.7 ARIMA Algorithm

ARIMA algorithm is also proposed by Szathmary et al. It uses these MRGs to generate the complete set of rare itemsets which is generated in MRG-Exp Algorithm. This is done by combing two k-itemsets with k − 1 items in common into a k + 1-itemset. When MZG reaches to border, this algorithm stops the search for non-zero rare itemsets. Because above that there are only zero rare itemsets [9].

2.8 Apriori-Rare Algorithm

Another algorithm Apriori-Rare has been proposed by Szathmary et al. to find all minimal rare itemsets. It identifies two set of items: Maximal Frequent Itemset (MFI) and minimal Rare Itemset (mRI). An itemset is a MFI if it is frequent but not all its supersets. An itemset is an mRI if it is rare but all its proper subsets are not. Generator of frequent itemsets (FIs) is also identifies by this. A Frequent Generator (FG) is an itemset which does not having proper subset with the same support. Specifying suitable threshold is the most important factor in this algorithm [10].

2.9 FP-Growth Algorithm

FP-Growth Algorithm is proposed by Han et al. which uses frequent-pattern tree (FP-tree) for storing a transactions of database and reduce database scanning. One scan is for finding the items which satisfy minimum frequency support threshold; another scan is for initial FP-tree construction. This algorithm also supports multiple minsup framework. In this, different models can be used as per user and application requirement. Broadly, they are: minimum constrain model, maximum constrain model and other models [11].

2.10 Maximum Constraint Based Conditional Frequent-Growth Algorithm

Maximum Constraint Based Conditional Frequent-Growth (MCCFP-Growth) Algorithm is extension of FP-Growth algorithm. It accepts input parameter as transactional dataset and items MIS value. Using MIS value this algorithm finds frequent patterns with a single scan on input dataset. MCCFP-growth algorithm involves three steps: one is tree construction, second is compact MIS-tree derivation, and third is mining frequent patterns. This algorithm takes more time for database scan because of pruning items. It also occupies more memory space [12].

2.11 Conditional Frequent Pattern Growth Algorithm

Conditional Frequent Pattern Growth (CFP-Growth) Algorithm is an extension of Frequent Pattern Growth (FP-Growth) Algorithm. To decrease the search space it uses heuristic that is “The items having support above lowest MIS value can use for generating frequent pattern. “ This approach is not efficient because still considers some of the items which cannot generate any frequent pattern [13].

2.12 Improved Conditional Frequent Pattern Growth Algorithm

Improved Conditional Frequent Pattern Growth (ICFP-Growth) Algorithm is an extension of Conditional Frequent Pattern Growth (CFP-Growth) Algorithm. To overcome the limitations of that algorithm this algorithms uses three heuristic. They are, The items having support above lowest MIS value can use for generating frequent pattern, for

MIS-tree, items are arranged in sorted descending order of their MIS values, all leaf nodes of the infrequent items can be pruned in MIS-tree [14].

2.13 RP-Tree Algorithm

The RP-Tree algorithm is a modification of the FP-Growth algorithm. Similar to FP-Growth algorithm, this algorithm performs database scan for counting support. In the second scan for building initial tree, RP-Tree uses the transactions having at least one rare item. In this way, the transactions having non-rare items are not included in RP-Tree construction. This algorithm provides complete set of rare-item itemset because Rare-items will never be the predecessor of a non-rare item. RP-Tree is the first algorithm that uses the tree data structure and identifies most off all rare association rules [15].

2.14 RP-Tree-IG Algorithm

Frequent items are more interesting than the rare items. For identify that rules information gain component is introduce in RP-tree algorithm for removing frequent items which is not important. It is done by treating rare items as classifications and each frequent item as a separate attribute. During the information gain calculation, if the transaction contains more than one rare items, it splits into multiple transactions [16]. Information gain is calculated as:

$$IG(X) = Entropy(Y) - Entropy(\frac{Y}{X})$$

Where Y is the set of rare items, and X is a frequent item. The items which have higher information gain than pre-defined threshold is used of item set generation.

CONCLUSION

Data mining is one of the largest and challenging areas of research with the major topic "Association Rule Mining". Most association rule mining techniques concentrate on finding frequent rules. But rare association rule is more useful and interesting than frequent association rules. This paper provides brief introduction about the algorithms which is used in the area of rare association rule mining. Several algorithms can be applied for discover rare item sets. The main purpose of this Survey is to help the researchers to select the one according to their need.

REFERENCES

[1] R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, vol.22, pp.207-216, 1993.
[2] K. Sotiris, K. Dimitris, "Association rules mining-A recent overview", Proceedings of International Transactions on Computer Science and Engineering-GESTS, vol.32, pp.71-82, 2006.
[3] R. Agrawal, R. Srikant, "Fast algorithms for mining association rules in large databases", "Proceedings of the 20th International Conference on Very Large Data Bases", pp.487-499, 1994.
[4] B. Liu, W. Hsu, Y. Ma, "Mining association rules with multiple minimum supports", Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.337-341, 1999
[5] K. Yun, D. Ha, B. Hwang, K. H. Ryu, "Mining association rules on significant rare data using relative support", Journal of Systems and Software-Elsevier, vol.67, pp.181-191, 2003.
[6] Y. S. Koh, N. Rountree, "Finding Sporadic Rules Using Apriori-Inverse", Advances in Knowledge Discovery and Data Mining-Springer, vol.3518, pp. 97-106, 2005.
[7] L. Troiano, G. Scibelli, C. Birtolo, "A Fast Algorithm for Mining Rare Itemsets", Proceedings of the Ninth International Conference on Intelligent Systems Design and Applications-IEEE, pp.1149-1155, 2009.
[8] M. Adda, L. Wu, Y. Feng, "Rare itemset mining", Proceedings of the Sixth International Conference on Machine Learning and Applications, pp.73-80, 2007.