ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

Association–Rule Mining Techniques: A general survey and empirical comparative evaluation

Mahendra Tiwari[1], Randhir Singh[2], Shivendra Kumar Singh[3]

Research Scholar, UPRTOU, Allahabad, India[1]

Asst.Professor, UIM, Allahabad, India[2]

Asst. Professor, UCER, Allahabad, India[3]

ABSTRACT - In this paper Association rule mining algorithms are discussed and compared on certain criteria. This paper also considers the use of Association rule mining in Use of decide of which products to recommend to customers. For demonstration a comprehensive experimental study against 2 Data sets of UCI are taken to evaluate the accuracy of ARM algorithms.

Keywords: UCI, ARM, DM, experiment

I. INTRODUCTION

Association mining is the process that enables us to discover which combinations of products that customers purchase and the relationships that exist at all levels in product hierarchy. The relationships discovered by the data mining are expressed as association rules.

Traditionally, the technique has been used to perform market basket analysis. In addition to the rule, the associations mining also calculate some statistics about the rule. Four statistical measures are usually used to define the rule and these are the Confidence in the association, the Support for the association, the Lift value for the association and the Type of the association.

II. ASSOCIATION RULE MINING

Frequent Item sets, Closed Item sets, and Association Rules:

Let $\{i_1, i_2, \ldots, i_m\}$ be a set of items[1, 28]. Let D, the task- relevant data, be a set of database transactions where each transaction T is a set of items such that $T \subseteq I$, each transaction is associated with an identifier, called TID. Let A be a set of items. A transaction T is said to contain A if and only if $A \subseteq T$. An association rule is an implication of the form $A \Rightarrow B$, where $A \subset I, B \subset I, and\ A \cap B = \Phi$. The rule $A \Rightarrow B$ holds in the transaction said D with Support s, Where s is the percentage of transactions in D that contain $A \cup B$(i.e., the union of sets A and B, or say, both A and B). This is taken to be the probability, $P(A \cup B)$. The rule $A \Rightarrow B$ has confidence c in the transaction said D, where c is the percentage of transactions in D containing A that also contain B. This is taken to be the conditional probability, $P(B/A)$. That is,

$support(A \Rightarrow B) = P(A \cup B)$ $confidence(A \Rightarrow B) = P(B/A)$.

Rules that satisfy both minimum support threshold $(min.sup)$ and a minimum confidence threshold $(min.conf)$ are called strong. By convention, I write support and confidence values as soon as to occur between 0% and 100% rather than $0\ to\ 1.0$.

A set of items is referred to as an item set. An item set that contains k items is a k-item set. The set $\{computer, antivirus\_software\}$ is a 2-inemset. The occurrence frequency of an item set is the number of transactions that contain the item set. This is also known, simply, as the frequency, support count, or count of the item set. Whereas the occurrence frequency is called the absolute support, If the relative support of an item set I satisfies a prespecified minimum support threshold (i.e., the absolute support of I satisfies the corresponding minimum support count threshold), then I is a frequent item set. The of frequent k −item sets is commonly denoted by L_k.

$$confidence(A \Rightarrow B) = P\left(\frac{B}{A}\right) = \frac{support(A \cup B)}{support(A)} = \frac{Support.count(A \cup B)}{supprt.count(A)}$$

In general, association rule mining can be viewed as two-step process.

Find all frequent items: By definition, each of these item sets will occur at least as frequently as predetermined minimum support count, min, and sup.

Generate strong association rules from the frequent item sets: By definition. These rule must satisfy minimum support and minimum confidence.

2.1 The Apriori Algorithm-Finding Frequent lettersets Using Candidate Generation:

Apriori is a seminal algorithm proposed by R. Agrawal and R.Srikant in 1994 for mining frequent item sets for Boolean association rules. The name of the algorithm is based on the fact that the algorithm uses *prior knowledge* of frequent item set properties. Apriori employs an iterative approach known as a *level-wise* search, where *k* -item sets are used to explore *(k+* 1) -itemsets. First, the set of frequent 1 -itemsets is found by scanning the database to accumulate the count for each item, and collecting those items that satisfy minimum support. The resulting set is denoted *L1*. Next, *L1* is used to find *L2*, the set of frequent 2-itemsets, which is used to find *L3*, and so on, until no more frequent k-item sets can be found. The finding of each *Lt* requires one full scan of the database.

To improve the efficiency of the level-wise generation of frequent item sets, an important property called the Apriori property, presented below, is used to reduce the search space. I will first describe this property, and then show an example illustrating its use. Apriori property: *All nonempty subsets of a frequent itemset must also be frequent.* The Apriori property is based on the following observation. By definition, if an itemset *I* does not satisfy the minimum support threshold, *minimum support*, then *I* is not frequent; that is, *P(i) < minimum support.* If an item *A* is added to the itemset *I.* Then the resulting itemset (i.e., *I* U *A)* cannot occur more frequently than *I.* Therefore, *I* U *A* is not frequent either; that is, *P(JUA) < minimum support.*

This property belongs to a special category of properties called ant monotone in the sense that *if a set cannot pass a test, all of its supersets will fail the same test as well,* it is called *antimonotonc* because the property is monotonic in the context of failing a test.7 *"How is the Apriori property used in the algorithm?"* To understand this, let us look at how L_k - 1 is used to find *L* for

2.2 Frequent Pattern Growth (FP) -Finding Frequent Item sets without Candidate Generation:

In many cases the Apriori candidate generate-and-test method significantly reduces the size of candidate sets, leading to good performance gain.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

However, it can suffer from two nontrivial costs: It may need to generate a huge number of candidate sets. For example, if there are 10^4 frequent 1-itemsets, the Apriori algorithm will need to generate more than 10^7 candidate 2-itemsets. moreover, to discover a frequent pattern of size 100, such as it has to generate at least 2^{100} -1 10^{30} candidates in total. It may need to repeatedly scan the database and check a large set of candidates by pattern matching. It is costly to go over each transaction in the database to determine the support of the candidate itemsets. FP-growth, adopts a divide-and-conquer strategy as follows First, it compresses the database representing frequent items into a frequent-pattern tree, or FP-tree, which retains the itemset association information. It then divides the compressed database into a set of conditional databases (a special kind of projected database), each associated with one frequent item or "pattern fragment,"

The mining of transaction database, D, of using the frequent- pattern growth approach.

The first scan of the database is the same as Apriori, which derives the set of frequent items (1 -itemsets) and their support counts (frequencies). Let the minimum support count be 2. The set of frequent items is sorted in the order of descending support count. This resulting set or list is denoted L. Thus, we have L ={{I2: 7}, [I1: 6), {I3: 6}, {I4: 2}, {I5: 2}}.

An FP-tree is then constructed as follows. First, create the root of the tree, labeled with "null." Scan database D a second time. The items in each transaction are processed in L order (i.e., sorted according to descending support count), and a branch is created for each transaction. For example, the scan of the first transaction, "T100: I1, I2, I5," which contains three items (I2, I1, I5 in order), leads to the construction of the first branch of the tree with three nodes, (I2: 1), (I1:1), and (I5: 1), where 12 is linked as a child of the root, I1 is linked to I2, and I5 is linked to Il.

The second transaction, T200, contains the items I2 and I4 in *L* order, which would result in a branch where I2 is linked to the root and I4 is linked to I2. However, this branch would share a common prefix, I2, with the existing path for T100. Therefore, I instead increment the count of the I2 node by 1, and create a new node, <I4: I>, which is linked as a child of <I2: 2>. In general, when considering the branch to be added for a transaction, the count of each node along a common prefix is incremented by 1, and nodes for the items following the prefix are created and linked accordingly.

a. **Mining Frequent Itemsets using vertical data format (ECLAT):**

ECLAT is a method that transforms a given data set of transactions in the horizontal data format of TID-itemset into the vertical data format of item-TID_set. It mines the transformed data set by TID_set intersections based on the Apriori property and additional optimization techniques, such as diffset.

III. Empirical comparison of traditional ARM techniques

Table 1: Comparison of DM techinques

Data Mining Measure	Association Rule Mining: (Apriori, FP, ECLAT)
Accuracy	Association rules should not be used directly for prediction without further analysis or domain knowledge. They do not necessarily indicate causality. They are however , helpful starting point for further exploration, making then a popular tool for understanding data.
Clarity	Frequent itemset mining leads to the discovery of association and correlations arrange items in large transactional or relational data set, with massive appoints of data continuously being collected and stored the discovery of interesting correlation relationship among huge amounts of business transaction records can help in may business decision-making processes suction customer shopping behavior analysis.
Raw data	The name of the algorithm is based on the fact the algorithm uses prior knowledge of frequent item set. All nonempty subsets of a frequent itemset must also be frequent.
Pattern set	Apriori algorithm is a seminal algorithm for mining frequent itemsets for Boolean association rules. It explores the level-wise mining Apriori property that all nonempty subsets of a frequent itemset must also be frequent FP growth is a method of mining frequent item sets without candidate generation. It constructs a highly compact data structures to express the original transition database. Rather than employing the generate and test strategy of Apriori like methods it focuses on frequent Pattern growth, which avoids costly candidate generation.
Data Format	Both the Apriori and FP growth methods mine frequent patterns from a set of transactions in TID itemset format, where TID is a transaction id and itemset is the set of items bought in transaction TID. This data format is known as horizontal data format. Data can also be presented in item – TID set format where item is an item name and TID set. Set is the set of transaction identifiers containing the items. This format
Scalability	A major challenge in mining frequent itemsets from a large data set is the fact that such mining often generates a huge numbers of itemsets satisfying the minimum support (min_sup) threshold, especially when min_sup is set low. The basic idea is to pick random samples of the given Data D, and then search for frequent itemsets in s instead of D. in this way; we trade off some degree of accuracy against efficiency.

Use of Association Mining to decide which products to recommend to customers:

A major challenge for any retailer is to decide how they select the most appropriate products to recommend to customers?

How data mining can be used to identify *cross-sell* and *up-sell* opportunities within retail business. This is achieved by matching the expected appeal of a product, not previously purchased by a customer, to the spending by the customer on related products. An association mining is used to determine product appeal and this is combined with clustering analysis to identify specific product items to recommend. The technique can be used to automate personalized product recommendation systems, or to determine product placement within retail outlets. Associations mining is the process that enables us to discover which combinations of products the customers purchase and the relationships that exist at all levels in product hierarchy.

The relationships discovered by the data mining are expressed as association rules. Association rules take the form:

Left-hand side implies right-hand side.

Traditionally, the technique has been used to perform market basket analysis. In the context of market basket analysis an example association rule may have the following form:

If product A is purchased, then this implies product B will also be purchased at the same time.

In the language of association rules, the left-hand side is called the rule "Body" and the right-hand side, the rule "Head". In general, the rule Body can contain multiple items, but the rule Head only has one item, for example:

If product A and B and C are purchased, then this implies that product D will also be purchased.

In addition to the rule, the associations mining also calculates some statistics about the rule. Four statistical measures are usually used to define the rule and these are the *Confidence* in the association, the *Support* for the association, the *Lift* value for the association and the *Type* of the association.

IV. EXPERIMENTAL SET UP

We used 2 data set for experiment with WEKA ,One of them from UCI Data repository that is German credit data set, and other is Supermarket data set inbuilt inWeka 3-6-6. Credit data set is in csv file format ,and supermarket data set is in arff file format. German credit data set contains 20 attributes while Supermarket data set contains 217 attributes.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

Input Data Set:Description of the German credit dataset.
Title: German Credit data
Number of Instances: 1000
Number of Attributes german: 20 (7 numerical, 13 categorical)
Attribute description for german

1. Status of existing checking account
2. Duration in month
3. Credit history
1. Purpose
2. Credit amount
3. Savings account/bonds
4. Present employment since
5. Installment rate in percentage of disposable income
6. Personal status and sex
7. Other debtors / guarantors
8. Present residence since
9. Property
10. Age in years
11. Other installment plans
12. Housing
13. Number of existing credits
14. Job
15. Number of people being liable to provide maintenance for
16. Telephone
17. foreign worker

Description of the Supermarket dataset.

There are 15 selected attribute of 217 attributes of supermarket.arff are taken. The attributes are bread and cake, tea, biscuits, canned fruit, frozen foods, pet foods, soft drinks, medicines, cheese, chickens, milk-cream, small goods, dairy foods**.**

Result:

Apriori Algorithm

=== Run information ===
Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation: supermarket-weka.filters.unsupervised.attribute.Remove-R1-13,15-18,20-28,30-42,44-48,50-51,53-61,63,65-73,75,77-83,85,87-94,96-102,104,106-217
=== Associator model (full training set) ===
Apriori
Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18
Generated sets of large itemsets:
Size of set of large itemsets L(1): 11
Size of set of large itemsets L(2): 20
Size of set of large itemsets L(3): 8

FPGrowth Algorithm

=== Run information ===
Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation: supermarket-weka.filters.unsupervised.attribute.Remove-R1-13,15-18,20-28,30-42,44-48,50-51,53-61,63,65-73,75,77-83,85,87-94,96-102,104,106-217
Instances: 4627
Attributes: 15
=== Associator model (full training set) ===
Apriori
=======
Minimum support: 0.1 (463 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18
Generated sets of large itemsets:
Size of set of large itemsets L(1): 11
Size of set of large itemsets L(2): 20
Size of set of large itemsets L(3): 8

V. CONCLUSION

In this paper we compared and investigated the association rule mining algorithms, the main contributions are: Descriptive survey on association rule mining,includes different strategies used to produce frequent itmsets. Comparison of traditional association rule mining techniques such as Apriori, F-P growth, & Eclat includes accuracy, raw data, pattern set etc. An experimental comparison studies using two data sets between association rule mining algorithms in term of which product recommend to customers.

REFERENCES

1. Agrawal,R. Amielinski, T., and Swami, A., (1993), Mining association rule between sets of items in large database, ACM SIGMOD int. conf. on mgmt of Washington, DC, May 26-28.
2. Antonie, M., Zaiane, O, R., Coman, A. (2003). Associative Classifiers for Medical Images. Lecture notes in AI 2797, Mining multimedia and complex datapp 68-83. Springer-verlag
3. Cendrowska,J. (1987). PRISM: An algorithm for inducing modular rules. Int. Journal of Man-Machine studies. Vol. 27, No. 4, pp-349-370.
Dong, G., Li,J. (1999), Efficient mining of frequent patterns: Discovering trends and differences. In proceeding of SIGKDD 1999, San Diego, California.
Kusiak, (2002) Data Mining and Decision making, in B.V. Dasarathy (Ed.). Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools and Technology TV, ol. 4730, SPIE, Orlando, FL, pp. 155-165.
4. Rygielski. D.,(2002) , data mining techniques for customer relationship management, Technology in society 24.
5. Chaoji V.(2008), An integrated generic approach to pattern mining: Data mining template library, Springer.
6. Hen L., S. Lee(2008), performance analysis of data mining tools cumulating with a proposed data mining middleware, Journal of Computer Science.
7. Bitterer, A., (2009), open –source business intelligence tool production deployment will grow five fold through2010, Gartner RAS research note G00171189.
8. Phyu T.(2009), Survey of classification techniques in data mining, Proceedings of the International Multiconference of Engineering and Computer Scientist(IMECS), vol 1.
9. Pramod S., O. Vyas(2010), Performance evaluation of some online association rule mining algorithms for sorted & unsorted datasets, International Journal of Computer Applications, vol 2,no. 6
10. Prasad P, Latesh Generating customer profiles for retail stores using clustering techniques, International Journal on computer science & Engineering (IJCSE).
11. Rabinovitch, L. (1999),America's first department store mines customer data. Direct marketing (62).
12. Grossman, R., S. Kasif(1999), Data mining research: opportunities and challenges. A report of three NSF workshops on mining large, massive and distributed data, pp 1-11.
13. Brijs T. et all(2000), a data mining framework for optimal product selection in a retail supermarket: The generalized PROFSET model. Data Mining & Knowledge Discovery, 300
14. Dhond A. et all (2000), data mining techniques for optimizing inventories for electronic commerce. Data Mining & Knowledge Discovery 480-486
15. Jain AK, Duin RPW(2000), statistical pattern recognition: a review, IEEE trans pattern anal mach Intell 22:4-36
16. Zhang, G.(2000), Neural network for classification: a survey, IEEE Transaction on system, man & cybernetics, part c 30(4).