International Journal of Engineering & Technology, 7 (2.21) (2018) 414-416

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Modified classic a priori algorithm for association rule mining

G. Anitha[1*], R.A. Karthika[2], G. Bindu[3], G.V. Sriramakrishnan[4]

[1]*Department of CSE, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, India.*
[2]*Department of CSE, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, India.*
[3]*Department of CSE, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, India.*
[4]*Department of IT, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, India*
**Corresponding author E-mail: agunasekaran@gmail.com*

Abstract

In today's real world environment, information is the most critical element in all aspects of the life. It can be used to perform analysis and it helps to make decision making. But due to large collection of information the analysis and extraction of such useful information is tedious process which will create a major problem. In data mining, Association rules states about associations among the entities of known and unknown group and extracting hidden patterns in the data. Apriori algorithm is used for association rule mining. In this paper, due to limitations in rule condition, the algorithm was extended as new modified classic apriori algorithm which fulfills user stated minimum support and confidence constraints.

Keywords: *Apriori, frequent itemset, support, confidence, candidate itemset.*

1. Introduction

In today's real world environment peoples are inundated with data such as business, healthcare, world wide web, governmental, scientific and financial data etc,. All such data cannot be processed by humans in short period to make analyze and prediction since the computerization of all fields. As the need arises to people to generate and collect data from various sources, some efficient mechanism is needed to analyze, classify and summarize the data to uncover and characterize the trends in the data. This is the challenging task in the world of computing environment. The biggest challenge is to analyze this enormous volume of existing and newly appearing data that require processing to solve a problem. Also there is an emergency need for new techniques and tools to mine the vast amount of data to transform into useful knowledge. This has led to the generation of a new concept called data mining and its various applications which investigates anonymous reliable patterns that are important for success in all fields such as business. Data mining is the process in which all the data and information can be extracted from large dataset for the purpose of future benefit. Commonly, association rule state a statistical correlation between certain data items. The basic form of association rule is so-called first generation association algorithm which has significant limitations. They assume that the data comes from a single repository and may use a selective data mining algorithm. Furthermore, these algorithms tend to be automation process and therefore fail to allow guidance from knowledgeable users at important stages in search for data regularities. This paper focus on a new kind of association rule mining, known as NMCA apriori mining algorithm.

2. Background and related work

Association rule is a fundamental data mining activity which uncovers the relationship between co-occurrence of items in a large set of items [1]. Using this concept all types of business transactions is analyzed. Market basket transaction is one of the best examples for an association rule and characterizes customer buying pattern. For instance, the association rule like, ***remote → battery [support = 15%, confidence = 85%]***. This means that 15% of peoples get remote and battery at the same time, and 85% of customers buy remote and also battery.

An order of getting a product is not important in association rule mining concept [2].

The issue of rule can be put as: Assume T = (T1, T2, …,Tn) be a transaction set, I = {A1, A2, …, An} be an item set, where T$\subseteq I$ *[3]*. Then the association rule is: $X \rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$. A transaction T$i \in T$ is said to contain an itemset Y if Y $\subset$ T$_i$ [3]. Support and confidence are two interestingness measures for a rule [1].

Support: Support of a rule, $X \rightarrow Y$, is a measure of how frequently the items involved in a transaction that occur together. The probability notation for support is denoted as: support $(X \rightarrow Y)$ = P($X \cup Y$).

Let n be the number of transactions in T, the support of the rule $X \rightarrow Y$ is computed as follows:

$$Support = (X \cup Y).count / n$$

Confidence: Confidence of a rule, $X \rightarrow Y$, is the conditional probability of B given A which defines certainty of the rule. Using probability notation: confidence $(X \rightarrow Y)$ = P (B/A). The confidence of the rule $X \rightarrow Y$ is computed as follows:

$$confidence = (X \cup Y).count / X.count$$

In this paper, for a given set of transaction T, the task of association rule mining is to uncover association rules in T *which* have support and confidence greater than or equal to the user-specified minimum support (min_sup) and minimum confidence (min_conf) [4][6].

Previous traditional association rule mining produce only a subset of rules with the help of various heuristics. Table 1 shows a transaction data set.

Table 1: Example of Transaction Data Set

TID	ITEMS
T1	Crab, Pie, Cake
T2	Crab, Dairy_Milk
T3	Dairy_Milk, Boots
T4	Crab, Pie, Dairy_Milk
T5	Crab, Pie, Wonder_cake, Dairy_Milk, Cake
T6	Pie, Wonder_cake, Cake
T7	Pie, Cake, Wonder_cake

Let min_sup = 25% and min_conf = 75%. The association rule is presented as : Pie, Wonder_cake→ Cake [support = 3/7, confidence = 3/3]; the rule is applicable when its support is 42.86% (> 25%) and its confidence is 100% (>75%). This rule is also applicable, whose resultants have two items: Wonder_cake→ Cake, Pie [support = 3/7, confidence = 3/3]. From this point, an excessive association rules can be find out for association rule mining.

3. Problem definition

In this paper the mining method used by the target may appear as a subsequent rule. Assume *T* be a transaction sets, T={T1,T2,…,Tn}. All individual transaction is acknowledged by a class *y*. Let *I* be set of all items, I={I1,I2,…In}, Y be the set of all class labels and $I \cap Y = \emptyset$. Then the classic association rule is $X \rightarrow y$, where $X \subseteq I$, and $y \in Y$ *[7]*. Normally, a classic association rule is different from normal association rule as follows: First the result of NMCA has only one rule item, whereas normal association rule can have more than one rule item. Second the result *y* may come from the class label *Y*, i.e., $y \in Y$. In NMCA class label can't be as a rule condition, whereas it is not the case in simple association rule. The concept of this paper is to produce NMCAs to fulfill customer stated minimum support (min_sup) and confidence (min_conf) constraints. Consider a set of transaction data set in Table 2. From table each data set is a transaction and is labeled with a class.

I = {Apple, Orange, Fruits, Place, Onion, Peas, Cabbage, Beans, Carrot, Potato}

Y = {Fruit, Vegetable}

Table 2: A Transaction Data Set

TID	TRANSACTIONS	CLASS
T1	Apple, Orange, Fruits	Fruit
T2	Apple, Fruits	Fruit
T3	Apple, Orange, Fruits, Onion	Fruit
T4	Peas, Cabbage	Vegetable
T5	Cabbage, Potato, Carrot	Vegetable
T6	Peas, Carrot, Onion, Beans	Vegetable
T7	Cabbage, Beans, Place, Onion	Vegetable

Let minsup = 20% and minconf = 70%. The class association rules are:

Apple, Fruits→ Fruit [support= 3/7, confidence = 3/3];

Onion→ Vegetable [support= 2/7, confidence = 2/3];

The next two steps involved in a new modified classic association algorithm: First, mining of data using APriori algorithm. Second, post-processing operation can be carried out on the resulting rules. But it is not always possible because all the created rules can be large.

4. implementation and discussion

In this paper, NMCAs can be mined directly using single step process. The main task of this algorithm is to uncover association rules which have above minimum support (min_sup). The form of the rule is: (*cond, y*), where cond $\subseteq I$ and $y \in Y$ is a class label [10]. The cond support count (called cond_supcount) and itemstate support count (called itemstate_supCount) are the number of transactions in *T* which contain the cond. Each state presents a rule like: *cond→ y*, whose support is (item*state_supCount* / *n*) and confidence (item*state_supCount / cond_supCount [11]*.

Itemstate which satisfy the min_sup are called frequent itemstate and other itemstate are called infrequent itemstates. For example, as shown in Table 2, the itemstate in T is ({Apple, Fruits}, Fruit). The support count of the rule cond {Apple, Fruits} is 3, and the support count of the itemstate is also 3. Then the support of the itemstate is 3/7(= 42.9%), and the confidence of the itemstate is 100%. If min_sup = 20%, then the itemstate satisfies the min_sup threshold which is frequent. If min_conf = 70%, then the itemstate satisfies the min_conf threshold. So the itemstate is confident.

The class association rule is:

Apple, Fruits→ Fruit [support= 3/7, confidence = 3/3]

The new algorithm, called The New Modified Classic **Apriori Algorithm**, is shown in Fig.1, is based on normal apriori algorithm.

1) **Algorithm NMCApriori (*T*)**
2) Ci = firstpass(*T*); // initial pass
3) $Ai = \{a \mid a \in Ci, a.\text{itemstate\_}supCount / n \geq min\_sup\}$;
4) $NMCA_1 = \{a \mid a \in Ai, a.\text{itemstate\_}supCount / a.cond\_supCount \geq min\_conf\}$;
5) **for** ($k = 2$; $A_{k-1} \neq \emptyset$; k++) **do**
6) 5 C_k=candidategenaration(A_{k-1});
7) **for** any individual $t \in T$ **do**
8) **for** any individual $c \in C_k$ **do**
9) **if** *c.condset* is contained in *t* **then** // *c* is a subset of *t*
10) *c.cond_supCount*++;
11) **if** *t.class= c.class* **then**
12) *c*.itemstate_*supCount*++
13) **end**
14) **end**
15) $A_k = \{c \in C_k \mid c.\text{itemstate\_}supCount / n \geq min\_sup\}$;
16) $NMCAk = \{a \mid a \in A_k, f.\text{itemstate\_}supCount / f.\text{cond\_supCount} \geq min\_conf\}$;
17) **end**
18) return $NMCA = \cup_k NMCA_k$;

Figure. 1: The new modified classic apriori algorithm (NMCA)

In first pass, the support count of 1-itemstatecount is computed. The rule is stated as: $C_i = \{(\{x\}, y) \mid x \in I, y \in Y\}$, where every item in I is associated with class label. Line 2 defines frequency of candidate 1-itemstate. Line 3 is used to generate one condition rules. A set of (k-1)-itemstates can be found to be frequent in the $(k\text{-}1)^{th}$ pass which is called **candidate *k*-itemstates**. In line 6-13, during each pass *cond_supCount* and itemstate*_supCount* are updated for each candidate *k*-itemstate. Line-14 finds actual frequent *k*-itemstates and line 15 generates *k* condition NMCAs.

If an itemstate / rule have a confidence of 100%, then the extension of itemstate with more than one conditions will also result with 100% confidence. But these added condition rules may be redundant which do not deliver any more information. This itemstate can significantly decrease the amount of rule generation. So the candidate generation for the next successive levels cannot be extended.

1) **Function** cgen(T_{k-1})
2) C_k= *Null*; // initialize the candidate
3) **for** $t_1, t_2 \in T_{k-1}$ // traverse all frequent itemset
4) with $a_1 = \{i_1, \ldots, i_{k-2}, i_{k-1}\}$
5) and $a_2 = \{i_1, \ldots, i_{k-2}, i'_{k-1}\}$
6) and $j_{k-1} < j'_{k-1}$ **do** // lexicographic order
7) $c = \{j_1, \ldots, j_{k-1}, j'_{k-1}\}$; // joining the two itemsets *a*1 and *a*2
8) $C_k = C_k \cup \{c\}$; // addition of new itemset *c*
9) **for** all (k-1) of *c* **do**
10) **if** ($s \notin T_{k-1}$) **then**
11) delete *c* from C_k; // delete *c* from the candidate set
12) **end**
13) **end**
14) return C_k;

Figure.2: The candidate generation function

Candidate generation algorithm is shown in Fig.2, has two steps such as join and prune step [9] [1]. The join step is shown in line 2-6. Line 7 joins two frequent (*k*-1)-itemset to yield a candidate item C_k which is then added to the set. Pruning step defines all the k-1 subsets are in Tk-1. If the itemset is not frequent and not in T_{k-1}, C_k is thus deleted. The implementation of the new algorithm is as follows: Let minsup = 15%, and minconf = 70%. The condSupCount and itemstate_supCount is denoted within the brackets after each itemset.

*T*1: {({Fruits}, Fruit):(3, 3), ({Apple}, Fruit):(3, 3), ({Orange}, Fruit):(2, 2), ({Peas}, Vegetable):(2, 2), ({Cabbage}, Vegetable):(3, 3), ({Onion}, Vegetable):(2, 2), ({Beans}, Vegetable):(2, 2)

*NMCA*1: Fruits→ Fruit [support = 3/7, confidence = 3/3]
Apple → Fruit [support = 3/7, confidence = 2/2]
Orange →Fruit [support = 2/7, confidence = 2/2]
Peas → Vegetable [support = 2/7, confidence = 2/2]
Cabbage → Vegetable [support = 3/7, confidence = 3/3]
Beans →Vegetable [support = 2/7, confidence = 2/2]

*C*2: {({Fruits, Apple}, Fruit), ({Fruits, Orange}, Fruit), ({Apple, Orange}, Fruit), ({Peas, Cabbage}, Vegetable), ({Peas, Onion}, Vegetable), ({Peas, Beans}, Vegetable), ({Cabbage, Onion}, Vegetable), ({Cabbage, Beans}, Vegetable), ({Onion, Beans}, Vegetable)}

*T*2: {({Fruits, Apple}, Fruit): (3, 3), ({Fruits, Orange}, Fruit): (2, 2), ({Onion, Beans}, Vegetable): (2, 2)}

*NMCA*2: Fruits, Apple→ Fruit [support = 3/7, confidence = 3/3]
Fruits, Orange→ Fruit [support = 2/7, confidence = 2/2]
Onion, Beans→ Vegetable [support = 2/7, confidence = 2/2]

Classic association rules are used in the following ways. First, CARs are used in machine learning models. Second, they are useful in various applications [8].

5. conclusion

The field of data mining has importance in finding patterns in data, clustering and classifying customer behaviors, and discovery of knowledge etc., in a variety of domains. A number of data mining algorithms have been proposed for association rule mining, which have different mining methods and ideas. But all the results are mostly same based on the respective association rule used. For given transaction set T, the set of association rules in T is discovered using minimum support and minimum confidence [7]. In this paper, the New Modified Classic Association Apriori Algorithm is used to fulfill the user stated minimum support and minimum confidence constraints. The new candidate generation function is same as candidate generation of normal Apriori algorithm. The only difference with new candidate generation is that item states with the same class are joined by their cond. From this, Data mining has wide application domain in various organization where the data is generated. Data mining is considered one of the most important frontiers in database and information systems and one of the most promising interdisciplinary developments in Information technology with various objectives and goals.

References

[1] Han J & Kamber M, "Data Mining: Concepts and Techniques", *Morgan Kaufmann Publishers*, Book, (2000).

[2] Aggarwal CC, Procopiuc CM & Yu PS, "Finding Localized Associations in Market Basket Data", *Knowledge and Data Engineering*, Vol.14, No.1, (2002), 51–62.

[3] Borgelt C & Kruse R, "Induction of Association Rules: Apriori Implementation", *Proc. 15th Conf. on Computational Statistics (Compstat 2002, Berlin, Germany)*. Physika Verlag, Heidelberg, Germany, (2002).

[4] Goethals B & Zaki MJ, "Advances in Frequent Itemset Mining Implementations: Report on FIMI'03", *SIGKDD Explorations*, Vol.6, No.1, (2004), pp.109–117.

[5] Hipp J, G¨untzer U & Nakhaeizadeh G, "Algorithms for Association Rule Mining A General Survey and Comparison", *SIGKDD Explorations*, Vol.2, No.2, (2000), pp.1–58.

[6] Rao S & Gupta R, "Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm", *International Journal of Computer Science And Technology*, (2012), pp.489-493.

[7] Fayyad U, Piatetsky-Shapiro G & Smyth P, "From data mining to knowledge discovery in databases," *AI magazine*, Vol.17, No.3, (1996), pp.1-37.

[8] AL-Zawaida FH, Jbara YH & Marwan AL, "An improved algorithm for mining association rules in large databases", *World of Computer Science and Information Technology Journal*, Vol.1, No.7, (2011), pp.311-316.

[9] Agrawal R, Imielinski T & Swami A, "Mining association rules between sets of items in large databases", *ACM SIGMOD Record*, Vol.22, (1993), pp.207–216.

[10] Patel MR, Rana DP & Mehta RG, "FApriori: A modified Apriori algorithm based on checkpoint", *IEEE International Conference on Information Systems and Computer Networks (ISCON)*, (2013), pp.50-53.

[11] Lekha A, Srikrishna CV & Vinod V, "Utility of association rule mining: A case study using Weka tool", *IEEE International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT)*, (2013), pp.1-6.