Association Rule Mining Using Apriori Algorithm

Dr. K. V. Sobha Rani[1] and Dr. CH.V.SivaRam Prasad[2]

[1]*Assistant Professor, Department of Computer Applications, PR Govt Degree College, Kakinada, A.P., India*

[2]*Proofessor, Department of Mathematics, Aditya Engineering College, Kakinada, A.P., India*

ABSTRACT

Data mining has evolved into an important and active area of research because of theoretical challenges and practical applications associated with the problem of discovering (or extracting) interesting and previously unknown knowledge from very large real-world databases. In this paper, generation of association rules for mushroom dataset is considered. Apriori algorithm is used for the generation of association rules with various levels of minimum support and minimum confidence. Generated association rules are analyzed to identify the attributes which cause the mushroom edible or poisonous.

Keywords: *Data Mining (DM), Association Rules (AR), Frequent Itemsets (FI), Apriori (Ap).*

I. INTRODUCTION

Most organizations possess large volumes of data about their business processes and resources. While this data can provide plenty of statistical information, very little useful knowledge can be procured from it. In order to gain such useful knowledge, there is a need to discover patterns in the data, associated with the past behaviour of business processes. These patterns are used to dictate future strategy so as to maximize performance and profit. Such a knowledge discovery process is called Data Mining [4].

Data mining has been applied successfully in fraud detection, manufacturing, banking, and telecommunications, as well as in astronomy, remote sensing, and protein sequencing [2]. Westphal and Blaxton [5] categorized data mining functions as classification, segmentation, and associations. Classification involves assigning labels to new data based on the knowledge extracted from historical data. Segmentation (called also clustering) divides a population into smaller sub-populations with similar behaviour according to a predefined metric. It maximizes homogeneity within a group and maximizes heterogeneity between the groups. Link analysis, alternatively referred to as affinity analysis or association, refers to the data mining task of uncovering relationships among data. The best example of this type of application is to determine association rules. An association rule is a model that identifies specific type of data associations. These associations are often used in the retail sales community to identify items that are frequently purchased together [1].

II. IMPORTANCE OF ASSOCIATION RULES

Discovery of association rules is an important component of data mining. Association rules have been widely used by the retail industry under the name "market basket analysis". However, the concept of association rules is general and has wide applicability. Associations are used in many other applications such as predicting the failure of telecommunication switches [3]. The focus of this paper is on the application of association rules to mushroom data. The mushroom dataset consists of 23 fields and 8,124 records. The generated association rules analyzed to identify the combinations of attributes which cause the mushroom edible or poisonous. The discovery of association rules for mushroom dataset is useful for the people, who deal with mushrooms in decision making such as to know the combinations of attributes which cause the mushroom edible or poisonous. The mushroom dataset consists of fields such as cap-shape, cap-color, odor, cap-surface, bruises, gill-attachment, gill-spacing etc., The last field, state, specifies whether the mushroom is edible or poisonous.

III. BASIC CONCEPTS

Consider a small store that sells the following set of items: {Bagels, Bread, Butter, Cereal, Jelly, Juice, Milk}. List of items bought by six hypothetical customers are shown in Table 1. This table will be used

to illustrate the concepts presented in this section. Each row of the table is referred to as a transaction.

A. Definition 1

- Given a set S of items, any nonempty subset of S is called an itemset.
- Given an itemset I and a set T of transactions, the support of I with respect to T, denoted by $support_T(I)$, is the number of transactions in T that contain all the items in I.
- Given an itemset I, a set T of transactions and a positive integer α, I is a large itemset with respect to T and α if $support_T(I) \geq \alpha$. α is called as support threshold.

For simplicity, when the transaction set T is clear from the context, we use "support" instead of "support with respect to T".

Let S denote the set {Bagels, Bread, Butter, Cereal, Jelly, Juice, Milk} and let T denote the set of transactions shown in Table 1

Table 1: A small set of transactions

No.	Items Purchased
1	{Bread, Butter, Cereal, Juice, Milk}
2	{Cereal, Juice, Milk}
3	{Bagels, Butter, Cereal, Juice, Milk}
4	{Bread, Cereal, Jelly, Juice, Milk}
5	{Bagels, Jelly, Juice, Milk}
6	{Jelly, Juice, Milk}

Examples of itemsets are I_1={Cereal, Juice, Milk} and I_2={Bagels, Bread}.

The support of itemset I_1 with respect to T is 4 since I_1 appears in exactly four of the transactions shown in Table 1. The support of itemset I_2 with respect to T is zero since no transaction in T contains both Bagels and Bread.

If the support threshold is 3, then I_1 is a large itemset since the support of I_1 is 4. Another itemset for this support threshold is I_3={Jelly, Juice, Milk}, which has a support of 3. Since the support of I_2 is zero, I_2 is not a large itemset for any poisitive support threshold.

B. Definition 2

- An association rule is a pair of disjoint itemsets. If L and R denote the two disjoint itemsets, the association rule is written as $L \Rightarrow R$.
- The support of the association rule $L \Rightarrow R$ with respect to a transaction set T is the support of the itemset $L \cup R$ with respect to T.
- The confidence of the rule $L \Rightarrow R$ with respect to a transaction set T is the ratio support($L \cup R$)/support(L).

Consider the itemsets A_1={Juice, Milk} and A_2={Cereal}. Since A_1 and A_2 are disjoint, $A_1 \Rightarrow A_2$ (or equivalently, {Juice, Milk} $\Rightarrow$ {Cereal}) is an association rule. Let R_1 denote this association rule.

The support of R_1 is the support of the itemset {Juice, Milk, Cereal}. From Table 1, it can be seen that this support value is 4. Also from Table 1, the support of the itemset {Juice, Milk} is 6. Therefore, the confidence of Rule R_1 is 4/6 or 66.67%.

A given association rule $L \Rightarrow R$ is important if it has high confidence for a given support threshold. In the context of a supermarket, such a rule indicates that a customer who buys the items in set L is also likely to buy the items in the set R. For a given support threshold, rules with larger confidence values are more important than those with smaller confidence values.

IV. ALGORITHM FOR ASSOCIATION RULES

In this paper, Apriori procedure is considered for generation of association rules. It is discussed in the following section

A. Apriori Algorithm

Apriori is an algorithm for extracting association rules from data. It contains the search space for rules by discovering frequent itemsets and only examining rules that are made up of frequent itemsets. Apriori deals with items and itemsets that make up transactions. Items are flag-type conditions that indicate the presence or absence of a particular thing in a specific transaction. An itemset is a group of items which may or may not tend to co-occur within transactions. Apriori proceeds in two stages. Firstly, it identifies frequent itemsets in the data, and then it generates rules from the table of frequent pitemsets.

The first step in Apriori is to identify frequent itemsets. A frequent itemset is defined as an itemset with support greater than or equal to the user-specified minimum support threshold s_{min}. The support of an itemset is the number of records in which the itemset is found divided by the total number of records.

The algorithm begins by scanning the data and identifying the single-item itemsets (i.e. individual items, or itemsets of length 1) that satisfy this criterion. Any single item that does not satisfy the criterion is not to be considered further, because adding an infrequent

item to an itemset will always result in an infrequent itemset.

Apriori then generates itemsets recursively using the following steps:

- Generates a candidate set of itemsets of length k (containing k items) by combining existing itemsets of length (k-1). For every possible pair of frequent itemsets p and q with length (k-1), it compares the first (k-2) items (in lexicographic order); if they are the same, and the last item in q is (lexicographically) greater than the last item in p, it adds the last item in q to the end of p to create a new candidate itemset with length k.
- Prunes the candidate set by checking every (k-1) length subset of each candidate itemset; all subsets must be frequent itemsets, or the candidate itemset is infrequent and is removed from further consideration.
- Calculates the support of each itemset in the candidate set, as support = N_i/N where N_i is the number of records that match the itemset and N is the number of records in the training data.
- Itemsets with support $\geq s_{min}$ are added to the list of frequent itemsets.
- If any frequent itemset of length k is found, and k is less than the user-specified maximum rule size k_{max}, it repeats the process to find frequent itemsets of length (k+1).

When all frequent itemsets have been identified, the algorithm extracts rules from the frequent itemsets. For each frequent itemset L with length k > 1, Apriori generates rules using the following steps:

- Calculates all subsets A of length (k-1) of the itemset such that all the fields in A are input fields and all the other fields in the itemset (those that are not in A) are output fields. Call the latter subset A^I. (In the first iteration this is just one field, but in later iterations it can be multiple fields).
- For each subset A, it calculates the evaluation measure (rule confidence by default) for the rule $A \Rightarrow A^I$.
- If the evaluation measure is greater than the user-specified threshold, it adds the rule to the rule table, and, if the length k of A is greater than 1, it tests all possible subsets of A with length (k-1).

B. Association Rules using Apriori Algorithm

Case I: Generated association rules by using Apriori algorithm on the fields bruises (with value no), gill-attachment (with value free), odor (with value foul) and state (with value Poisonous) at 10% minimum support and 35% minimum confidence are as follows:

Rules for Poisonous state – contains 7 rule(s)

Rule 1 for Poisonous (1584, 100%)
If odor=foul
then Poisonous
Rule 2 for Poisonous (1532, 62.28%)
If bruises=no
then Poisonous
Rule 3 for Poisonous (2138, 38%)
If gill-attachment=free
then Poisonous
Rule 4 for Poisonous (1296, 100%)
If odor=foul
and bruises=no
then Poisonous
Rule 5 for Poisonous (1584, 100%)
If odor=foul
and gill-attachment=free
then Poisonous
Rule 6 for Poisonous (1514, 61.99%)
If bruises=no
and gill-attachment=free
then Poisonous
Rule 7 for Poisonous (1296, 100%)
If odor=foul
and bruises=no
and gill-attachment=free
then Poisonous

Rule 1 specifies that if the mushroom smells foul, then there is 100% chance of the mushroom being Poisonous and 1584 instances support this rule. Rule 2 specifies that if the mushroom has no bruises, then there is a 62.28% chance that it is Poisonous and 1532 instances support this rule. Rule 3 specifies that the mushroom has free gill-attachment, then there is 38% chance that the mushroom is Poisonous and 2138 instances support this rule. Rule 4 states that if the mushroom smells foul and has no bruises, then there is a 100% chance of the mushroom being Poisonous and 1296 instances support this rule. Rule 5 states that if the mushroom smells foul and has free gill-attachment, then there is a 100% chance of the mushroom being Poisonous and 1584 instances support this rule. Rule 6 states that if the mushroom has no bruises and has free gill-attachment, then there is a 61.99% chance that it is Poisonous and 1514 instances support this rule. Rule 7 states that if the mushroom smells foul and has free gill-attachment and no bruises, then there is a 100% chance that it is Poisonous and 1296 instances support this rule.

From above, association rules 1, 4, 5 and 7 are important. So

1. odor (with value foul)
2. odor (with value foul) and bruises (with value no)
3. odor (with value foul) and gill-attachment (with value free)
4. odor (with value foul) and bruises (with value no) and gill-attachment (with value free)

are the combinations of important attributes which cause the mushroom Poisonous.

Case II: Generated association rules by using Apriori algorithm on fields stalk-shape (with value tapering), stalk-root (with value equal), stalk-surface-above-ring (with value smooth) and state (with value Edible) at 10% minimum support and 35% minimum confidence are as follows:

Rules for Edible state – contains 5 rule(s)
Rule 1 for Edible (864, 77.14%)
If stalk-root=equal
then Edible
Rule 2 for Edible (2592, 90%)
If stalk-shape=tapering
then Edible
Rule 3 for Edible (3064, 82.01%)
If stalk-surface-above-ring = smooth
then Edible
Rule 4 for Edible (768, 100%)
If stalk-root=equal and
stalk-shape=tapering
then Edible
Rule 5 for Edible (2208, 93.87%)
If stalk-shape=tapering and
stalk-surface-above-ring=smooth
then Edible

Rule 1 specifies that if the mushroom's stalk-root is equal, then there is 77.14% chance of the mushroom being Edible and 864 instances support this rule. Rule 2 specifies that if the mushroom stalk-shape is tapering, then there is a 90% chance that it is Edible and 2592 instances support this rule. Rule 3 specifies that the surface of the stalk above the ring is smooth, then there is 82.01% chance that the mushroom is Edible and 3064 instances support this rule. Rule 4 states that if the mushroom's stalk-root is equal and stalk-shape is tapering, then there is 100% chance of the mushroom being Edible and 768 instances support this rule. Rule 5 states that if stalk-shape of the mushroom is tapering and the surface of the stalk above the ring is smooth, then there is 93.87% chance that it is Edible and 2208 instances support this rule.

From above rules, it is observed that there is a strong relation between Edible state, equal stalk-root, and tapering stalk-shape.

References

[1] Dunham M. H., and Sridhar S., "*Data Mining-Introductory and Advanced Topics"*, Pearson Education, 2003.
[2] Fayyad U., "*Taming the Giants and the Monsters: Mining Large Databases for Nuggets of Knowledge",* Database Programming and Design Magazine, March 1998.
[3] Han J., and Kamber M., "*Data Mining Concepts and Techniques"*, Elsevier 2001.
[4] Pujari A. K., "*Data Mining Techniques"*, University Press, 2001.
[5] Westphal C., and Blaxton T., "*Data Mining Solutions,"* John Wiley, NewYork, 1998.