



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

Implementation of classification using association rule mining

Kanu Patel¹ Vatsal Shah²

^{1,2} Assistant Professor, I.T Department, B.V.M. Engineering College
Gujarat Technological University, Gujarat, INDIA

Abstract: Association rule mining and classification are two major task of data mining. I propose a method for classification rules from single-label dataset using association rule analysis Finding association rule from dataset we have to apply various algorithms like Apriori, Fp-Growth, etc. I proposed Apriori algorithm for finding association rule from dataset. The number of associations present in even moderate sized databases can be, however, very large – usually too large to be applied directly for classification purposes. Therefore, any classification learner using association rules has to perform three major steps: Mining a set of potentially accurate rules, evaluating and pruning rules, and classifying future instances using the found rule set. Implementation of apriori algorithm gives accurate and classify rule. This approach is more effective, accurate and efficient than other tradition algorithms.

Keywords: Rule mining; Association rule, Mulans; Classification; Apriori ; Weka

I. Introduction

Data mining, the extraction of hidden predictive information from large databases [1], is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses. Remember that the mining of gold from rocks or sand is referred to as gold mining rather than rock or sand mining. Thus, data mining should have been more appropriately named “knowledge mining from data,” which is unfortunately somewhat long. “Knowledge mining,” a shorter term may not reflect the emphasis on mining from large amounts of data. Nevertheless, mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material. Thus, such a misnomer that carries both “data” and “mining” became a popular choice.

The classification problem is to build a model, which, based on external observations, assigns an instance to one or more labels. A set of examples is given as the training set, from which the model is built. A typical assumption in classification is that labels are mutually exclusive, so that an instance can be mapped to only one label. However, due to ambiguity or multiplicity, it is quite natural that most of the applications violate this assumption, allowing instances to be mapped to multiple labels simultaneously. For example, a movie being mapped to action or adventure, or a song being classified as rock or ballad, could all lead to violations of the single-label assumption? Multi-label classification consists in learning a model from instances that may be associated with multiple labels, that is, labels are not assumed to be mutually exclusive. Most of the proposed approaches.[1] for multi-label classification employ heuristics, such as learning independent classifiers for each label, and employing ranking and threshold schemes for classification. Although simple, these heuristics do not deal with important issues such as small disjoints and correlated labels.

II. Preliminary

In this section, we explain the concept of association rule mining and classification.

A. Association rules Mining

The original definition by Agrawal et al. In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. Association rule mining is defined as: Let $I = \{i_1, i_2, \dots, i_n\}$ be a set of n binary attributes called *items*. Let $D = \{t_1, t_2, \dots, t_n\}$ be a set of transactions called the *database*. Each transaction in D has a unique transaction ID and contains a subset of the items in I . A *rule* is defined as an implication of the form $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The sets of items (for short item sets) X and Y are called antecedent (left-hand-side or LHS) and consequent (right-hand-side or RHS) of the rule respectively. To illustrate the concepts, we use a small example from the supermarket domain. The set of items is $I = \{\text{milk, bread, butter, beer}\}$ and a small database containing the items (1 codes presence and 0 absence of an item in a transaction) is shown in the table to the right. An example rule for the

supermarket could be $\{\text{butter, bread}\} \Rightarrow \{\text{milk}\}$ meaning that if butter and bread is bought, customers also buy milk..

B. Classification.

Classification refers to an algorithmic procedure for assigning a given piece of input data into one of a given number of categories. An example would be assigning a given email into "spam" or "non-spam" classes or assigning a diagnosis to a given patient as described by observed characteristics of the patient (gender, blood pressure, presence or absence of certain symptoms, etc.). An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm, which maps input data to a category. Classification is a well-known task in data mining and is used to predict the class of an unseen instance as accurately as possible. Whilst single-label classification, which assigns each rule in the classifier the most obvious class label, has been widely studied [6] little work has been conducted on multi-label classification. Furthermore, the majority of the research carried out to date on multi-label classification relates mainly to text categorization

Data Mining provides the following techniques for classification:

1. **Decision Tree** : Decision trees automatically generate rules, which are conditional statements that reveal the logic used to build the tree.
2. **Naive Bayes** : Naive Bayes uses Bayes' Theorem, a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data.
3. **Generalized Linear Models (GLM)** : GLM is a popular statistical technique for linear modeling. Oracle Data Mining implements GLM for binary classification and for regression. GLM provides extensive coefficient statistics and model statistics, as well as row diagnostics. GLM also supports confidence bounds.
4. **Support Vector Machine**: Support Vector Machine (SVM) is a powerful, state-of-the-art algorithm based on linear and nonlinear regression. Oracle Data Mining implements SVM for binary and multiclass classification. The nature of the data determines which classification algorithm will provide the best solution to a given problem. The algorithm can differ with respect to accuracy, time to completion, and transparency. In practice, it sometimes makes sense to develop several models for each algorithm, select the best model for each algorithm, and then choose the best of those for deployment.

III. The proposed algorithm

A. Apriori

The Apriori Algorithm proposed by Agrawal et. al. in 1994, finds frequent items in a given data set using the ant monotone constraint [4,5]. Apriori is an influential algorithm in market basket analysis for mining frequent item sets for Boolean association rules. The name of Apriori is based on the fact that the algorithm uses aprior knowledge of frequent itemset properties. Apriori employs an iterative approach known as a level-wise search, where k itemsets are used to explore (k+1)-itemsets [6]. First, the set of frequent 1-itemsets is found, denoted by L1. 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-itemsets can be found. Property:

All non empty subsets of frequent item sets must be frequent. Apriori algorithm is an influential algorithm for mining frequent itemsets for Boolean association rules. This algorithm contains a number of passes over the database. During pass k, the algorithm finds the set of frequent itemsets L_k of length k that satisfy the minimum support requirement. The algorithm terminates when L_k is empty. A pruning step eliminates any candidate, which has a smaller subset. The pseudo code for Apriori Algorithm is following:

Algorithm:

C_k: candidate itemset of size k

L_k: frequent itemset of size k

L1 = {frequent items};

For (k=1; L_k != null; k++) do begin

C_{k+1} = candidates generated from L_k;

For each transaction t in database do Increment the count of all candidates in

C_{k+1} that are contained in t

Lk+1 = candidates in Ck+1 with min_support
End

In the above example, a frequent item set with size 3 is finally mined. And its support is higher than the minimum support. With the Apriori algorithm, only frequent item sets satisfy minimum support threshold can be generated.

B. WEKA implementation.

In order to compare the different approaches we use standard benchmark datasets from the UCI Machine Learning Repository and Mulan: A Java Library for Multi-Label Learning. Class association rule mining as well as association rule mining in general is only possible for nominal attributes. Therefore we need to discredit the numeric attributes in our dataset. For this purpose the WEKA's implementation of the J48 method in trees class. For classification of association rule mining, we have two different types of datasets. Single label class attributes and Multi-label class attributes. I had implemented both datasets. In single label classification we have to use WEKA tools. But some limitation of weka we can't apply for multi label datasets.

Step 1: Open Weka tools click on preprocesses and select dataset file location. I have to use "weather. Nominal" datasets. At that time tools will generate all label attribute and also class label attribute or predicated attribute. Procedure FP-growth

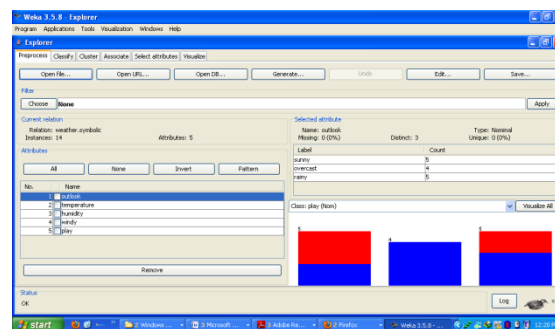


Figure 1. Weka implementation for single

Step 2: Selecting dataset after we have to generate association rules. For that click on association button and choose association algorithm (select apriori) and set some parameter in algorithms like as number of rule. It's give best of rules and display information.

=== 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: weather. Symbolic
Instances: 14
Attributes: 5 outlook
temperature
humidity
windy
play → Class label attribute.

Best rules found:

1. outlook=overcast 4 → play=yes
2. temperature=cool 4 → humidity=normal
3. humidity=normal windy=FALSE 4 → play=yes
4. outlook=sunny play=no 3 → humidity=high
5. outlook=sunny humidity=high 3 → play=no
6. outlook=rainy play=yes 3 → windy=FALSE
7. outlook=rainy windy=FALSE 3 → play=yes
8. temperature=cool play=yes 3 → humidity=normal
9. outlook=sunny temperature=hot 2 → humidity=high

10. temperature=hot play=no 2 → outlook=sunny

Step 3: In Classify, we have to select our class label attribute and also choose a classification method and give cross-validation is 10. So this classifier prunes the rules that are generated by apriori algorithm.

=== Classifier model (full training set) ===

J48 Pruned

outlook = sunny

| humidity = high : no (3.0)

| humidity = normal : yes (2.0)

outlook = overcast : yes (4.0)

outlook = rainy

| Windy = TRUE : no (2.0)

| Windy = FALSE : yes (3.0)

IV. Experiments

In this section association rule, right end side comes all attribute but when will generates classify rule then right end side only class label attribute presents and all five classify rules are already comes in association rule. In some case, we get idea about label attribute for first predicted attribute so why we check second attribute it's a drawback of association rule. For that refer rule no 2 and 8.

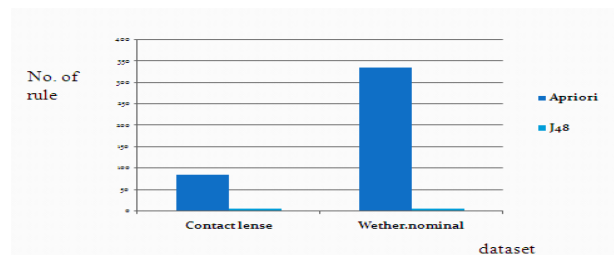


Figure 2 Comparison Apriori with J48

V. Conclusion

In conclusion, this paper analyzes a new approach for classification rules has been proposed that has many features using associative rule classification methods It produces classifiers that contain rules with single labels. It employs an efficient method for discovering rules that requires number of transaction scan over the training data so we can reduce our time using this algorithm. Which prunes redundant rules, and ensures only effective ones are used for classification. so we can optimize the memory space and generate accurate rule. Using this method we generate generalize rule and reduce number of association rule.

VI. References

- [1] M. Boutell, J. Luo, X. Shen, and C. Brown. Learning multi-label scene classification. Pattern Recognition, 37(9):1757–1771, 2004.
- [2] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules in Large Databases,” Proceeding of the 20th International Conference on VLDB, pp. 478-499, 1994.
- [3] Li, W., Han, J., Pei, J.: CMAR: accurate and efficient classification based on multiple class association rule. In: Proceedings of the ICDM’01, pp. 369–376. San Jose, CA (2001).
- [4] J. Han, “Mining knowledge at multiple concept levels”, Proceeding of In ACM International Conference on Information and Knowledge Management (CIKM’95), Baltimore, Maryland, USA, pp. 19 - 24 , November 1995.
- [5] J. Han and M. Kamber, “Data Mining: Concepts and Techniques”, Morgan Kaufmann Publishers, August 2000.
- [6] J. Han, J. Pei and Y. Yin, “Mining frequent patterns without candidate generation”, Proceeding of 2000 ACM-SIGMOD International Conference Management of Data (SIGMOD’00), pp. 1-12, 2000.
- [7] J. Han, J. Pei, Y. Yin, and R. Mao, “Mining frequent patterns without candidate generation: a frequent pattern tree approach”, Data Mining and Knowledge Discovery 8, pp. 53-87, 2004.