A Framework for Integrating Association and Classification Rules in Data Mining

Dr. P. Devaraaju

Assistant Professor
Dept. of Computer Science and Technology, S K University, Anantapuramu - AP

Abstract-- **Large amounts of data are being generated and stored every day in Organizational computer database systems. Data mining is to discover knowledge from large amounts of data and is widely used in business world. Mining association rules from transactional data is becoming a popular and important knowledge discovery technique. Association rule mining is a data mining task that discovers relationships among items in a transactional database. One of the branches of data mining is Associative Classification (AC). AC algorithms integrate association rules discovery and classification to build a classifier from a training data for predicting the class of unforeseen test data. AC algorithms typically build a classifier by discovering the full set of Class Association Rules (CARs) from the training dataset and then select a subset to form a classifier. CARs are association rules of the form A $\Longrightarrow$ c, where A is an itemset and c is a class.**

Despite achieving high accuracy compared to other classification approaches such as Decision Tree, the approach suffers from the overhead of exhaustive search through a large pool of candidate rules. Moreover, the rule discovery process in traditional AC algorithms is not well integrated with the classification process.

Keywords: ***Data Mining, Association, Classification, CAR***

I. INTRODUCTION

Data mining is to discover knowledge from large amounts of data and is widely used in business world. The previously unknown knowledge mined increases business intelligence, provides better support for decision making and consequently promotes the business competition. In order to discover rich and useful knowledge, many different types of data mining techniques are used. Mining association rules [1] from transactional data is becoming a popular and important knowledge discovery technique. Association rule mining is a data mining task that discovers relationships among items in a transactional database. An association rule is an implication of the form $A \Longrightarrow B$, where A and B are frequent itemsets in a transaction database and $A \cap B = \phi$. In practical applications, the rule $A \Longrightarrow B$ can be used to predict that 'if *A* occurs in a transaction, then *B* will likely also occur in the same transaction', and we can apply this association rule to place '*B* close to *A*' in the store layout and product placement of supermarket management. Association rules have been extensively studied in the literature for their usefulness in many application domains such as recommender systems, diagnosis decisions support, telecommunication, intrusion detection, etc. The efficient discovery of such rules has been a major focus in the data mining research community.

One of the branches of data mining is Associative Classification (AC). AC algorithms integrate association rules discovery and classification to build a classifier from a training data for predicting the class of unforeseen test data. AC algorithms typically build a classifier by discovering the full set of Class Association Rules (CARs) from the training dataset and then select a subset to form a classifier. CARs are association rules of the form A $\Longrightarrow$ c, where A is an itemset and c is a class.

Despite achieving high accuracy compared to other classification approaches such as C4.5, the approach suffers from the overhead of exhaustive search through a large pool of candidate rules. Moreover, the rule discovery process in traditional AC algorithms is not well integrated with the classification process.

II. PROBLEM DEFINITION

The associative classifier is a classifier that uses association rule mining in the training phase in order to generate classification rules. To use this classifier, datasets have to be transformed in a transactional format. Considering each attribute-value pair in a dataset as an item results in a transactional dataset in which a row of data looks like a transaction of items. Among items of each transaction, one is the class label of the related object. Using an association rule mining technique on the resulting transactional data, frequent itemsets are mined and the ones of the form $\{A, c\}$ are extracted where A is a set of features and c is a class label (A and c are disjoint subsets of items). Among these frequent itemsets, the confident ones are chosen to build classification rules of the form $A \Longrightarrow c$. Then, these rules are used to predict class labels for objects with an unknown class.
Given a training data set T, for a rule R : P→c

- The support of R, denoted as sup(R) , is the number of rows in T matching R condition and having a class label c

- The confidence of R , denoted as conf(R), is the number of rows matching R condition and having class label c over the number of objects matching R condition.
- Any Item has a support larger than the user minimum support is called frequent itemset.

III. A FRAMEWORK OF ASSOCIATIVE CLASSIFICATION

The Associative Classification algorithm generates all the frequent *ruleitems* by making multiple passes over the data. In the first pass, it counts the support of individual *ruleitem* and determines whether it is frequent. In each subsequent pass, it starts with the seed set of *ruleitems* found to be frequent Our objectives are (1) to generate the complete set of CARs that satisfy the user-specified minimum support (called minsup) and minimum confidence (called *minconf*) onstraints, and (2) to build a classifier from the CARs.

The proposed algorithm, which is given below, will use this notations and formulae to find class association rules.

The algorithm is shown below which is used to find the Class Association Rules:

```
Input  : Transaction D, min sup, min conf & class label
Output : Associative classification rules
F_1= {large 1-ruleitems};
CAR_1 = genRules(F_1);
for (k=2; F_{k-1}=Ø; k++)
    C_k= candidateGen(F_{k-1});
    for each data case d ∈ D
        Cd = ruleSubset(Ck, d);
        for each candidate c ∈ Cd
            c.condsupCount++;
            if d.class=c.class then
                c.rulesupCount++;
            end
        end
    end
    Fk = { c ∈ C_k | c.rulesupCount ≥ minsup};
    CARk = genRules (Fk);
end
CARs= ∪_k CARk;
```

Figure 1 : Proposed Algorithm

3.1 Classifier Building

After all rules (CARs) are found, a classifier is built using the rules. Clearly, there are many possible methods to build a classifier from the rules. The selection of rules is based on a total order defined on the rules.

Definition: Given two rules, *ri* and *rj*, $ri \phi rj$ (also called *ri* precedes *rj* or *ri* has a higher precedence than *rj*) if

1. the confidence of *ri* is greater than that of *rj*, or
2. their confidences are the same, but the support of *ri* is greater than that of *rj*, or
3. both the confidences and supports of *ri* and *rj* are the same, but *ri* is generated earlier than *rj*.

Let *R* be the set of CARs, and *D* the training data. The basic idea of the classifier-building algorithm in CBA is to choose a set of high precedence rules in *R* to cover *D*. This method is related to the traditional covering method.

A CBA classifier is of the form: <*r*1, *r*2, …, *rn*, default_class> where $ri \in R$, $ra \phi rb$ if $b > a$. In classifying an unseen case, the first rule that satisfies the case classifies it. If no rule applies to the case, it takes the default class (default_class).

3.2 Associative Classification Example

Consider the training data shown in Table 1, which represents three attributes A1 (a1, b1, c1), A2 (a2, b2, c2) and A3 (a3, b3, c3) and two class labels (y1, y2). Assuming minsupp = 30% and minconf = 80%, the frequent one, two and three ruleitems for Table 1 are shown in figure 2, along with the relevant supports and confidences. In cases where a ruleitem is associated with multiple classes, only the class with the largest frequency is considered by current associative classification methods. Frequent ruleitems in bold in figure 2 represent those that pass the confidence and support thresholds, which are converted into rules. Finally the classifier is constructed using an ordered subset of these rules.

Table 1: Training data set

Trans.ID	**X1**	**X2**	**X3**	**Class**
1	a1	a2	b3	y1
2	a1	a2	c3	y2
3	a1	b2	b3	y1
4	a1	b2	b3	y2
5	b1	b2	a3	y2
6	b1	a2	b3	y1
7	a1	b2	b3	y1
8	a1	a2	b3	y1
9	c1	c2	c3	y2
10	a1	a2	b3	y1

<a2 b3>	**→**	**y1**	**40% (Supp)**	**100% (conf)**
<a1, a2, b3>	**→**	**y1**	**30%**	**100%**
<b3>	**→**	**y1**	**60%**	**85%**
<a1,b3>	**→**	**y1**	**50%**	**83%**
<a2>	**→**	**y1**	**40%**	**80%**
<a1>	**→**	**y1**	**50%**	**71%**
<a1,a2>	**→**	**y1**	**30%**	**75%**

Figure 2: Potential classifier for Table 1 Frequent Ruleitems

IV. EXPERIMENTAL RESULT

We have performed an empirical study to evaluate its performance of AC's with that of classifiers Decision tree. The experiments Were conducted on a 3.0GHz PC with 2G main memory and running windows XP using java. We tested the classifiers on 2 data sets in UCI Machine Learning Repository. The characteristics of the data set were summarized in Table 2. As can be seen from the table 2, AC outperforms Decision tree on accuracy.

Table 2 Selected UCI datasets

Database	**#instance**	**#attribute**	**#class**	**Decision Tree**	**AC**
Heart	270	13	2	79.6	81.2
Breast	699	10	2	93.3	95

V. CONCLUSION

Associative classification is a relatively new branch in classification. In this paper, we propose an integrating classification and association rules learn from the classified instances. The empirical studies show that our algorithm has good ability and outperforms the rule based classifiers in most cases. Associative classification is a promising approach in data mining. Associative classifiers produce more accurate classification models than traditional classification algorithms such as decision trees and rule induction approaches. One challenge in associative classification is the exponential growth of rules, therefore pruning becomes essential.

REFERENCES

[1]. Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. Mining association rules between sets of items in large databases. In SIGMOD-93, pages 207–216, May 1993.

[2]. R. Agrawal and R. Srikant. Fast algorithm for mining association rules in large databases. In Proceedings 20th International Conference on Very Large Data Bases, pages 478–499, September 1994.

[3]. G. Dong, X. Zhang, L. Wong, and J. Li. CAEP: Classification by Aggregating Emerging Patterns. In Proceedings of Discovery-Science-99, 1999.

[4]. B. Liu, W. Hsu, Y. Ma, Integrating Classification and Association Rule Mining, in: A. Press. (Ed.), In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD'98). (Menlo Park, CA, 80–86, 1998) 443 - 447.

[5]. M.-L. Antonie, O.R. Z, An Associative Classifier based on Positive and Negative Rules, in: Proceedings of the 9th ACM SIGMOD workshop on Research issues in data

[6]. J. R. Quinlan. C4.5: Program for Machine Learning. Morgan Kaufmann, 1992

[7]. K. Ali, S. Manganaris and R. Srikant. Partial Classification Using Association Rules. In Proceedings of KDD-97, 115-118, 1997.

[8]. X. Yin, J. Han, CPAR: Classification based on Predictive Association Rules, in: Proceedings of the Third SIAM International Conference on Data Mining, , . SIAM 2003, (San Francisco, CA, USA, 2003).

[9]. Wang J, Karypis G. HARMONY: Efficiently Mining the Best Rules for Classification. In: Proceedings of the 5th SIAM International Conference on Data Mining, Newport Beach, USA, 2005. 205-216.