Misclassification Penalties in Associative Classification

Sohil Gambhir, Prof. Nikhil Gondaliya

***Abstract*— The data mining is a method to find small amount of useful data from very large amount of data. There are two classical techniques in the field of data mining namely associative rule mining and classical rule mining. In order to have advantages of both a new approach was developed by combining both the methods. This new approach is called associative classification. It has given significant improvement like better accuracy over the conventional classification system e.g. C4.5. There are many methods developed for the associative classification in the due course like CBA, CMAR, CPAR, Hyper Heuristic, and CARGBA. However the effect of the misclassification penalties on the classification has not been examined. Out of all the available methods in associative classification the CPAR has the highest accuracy. This work is a study of effect of the misclassification penalties on the classification process of the associative classification method CPAR. From the many methods available bagging is selected for the misclassification penalty effect. A new approach namely M-CPPAR (Modified CPAR) is proposed in this work. After the study it can be concluded that if misclassification penalty effect is considered during the classification process the accuracy of the CPAR can be improved.**

***Index Terms*— M-CPAR, Misclassification Penalties, Data Mining.**

I. Introduction

The data mining is known is a method to find small amount of useful data from very large amount of data [1].

There are two classical techniques in the field of data mining namely association rule mining and classification rule mining. Both having their own advantages and disadvantages. The new developments have lead to a new approach in the classification called associative classification which is the integration of two techniques called classification rule mining and association rule mining [3].

Classification rule mining aims to discover a small set of rules in the database that forms an accurate classifier. Association rule mining finds all the rules existing in the database that satisfy some minimum support and minimum confidence constraints. For association rule mining, the target of discovery is not pre-determined, while for classification rule mining there is one and only one predetermined target.

Thus, great savings and conveniences to the user could result if the two mining techniques can somehow be integrated. The integration is done by focusing on a special subset of association rules whose right-hand-side is restricted to the classification class attribute. The integration is done by focusing on mining a special subset of association rules, called ***class association rules*** (CARs).

The new combined approach associative classification achieves higher accuracy but lesser speed than traditional classification approaches. There are various methods used for the associative classification [3, 4, 5]. However till today the effect of the misclassification penalties have not been studied on this new approach [6]. This is a study of misclassification penalty of associative classification technique called CPAR (Classification based on Prediction Association Rules) [5].
This adaptation is necessary for two main reasons: A transactional database normally used in association rule mining does not have many associations.

1. While classification data tends to contain a huge number of associations.

Adaptation of the existing association rule mining algorithm to mine only the CARs is needed to reduce the number of rules generated which will avoid combinatorial explosion. The adaptation involves discretizing continuous attributes based on the classification predetermined class target.

Data mining in the associative classification framework thus consists of three steps:

- Discretization of continuous attributes, if any. Discretization can be done using any of standard discretization algorithms available in this standard literature [8, 9, 10, 11, 12].
- Generating all the class association rules (CARs), and
- Building a classifier based on the generated CARs.

The associative classification has three new things:

1. It shows a new way to build accurate classifiers. Results show that classifiers built this way are, in general, more accurate than those produced by the state-of-the-art classification system like C4.5 classification system [2].
2. It makes association rule mining techniques applicable to classification tasks.
3. It helps to solve a number of important problems with the existing classification systems.

Sohil Gambhir is Master's student at the B.V.M. Engineering College, Vallabhvidyanagar, Gujarat, India. (e-mail: gambhir.sohil@gmail.com)
Prof. Nikhil Gondaliya is Associate Professor, Department of I.T. Engineering at G. H. Patel College of Engineering & Technology, Vallabh Vidyanagar, Gujarat, India. (e-mail: nikhilgondaliya@gcet.ac.in)

The major problems of the existing systems are solved as below.

1. Understandability problem

Many rules produced by standard classification systems are difficult to understand. Similarly many understandable rules that exist in the data are left undiscovered.

2. Interesting rule problem

The quest for a small set of rules of the existing classification systems results in many interesting and useful rules not being discovered.

3. Memory Problem

All the standard classification systems need to load the entire database into the main However in this approach the database can reside in the disk rather than the main memory. Hence the memory problem is solved.

Following are the major algorithms for the associative classification.

II. SURVEY OF HISTORICAL ASSOCIATIVE CLASSIFICATION METHODS

A. *CBA (Classification Based On Associations)*

The CBA is an ordered rule algorithm based on convergence analysis. It consists of two parts, a *rule generator* (called CBA-RG), which is based on algorithm Apriori [7] for finding association rules. Another part is a *classifier builder* (called CBA-CB) which generates the classifiers from the rules generated from the CBA-RG.

CBA generates all the association rules with certain support and confidence thresholds as candidate rules. It then selects a small set of the rules from them to form a classifier. At the time of the predication of the class label of the example the best rule (having highest confidence) is used for the classification.

In CBA-RG algorithm the data is scanned multiple times. In this multiple pass all the frequent rule items are generated. Here the rule item means a rule. In the first pass it counts the support and determine the whether it is frequent or not. In each subsequent pass it starts with the seed set of rules found to be frequent in the previous pass. It uses this seed set to generate new possibly frequent rules called the candidate rules. The actual support for these candidate rules are calculated during the pass over the data. At the end of the pass it determines which of the candidate ruleitems are actually frequent and produces the rules (CARs).

The CBA-CB algorithm used to build a classifier using CARs. To produce the best classifier out of the whole set of rules would involve evaluating all the possible subsets of it on the training data and selecting the subset with the right rule sequence that gives the least number of errors. This algorithm is a heuristic one. However, the classifier it builds performs very well as compared to that built by C4.5.

This algorithm satisfies two main conditions:

Condition 1: Each training case is covered by the rule with the highest precedence among the rules that can cover the case.

Condition 2: Every rule correctly classifies at least one remaining training case when it is chosen.

This algorithm is simple, but is inefficient because it needs to make many passes over the database. The experimental results show that data set taken from UCI ML repository [7] 16 out of 26 data sets it working better than the C4.5 classification system [2].

The limitations of this approach are as follows

- It generates huge amount of the mined rule.
- This leads to computational overhead.
- The classification is done based on single high confidence rule which can be biased

B. *CMAR (Classification based on Multiple Association Rules)*

Previous studies propose that associative classification suffers from the huge set of mined rules and sometimes biased classification or over fitting.

Since the classification is based on only single high-confidence rule. Hence another associative classification method, CMAR (Classification based on Multiple Association Rules) [4] is proposed. The classification is performed based on a weighted analysis using multiple strong association rules. CMAR is highly efficient and scalable. The classification is performed based on a weighted X^2 analysis using multiple strong association rules.

CBA also suffer some weakness as shown below.

First it is not easy to identify the most effective rule at classifying a new case.

Second a training data set often generates a huge set of rules.

To solve first problem instead of relying on a single rule for classification, CMAR determines the class label by a set of rules. To avoid bias, a new technique is development, called weighted X^2, which derives a good measure on how strong the rule is under both conditional support and class distribution.

To solve Second, to improve both accuracy and efficiency, CMAR employs a novel data structure, CR-tree, to compactly store and efficiently retrieve a large number of rules for classification.

Third, to speed up the mining of complete set of rules, CMAR adopts a variant of recently developed FP-growth method. FP-growth is much faster than Apriori-like methods.

CMAR consists of two phases: rule generation and classification. In rule generation CMAR computes the complete set of rules in the form of R: P -> C, where p is a pattern in the training data set, and c is a class label such that sup (R) and conf (R) pass the given support and confidence thresholds, respectively.

Furthermore, CMAR prunes some rules and only selects a subset of high quality rules for classification.

In the second phase CMAR extracts a subset of rules matching the object and predicts the class label of the object by analyzing this subset of rules. If all the rules give same class label then it is classified. Otherwise the combined group effect will be taken into consideration.

The CMAR outperforms both C4.5 and CBA on accuracy. The limitations are as follows,

- CMAR is significant advance compare to the CBA but still it is very slower.
- The overall accuracy can be further improved.

C. *CARGBA (Classification based on Association Rule Generated in a Bidirectional Approach)*

The CARGBA generates the rules in two steps. At first, it generates a set of high confidence rules of smaller length with support pruning and then augments this set with some high confidence rules of higher length with support below minimum support. The purpose of this rule generation is not knowledge extraction; rather the only purpose is using these rules for classification to obtain better accuracy.

In the second step, it generates rules that are as specific as possible. These rules have higher length and therefore lower support and thus they easily capture the specific characteristics about the data set. That is, if there is a classification pattern that exists over very few instances or there are instances that are exceptions to the general rule, then these instances will be covered by the specific rules. Since these instances are small in number, specific rules are produced without any support pruning. In short, this approach results in a better mixture of class association rules. All the rules generated by CARGBA rule generator will not be used in the classification. So, the second part builds a classifier with the essential rules and is called CARGBA Classifier Builder.

The experiments on 6 databases in UCI machine learning database repository show that CARGBA is consistent, highly effective at classification of various kinds of databases and has better average classification accuracy in comparison with C4.5, CBA and CMAR.

D. *Hyper Heuristic Approach*

In this investigation is done for the potential of associative classifiers as well as other traditional classifiers such as decision trees and rule inducers in solutions (data sets) produced by a general-purpose optimization heuristic called the hyper heuristic[20]. The hyper heuristic requires deciding which of several simpler search neighborhoods' to apply at each step while constructing a solution. After experimenting 16 different solution generated by a hyper heuristic called Peckish using different classification approaches, the results indicated that associative classification approach is the most applicable approach to such kind of problems with reference to accuracy.

The Peckish hyper heuristic, which is a robust and general-purpose optimization heuristic that requires to decide which of several simpler low-level heuristic techniques to apply at each step while building the schedule. This study focused on analyzing the behavior of low-level heuristics that were selected by the hyper heuristic and improved upon the quality of the current solution in order to extract useful rules. These rules can be used later to quickly predict the appropriate low-level heuristics to call next. The experimental tests showed a better performance for associative classification techniques (MCAR, MMAC, CBA) over decision trees (C4.5), rule induction (RIPPER) and PART algorithm with reference to the accuracy of predicting the appropriate set of low-level heuristics.

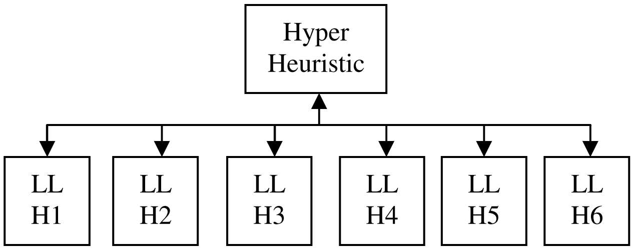


Figure 1 Hyper Heuristic General Framework

E. *CPAR (Classification based on Predictive Association Rules)*

The CPAR [5] is having the finest accuracy in all the associative classification algorithms.

CPAR combines the advantages of both associative classification and traditional rule-based classification. Instead of generating a large number of candidate rules as in associative classification, CPAR adopts a greedy algorithm to generate rules directly from training data.

To avoid over fitting, CPAR uses expected accuracy to evaluate each rule and uses the best k rules in prediction. Associative classification approach often generates a very large number of rules in association rule mining. And also it takes efforts to select high quality rules from among them. CPAR inherits the basic idea of FOIL in rule generation and integrates the features of associative classification in predictive rule analysis.

In comparison with CBA, CPAR has the following advantages:

(1) CPAR generates a much smaller set of high-quality predictive rules directly from the dataset;

(2) To avoid generating redundant rules, CPAR generates each rule by considering the set of "already- generated" rules; and

(3) When predicting the class label of an example, CPAR uses the best k rules that this example satisfies.

(4) CPAR uses dynamic programming to avoid repeated calculation in rule generation; and

(5) When generating rules, instead of selecting only the best literal, all the close-to-the-best literals are selected so that important rules will not be missed.

CPAR generates a smaller set of rules, with higher quality and lower redundancy in comparison with associative classification. As a result, CPAR is much more time efficient in both rule generation and prediction but achieves as high accuracy as associative classification.

III. M-CPAR (MODIFIED CPAR)

In this new proposed and modified CPAR it is taken into consideration the misclassification penalties when developing the classifiers. There are other changes in CPAR also.

The PNArray CPAR is primarily used to reduce the time complexity of the algorithm. Now PNArray implementation is modified as it will only store the positive and negative example from the data. However due to this the overall running time of the algorithm is increased but it has absolutely no effect on the accuracy of the algorithm.

Many methods like Statistical X^2 method Chimerge [11], Chi2 [12], Minimum description length principle [9], Entropy based discretization, Concept hierarchy [10]; All the clustering algorithms can be applied to get the discretization of the continuous value attribute.

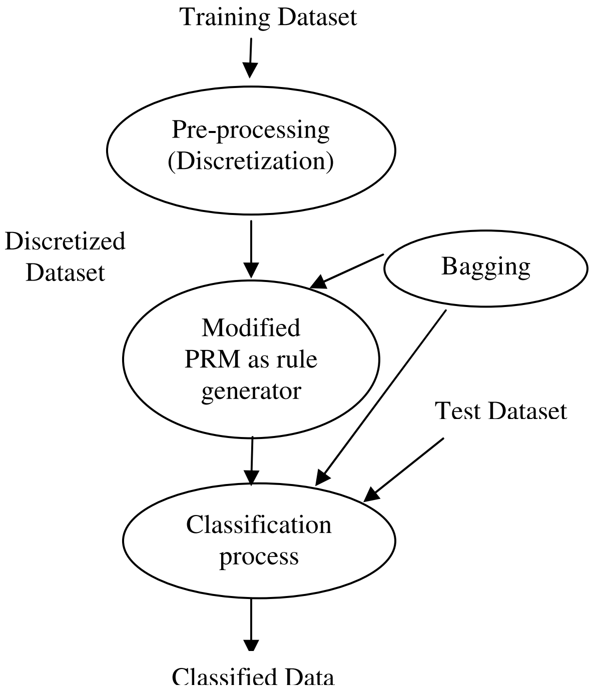


Figure 2 Modified CPAR Process Model

Here as it will more clear in the following section that due to simplicity of the data set, because there are no missing value in the data set, a very simple approach is taken that depending on the standard deviation the all the continuous value attributes are mapped to the continuous valued integers. Means from the starting point of the range of any attribute it is divided into the equal part until the ending point of the range has reached.

A. *Misclassification Penalty*

The misclassification of any future object may cause penalty. If this misclassification penalty is significant then it is better not to classify the object. This fact can be used when developing the classifier. To take into account the misclassification penalties first evaluation or estimation of the accuracy of the classifier has to be done [6]. There are various methods available for the same [23, 24, 25, 26, 27] out of which some are described as follows,

1. Holdout method
2. Random sub sampling
3. k-fold cross validation
4. Bagging (bootstrap aggregation)

1. Holdout method:

In the holdout methods the given data set is randomly partitioned into two independent sets, a training set and a test set. Typically two-third of the data is allocated to the training set and remaining one third of the data is allocated to the test set. The training set is used to derive the classifiers. The classifiers derived in this manner are used to classify the tuples in the test set. This method to estimate the classifier is pessimistic because only some part of the original data was used to derive the classifiers.

2. Random sub-sampling:

The random sub-sampling method is variation of the original Holdout method. In this method the procedure of the Hold out method is repeated k time. The accuracy is said to be average of all the k accuracy obtained from each of the iteration.

3. k-fold cross validation

In k-fold cross validation the initial data is randomly partitioned into k mutually exclusive subsets or "folds". Namely S_1, S_2... S_k, Each of approximately equal size. Training and testing is done k times. In the iteration i the subset Si is reserved as the test set. The remaining subsets are collectively used to train the classifier. This means in the iteration 1 the subset S_1 is reserved as test set while subsets S_2... S_k are used to train the classifier and so on. The accuracy estimate is the overall number of correct classifications from the k iterations, divided by the total number of samples in the initial data.

4 Bagging

The bagging is proposed by [23]. This is how the bagging works. For example there is patient and it is to be diagnosed. Then what can be done that multiple doctors can be consulted. If a certain diagnosis occurs more than others then that is the final result. The similar formula can also be applied to the classification process. The voting is done for each class label for the classification for each tuple. The maximum vote winner will be the predicated class label.

B. *M-CPAR Rule Generation & Classification*

```
set weight of every examples to 1
rule set R $\leftarrow \theta$
totalWeight $\leftarrow$ TotalWeight (P)
A $\leftarrow$Computer ModifiedPNArray from D {Change in M_CPAR}
While TotalWeight(P) > $\partial \bullet$ totalWeight
N' $\leftarrow$N, P' $\leftarrow$ P, A' $\leftarrow$A
  rule r $\leftarrow$emptyrule
```

```
    while true
        find the best literal p according to A'
        if gain(p) < min_gain then break
        append p to r

        for each example t in P' ∪ N' not satisfying r's body
            remove  t from  P' or N'
            change  A' according to the removal of t
        end
    end
    R ← R ∪ {r};
    For each example t in P satisfying r's body
        t· weight ← α· t · weight
        change A according to the weight decreased
    end
end
Return R
```

Algorithm 1 M-CPAR Rule Generation

The effect on the classification process is as follows. After the rule generation is over the classification is done on the sample data. For each tuple the rule classification is done. So there will be major changes in the new classification process as well as the classifier development process. The bagging algorithm works on voting principle. For every tuple each classifier rule is tested. If tuples satisfies rules body then its class prediction is recorded and considered as one vote. At the end the votes for the individual class label is counted and the class label having maximum number of votes is called the predicted class label. Now due to the consideration of the misclassification penalties in the classifier development process and in the classification process the accuracy of the algorithm will increase.

Classification Process

1. Select all the rules whose bodies are satisfied by the example
2. Select best k rule for each class
3. Bagging voting algorithm is used to find the estimated class label {Change M_CPAR}

Algorithm 2 M-CPAR classification algorithm

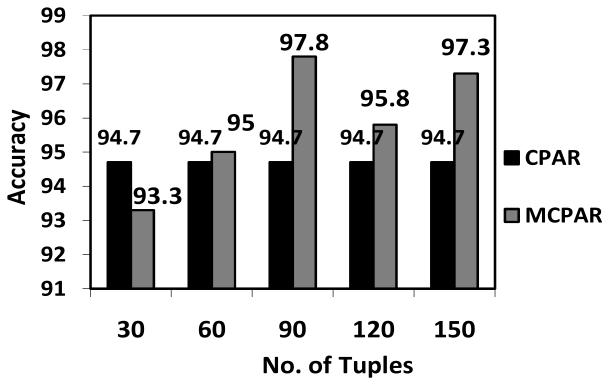
For the implementation purpose various parameters are set as follows.

- Data set used is IRIS data set from UCI ML repository.
- Delta=0.05, min_gain=0.07, alpha=2/3 and voting mechanism is used for the prediction of the class label.

C. Result

Accuracy of the algorithm is found as follows.

No of tuples	CPAR	M-CPAR
	Accuracy	Accuracy
30	94.7	93.3
60	94.7	95.0
90	94.7	97.8
120	94.7	95.8
150	94.7	97.3
Average	94.7	95.8



IV. Conclusion

As a final conclusion from the above stated result the accuracy of the CPAR can be significantly improved if the effect of the misclassification penalty is considered at the time of classification process. This M-CPAR model can be applied to all the data set available in the UCI Machine Learning repository [28].

In future work, this M-CPAR model can be applied to all the associative classification techniques to study the effect of the misclassification penalties on the same.

References

1. Data Mining Concepts and Techniques by Jiawei Han & Micheline Kamber.Pulication Elseiver.
2. C4.5: Programming for machine learning, Morgan by J. R. Quilan.
3. Integrating classification and associative rule mining by Liu B., Hsu W., and Ma W. In KDD'98, New York, NY, Aug. 1998..
4. CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules by Li W., Han J., Pei J. In ICDM'01, pp. 369{376, San Jose, CA, Nov.2001.
5. Classification based on Predictive Association Rules by Yin X., Han J. Proc. 2003 SIAM Int.Conf. on Data Mining (SDM'03), San Fransisco, CA, May 2003.
6. A survey of associative classification algorithms by Dhirendra Kumar Swami, R. C. Jain. ADIT JOURNAL OF ENGINEERING, VOL. 2, NO.1, DECEMBER 2005.
7. Fast algorithms for mining association rules by Rakesh Agarwal and Ramkrishan Srikant. In VLDB'94, Chile, Sept. 1994.
8. Supervised and unsupervised discretization of Continuous features by Ron Kohavi, James Dougherty and Mehran Sahami. Machine Learning Proceedings of theTwelfth International Conference,1995, Morgan Kaufmann Publishers┐ San Francisco┐ CA
9. Multi-interval Discretization of continuous-valued attributes for classification learning by Y. Fayyad and K. Irani.

10. Dynamic Generation and Refinement of Concept hierarchy for KDD by J. Han and Y. Fu. AAAI'94 Workshop on Knowledge Discovery in Databases (KDD'94), Seattle, WA, July 1994, pp. 157-168.
11. ChiMerge: Discretization of Numeric Attributes by Randy Kerber. AAAI'92 page 123-128.
12. Chi2: Feature selection and discretization of numeric attributes by Huan Liu and Rudy Sentiono. Proceedings of IEEE 7th international conference on Tools with AI 1995 page 388-391.
13. A Novel Algorithm for Associative Classification by Gourab Kundu, Sirajum Munir, Md. Faizul Bari, Md. Monirul Islam and K. Murase. In Proc. of the 14th International Conference on Neural Information Processing, Kitakyushu, Japan, November 13-16, 2007.
14. A parameter free associative classification method by Loic Cerf, Dominique Gay, Nazha Selmaoui, and Jean-Francois Boulicaut. Proceedings of DaWaK 2008 page 293-304.
15. Mining Data from a hyperheuristic approach using associative classification by Fadi Thabtah and Peter Cowling. Expert Systems with Applications: An International Journal Volume 34 Issue 2, February, 2008 Pages 1093-1101.
16. A lazy approach to Associative Classification by Elena Baralis, Silvia Chiusano, and Paolo Garza. IEEE Trans. Knowl. Data Eng. 20(2) page 156-171 (2008)
17. Fast effective rule induction by Cohen W. ICML 1995: 115-123.
18. Incremental reduced error pruning by Furnkranz, J., & Widmer, G. Proceedings of the 11th International Conference on Machine Learning (ML-94), pp. 70{77, New Brunswick, NJ, 1994. Morgan Kaufmann
19. Generating accurate rule sets without global optimization by Frank, E., & Witten, I.. Proc International Conference on Machine Learning, 1998
20. Hyperheuristics for managing a large collection of low level heuristics to schedule personnel by Cowling, P., & Chakhlevitch, K.. Adaptive and Multilevel Metaheuristics, SCI 136, pp. 3–29, 2008
21. FOIL: A midterm report by Quinlan and Cameron-Jones. In Proceedings of the European Conference on Machine Learning 1993
22. Bagging predicators by Leo Breiman. Machine Learning volume 24 issue 2, page 123 – 140 August 1996.
23. A decision-theoretic generalization of on-line learning and an application to boosting by Freund and Schapire. J. Comput. Syst. Sci. 55(1): 119-139 (1997).
24. Bagging, Boosting and C4.5 by Quilan. AAAI/IAAI, Vol. 1 1996: 725-730.
25. A study of cross validation and bootstrap for accuracy estimation and model selection by R. Kohavi. Proceeding JCAI'95 Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2.
26. Boosting and Naive Bayesian Learning by C. Elkan. Technical Report No. CS97-557, September 1997.
27. Introduction to bootstrap methods by Robert Stine. Socioligical Methods and Research, vol. 18, nos. 2 & 3, November 1989.
28. Data Set provided by UCI Machine Learning repository web reference is http://www.ics.uci.edu/~mlearn/MLRepository.html.