Using Classification to Evaluate the Output of Confidence-Based Association Rule Mining

Stefan Mutter[1,2], Mark Hall[2], and Eibe Frank[2]

[1]Department of Computer Science
University of Freiburg
Freiburg, Germany
mutter@informatik.uni-freiburg.de

[2]Department of Computer Science
University of Waikato
Hamilton, New Zealand
{mhall, eibe}@cs.waikato.ac.nz

Abstract. Association rule mining is a data mining technique that reveals interesting relationships in a database. Existing approaches employ different parameters to search for interesting rules. This fact and the large number of rules make it difficult to compare the output of confidence-based association rule miners. This paper explores the use of classification performance as a metric for evaluating their output. Previous work on forming classifiers from association rules has focussed on accurate classification, whereas we concentrate on using the properties of the resulting classifiers as a basis for comparing confidence-based association rule learners. Therefore, we present experimental results on 12 UCI datasets showing that the quality of small rule sets generated by Apriori can be improved by using the predictive Apriori algorithm. We also show that CBA, the standard method for classification using association rules, is generally inferior to standard rule learners concerning both running time and size of rule sets.

1 Introduction

Association rule mining is a widely-used approach in data mining. Association rules are capable of revealing all interesting relationships in a potentially large database. However, a major problem in association rule mining is its complexity. Even for moderate-sized databases it is intractable to find all the relationships. This is why a mining approach defines an interestingness measure to guide the search and prune the search space. Therefore, the result of an association rule mining algorithm is not the set of all possible relationships, but the set of all interesting ones. The definition of the term interesting, however, depends on the application. The different interestingness measures and the large number of rules make it difficult to compare the output of different association rule mining algorithms. There is a lack of comparison measures for the quality of association rule mining algorithms and their interestingness measures. In this paper we focus on *confidence-based* association rule mining. Here the term 'confidence-based' refers to those association rule learners that use the confidence (i.e. accuracy) of a rule as the interestingness measure.

Association rule mining algorithms are often compared using time complexity. That is an important issue of the mining process, but the quality of the resulting rule set is ignored. On the other hand there are approaches that investigate the discriminating power of association rules and use them to solve classification problems. This research area is called *classification using association rules*. An important aspect of classification using association rules is that it can provide quality measures for the output of the underlying mining process. The properties of the resulting classifier can be the basis for comparisons between different confidence-based association rule mining algorithms. A certain mining algorithm is preferable when the mined rule set forms a more accurate, compact and stable classifier in an efficient way.

The introduction of these quality measures—particularly the accuracy of the classifier—kills two birds with one stone. First, we are interested in the comparison of the quality of different confidence-based mining algorithms. Therefore, we use classification using association rules. Secondly, classification using association rules can be improved itself by using a mining algorithm that prefers highly accurate rules. In this paper we compare two different rule mining strategies combined with several different approaches to classification using association rules.

The rest of the paper is organised as follows. In Section 2 we explain the mining, pruning, and classification algorithms used for our study. Section 3 describes our experimental methodology, followed by a discussion of results. We present our conclusions in Section 4.

2 Algorithms

The algorithmic approach for classification using association rules can be divided into three fundamental parts: association rule mining, pruning, and classification.

2.1 Association rule mining

We used two different methods for rule mining—the Apriori algorithm introduced by Agrawal and Srikant [1] that has become a standard approach and the predictive Apriori algorithm published by Scheffer [11]. The most significant difference between the two relates to how the interestingness of an association rule is measured. Although both are confidence-based, the confidence is estimated differently. Both algorithms start the same way by building frequent item sets. An item set is called frequent when its support is above a predefined minimum support. It is possible to construct frequent item sets in reasonable time because of their support-based downward closure. An item set X of length k is frequent if and only if all subsets of X with length $k-1$ are frequent. This property allows the search space to be pruned substantially.

Rule discovery using Apriori is straightforward. For every frequent item set f and every non-empty subset s of f, Apriori outputs a rule of the form $s \Rightarrow (f - s)$ if and only if the confidence of that rule is above the user-specified threshold. The confidence is simply the accuracy of the rule. The rules are ranked according to this confidence value. If two or more rules share the same confidence they are ordered using their support and secondly the time of discovery. Hence Apriori's interestingness measure is the confidence of a rule.

The interestingness measure of predictive Apriori suits the requirements of a classification task. It tries to maximise the expected accuracy of an association rule instead of the accuracy on the training data (as measured by the simple confidence measure in

Apriori). The probability of a correct prediction given the database under consideration is called the predictive accuracy. Scheffer [11] defines it in the following way:

Let D be a database whose individual records r are generated by a static process P, let $X \Rightarrow Y$ be an association rule. The predictive accuracy $c(X \Rightarrow Y) = Pr(r \text{ satisfies } Y | r \text{ satisfies } X)$ is the conditional probability of $Y \subseteq r$ given that $X \subseteq r$ when the distribution of r is governed by P [11].

The confidence $\hat{c}(X \Rightarrow Y)$ of an association rule $X \Rightarrow Y$ is the relative frequency of a correct classification in the training database. Hence the confidence value is optimistically biased. However, it can be corrected using the support of the rule under consideration by computing the expected predictive accuracy $E(c(r)|\hat{c}(r), s(X))$ of a rule r $X \Rightarrow Y$ given its confidence $\hat{c}$ and the support of the rule body $s(X)$. Scheffer [11] calculates it as

$$E(c(r)|\hat{c}(r), s(X)) = \frac{\int cB[c, s(X)](\hat{c}(r))P(c)dc}{\int B[c, s(X)](\hat{c}(r))P(c)dc}$$

using Bayes formula and exploiting the fact that the likelihood $P(\hat{c}(r)|c(r), s(X))$ can be modeled by the binomial distribution $B[c, s(X)](\hat{c}(r))$. $P(c)$ is the prior distribution. We use discretisation to approximate these integrals. Scheffer suggests dividing the predictive accuracy c into 100 discrete intervals and using the midpoint of each interval for calculation.[1] Because of discretisation the above integrals become sums over all possible (100) values of c. The second problem to cope with is the prior distribution. In order to estimate it, for each discretised predictive accuracy value, the fraction of association rules which have this accuracy is counted. Thus, to derive the real prior distribution we would have to investigate the space of all association rules which is intractable. The solution is to draw many association rules at random under the uniform distribution. Scheffer suggests drawing 1000 random association rules for each possible length.[1] In addition, predictive Apriori has an inherent pruning strategy that prefers more general rules. Therefore it searches for the n best rules according to the following criteria:

1. the predictive accuracy of the rule is among the n best **and**
2. it is not subsumed by a rule with at least the same predictive accuracy

The parameter n is user specified. These criteria differ slightly from the ones published in the predictive Apriori paper [11] where all n best rules are not subsumed by any other rule in that list. This version of the predictive Apriori algorithm originates from an unpublished manuscript by Scheffer [10]. In our opinion it better suits the requirements of a classification task, because highly accurate rules which are subsumed by less accurate ones remain part of the output.

In contrast to Apriori, predictive Apriori uses an increasing support threshold. Out of each frequent item set rules are generated and added to the best n rules so far if the above conditions apply. If one rule has to be deleted out of the list of the best n because of subsumption, we have to recursively re-run the rule generating procedure because another rule which previously did not make it into the best n could now qualify for inclusion.

[1] In personal communication via e-mail.

1. Input: rule set sorted according to interestingness measure
2. **For** all rules r in sorted order **DO:**
3. **For** all training instances d **DO:**
4. **If** r covers d **Then**
5. mark d.
6. **If** r classifies d correctly **Then**
7. mark r.
8. **If** r is marked **Then**
9. delete all instances d that are marked.
10. insert r into an intermediate classifier C
11. select the majority class label out of the training instances d as default.
12. **For all** training instances **DO:**
13. classify using C and count the total number of errors made by C.
14. **Else**
15. delete all marks from each training instance d.
16. Find the rule r_{min} with the lowest number of errors.
17. From r_{min} accept all subsequent rules as long as the error does not increase. The last rule accepted is r_{stop}.
18. Delete all rules after r_{stop}.
19. The default class associated with r_{stop} is the default class of the classifier.

Fig. 1. The CBA pruning algorithm. The scheme is an adapted version of Liu et al. [6]

Note that association rules in their general form cannot be used directly for a classification task. The head Y of an association rule $X \Rightarrow Y$ (with rule body X) has to be restricted to one attribute-value-pair. The attribute of this attribute-value-pair is consequently the class attribute. An association rule of this form is called a *class association rule*. We have adapted both algorithms to mine class association rules. For Apriori, we used the method described by Liu et al. [6]. The training data is divided into n subsets—one for each class. Frequent item sets are then found for each subset separately. For each frequent item set a rule is generated by appending a consequent which is the class label of the subset from which it was mined.

Predictive Apriori constructs rules out of frequent item sets generated from all the data (after the class attribute has been deleted) and considers generating a rule for each value of the class attribute for every frequent item set. As before, a rule is kept if its expected predictive accuracy is among the n best. In addition, the prior distribution has to be estimated for class association rules.

2.2 Pruning

Pruning is an essential step in classification using association rules and a crucial difference between existing approaches. A very simple strategy is to bound the number of rules without any closer inspection of the rules themselves. An advantage of this strategy in a comparative study of association rule mining algorithms is that it does not change the sort order induced by the rule miner.

The second pruning approach used in this paper is the pruning method introduced by Liu et al. [6] in their CBA algorithm. It consists of an obligatory and an optional pruning step. The obligatory pruning step is summarised in Figure 1. Our implementation of CBA cannot reproduce all the results from the CBA paper [6]. However, using the algorithmic step in line 17 of Figure 1 (deleting all rules after r_{stop} and not after r_{min} as described in the CBA paper [6]) we were able to reproduce some of the results. The optional pruning step is based on the pessimistic error-rate-based pruning as introduced by Quinlan [8]. It works the following way: for every mined rule r $X \Rightarrow Y$ where

$X = x_1 \wedge \ldots \wedge x_n$ is a conjunction of items x_i, the algorithm checks to see if removing any single item from rule r results in a reduction of the pessimistic error rate. If so, r is pruned. When used, error-based pruning is always performed as a preprocessing step before the obligatory pruning step.

2.3 Classification

For a set of classification rules there are two fundamental approaches to using them for classification: a weighted vote algorithm or a decision list classifier.

The simplest voting scheme is a majority vote where each rule r is equally weighted with weight $\omega(r) = 1$. In addition we explored a weighting scheme using the inverse function $f(x) = \frac{1}{x}$ for weighting. For each rule r its weight $\omega(r)$ is calculated using:

$$\omega(r) = \frac{1}{rank(r)} \text{ where } rank(r) \in \{1, \ldots, R\} \text{ and } R \text{ is the number of rules.}$$

This weighting scheme emphasises the difference between the top-ranked rules and the bottom-ranked ones, because rules at the bottom of the sorted list only get a little weight and so only have a small influence.

The standard classifier for classification using association rules—the CBA algorithm [6]—is a decision list classifier. Compared to pruning, classification is very simple in CBA. It just searches the pruned and ordered list for the first rule that covers the instance to be classified. The prediction is the class label of that classification rule. If no rule covers the instance, CBA uses the default class calculated during pruning (see Figure 1). If the decision list is empty, the majority class of the training instances will be assigned to each test instance as default. CBA uses a rule limit. It stops the search for association rules after investigating 80000 rules regardless whether they fulfil the requirements of the minimum support and confidence or not. In our experiments we did not use this cutoff—instead we varied the number of rules mined in order to better explore the behaviour of the association rule learners.

3 Experiments

For our experiments we used 12 datasets from the UCI repository [2], shown in Table 1. A rule mining strategy is preferable if it allows a compact and accurate classifier to be formed from the mined set of rules. Therefore we used the following performance measures:

- the percentage of instances correctly classified in the test set,
- the average rank of the first rule that covers and correctly predicts a test instance,
- the number of mined rules generated by a class association rule miner,
- the number of rules after the pruning step,
- the time required for mining, and
- the time required for pruning.

Dataset	Instances	Numeric attributes	Binary attributes	Nominal attributes	Classes	Missing values (%)
balance	625	4	0	0	3	0.0
breast-w	699	0	0	9	2	0.3
ecoli	336	7	0	0	8	0.0
glass	214	9	0	0	6	0.0
heart-h	294	6	3	4	2	20.4
iris	150	4	0	0	3	0.0
labor	57	8	3	5	2	35.7
led7	1000	0	7	0	10	0.0
lenses	24	0	0	4	3	0.0
pima	768	8	0	0	2	0.0
tic-tac-toe	958	0	0	9	2	0.0
wine	178	13	0	0	3	0.0

Table 1. The UCI datasets used for the experiments and their properties. In the led7 dataset 10% of the instances are noisy.

In every experiment the support threshold s_{min} of Apriori was set to 1% of all instances and the confidence threshold $\hat{c}_{min}$ was set to 0.5—the standard thresholds for support and confidence. When indicated, we restricted the number n of mined rules for Apriori and predictive Apriori (the only input parameter for predictive Apriori).

Class association rule mining as well as association rule mining in general is only possible for nominal attributes.[2] Therefore we needed to discretise the numeric attributes in our datasets for these methods. We used Fayyad and Irani's [4] maximum entropy method for discretisation. To process missing values at classification time, they were treated as different from all other attribute values. All experimental results were obtained using one ten-fold stratified cross-validation. We report statistically significant results at the 5% significance level using the corrected resampled t-test proposed by Nadeau and Bengio [7] and applying a Bonferroni adjustment as suggested by Salzberg [9]. In addition we show the standard deviation for each result. All algorithms used in the experiments were implemented within the WEKA machine learning framework [12].[3]

3.1 Compactness of rule sets

As Table 2 shows, Apriori mines considerably more rules than predictive Apriori but most of them are pruned in the final set of classification rules. The advantage of predictive Apriori is that it generates fewer rules right from the start.

3.2 Rule ranking

In this section we investigate whether highly accurate and general rules are preferred by the ranking system of the mining algorithms. To this end we measured the average rank of the first rule that covered an instance and predicted it correctly. For this comparison, both association rule mining algorithms mined the 100 highest ranked rules for every dataset. We calculated a default class which is the majority class in the training

[2] Association rule mining for numeric-valued attributes is still an open area of research.
[3] WEKA is available from http://www.cs.waikato.ac.nz/ml/weka.

Dataset	Apriori	Pred. Apriori
balance	72.2 ± 13.07	78.7 ± 10.29 •
breast-w	5124.9 ± 65.50	906.8 ±325.34 ◦
ecoli	888.2 ± 152.35	304.4 ± 22.51 ◦
glass	6055.2 ± 454.27	472.1 ±144.56 ◦
heart-h	19886.8 ± 757.08	324.9 ± 18.71 ◦
iris	96.5 ± 14.25	29.8 ± 8.34 ◦
labor	96084.3 ±5569.42	228.9 ± 21.79 ◦
led7	510.6 ± 13.50	1565.9 ± 20.44 •
lenses	121.8 ± 3.52	28.8 ± 10.03 ◦
pima	3311.4 ± 311.11	179.5 ± 43.16 ◦
tic-tac-toe	7642.5 ± 42.23	6289.6 ±105.04 ◦
wine	87427.9 ±5066.97	1034.2 ±714.36 ◦

◦, • statistically significant improvement or degradation

(a)

Dataset	Apriori	Pred. Apriori
balance	28.9 ± 2.23	50.9 ± 2.85 •
breast-w	2975.9 ± 105.44	137.9 ± 46.91 ◦
ecoli	333.2 ± 24.52	221.2 ± 16.45 ◦
glass	1661.6 ± 161.29	307.6 ± 93.20 ◦
heart-h	1242.2 ± 62.28	188.4 ± 10.31 ◦
iris	28.4 ± 3.27	13.5 ± 1.35 ◦
labor	79371.8 ±4747.00	62.0 ± 7.90 ◦
led7	330.5 ± 15.76	1053.3 ± 22.18 •
lenses	31.4 ± 2.91	10.6 ± 4.43 ◦
pima	461.0 ± 46.47	87.0 ± 18.05 ◦
tic-tac-toe	1180.8 ± 17.84	1731.9 ± 28.83 •
wine	36396.5 ±3710.72	554.3 ±383.79 ◦

◦, • statistically significant improvement or degradation

(b)

Dataset	Apriori	Pred. Apriori
balance	14.1 ±0.57	14.1 ± 0.57
breast-w	103.5 ±4.84	37.8 ±11.28 ◦
ecoli	19.8 ±2.35	20.3 ± 1.83
glass	27.3 ±2.45	22.7 ± 2.87 ◦
heart-h	38.8 ±3.39	33.1 ± 2.85 ◦
iris	5.6 ±0.84	5.6 ± 0.70
labor	26.7 ±2.45	8.7 ± 1.83 ◦
led7	60.4 ±3.24	62.0 ± 2.91 •
lenses	20.2 ±1.03	2.6 ± 2.76 ◦
pima	39.7 ±6.62	39.0 ± 6.63
tic-tac-toe	155.8 ±5.92	166.6 ± 6.52 •
wine	30.7 ±5.06	7.6 ± 5.27 ◦

◦, • statistically significant improvement or degradation

(c)

Table 2. The compactness of the rule sets mined by different association rule mining algorithms. Table (a) compares the number of mined rules when we do not restrict their number. The number of rules left after error-based pruning is shown in (b) and (c) shows the size of the final rule set used by CBA when it receives the rules from Table (b) as input.

instances. If no rule covered a test instance and the default class predicted the instance correctly, the count was set to the number of mined rules incremented by one, otherwise it was set to the number of mined rules incremented by two. The results are summarised in Figure 2. Predictive Apriori outperforms Apriori in two third of the datasets. This fact indicates that predictive Apriori tends to rank high quality rules higher. The problem with using Apriori's confidence-based ordering without a support-based correction is that rules with low support tend to have high confidence. These rules are very specific and prone to noise.

3.3 Accuracy

In the context of a comparison of mining algorithms a weighted classifier with non-uniform weights emphasises the importance of a good rule ranking induced by the mining algorithms. This is because the higher the ranking of a rule is, the more strongly weighted is its prediction. The overall accuracy on a test set is higher for that mining algorithm that has a better sorting of the mined rules according to their accuracy. Figure 3 illustrates the results.

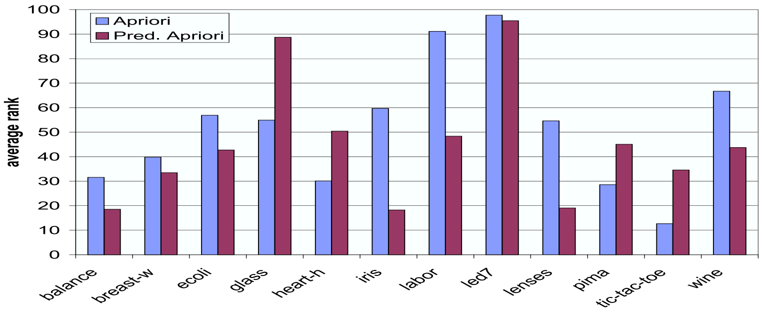

Fig. 2. The differences in the rule ranking between the mining algorithms. Apriori and predictive Apriori both mine the best 100 rules. The figure shows the average rank of the first rule that covers and correctly predicts a test instance. All differences except for breast-w, pima and wine are significant at the 5% level.

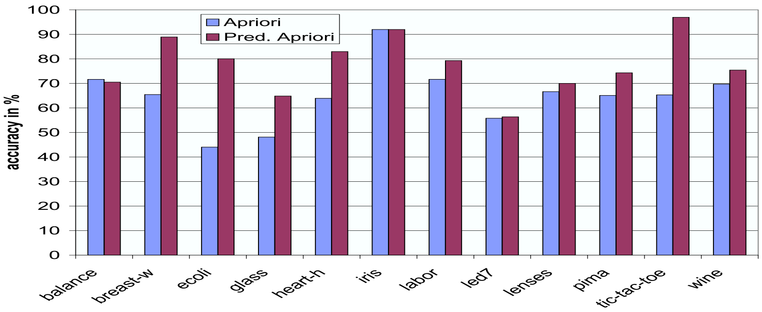

Fig. 3. The accuracies for different association rule mining algorithms using inversely weighted classifiers. The number n of mined rules was restricted to 100. Differences for breast-w, ecoli, heart-h, pima and tic-tac-toe are significant at the 5% level.

The predictive Apriori algorithm with an inversely weighted classifier outperforms Apriori on every data set except for balance. For both algorithms the number of mined rules was restricted to 100. The inverse weighting function relies on the discriminative power of the top ranked rules, because the classifier assigns much higher weights to them. This fact indicates that the predictive Apriori algorithm tends to rank high-quality rules at the top.

The next classification algorithm we used to compare our rule mining strategies is the majority vote algorithm. The main focus in this comparison is different than in the previous case. Here we are not interested in the individual ordering. A majority vote algorithm cannot provide any insights into this, because every rule has the same weight independent of the ranking. Instead, the majority vote algorithm can be used to evaluate how many rules are necessary to build an accurate classifier. The results can be found in Table 3. They reveal an interesting property. For a small number of mined rules the accuracy using predictive Apriori is higher than that for Apriori. To build a comparable

Data Set	Pred. Apriori $n = 10$	Apriori $n = 10$	Pred. Apriori $n = 100$	Apriori $n = 100$	Pred. Apriori $n = 200$	Apriori $n = 200$
balance	73.92± 5.20	74.40± 6.63	73.29± 5.41	76.46± 4.81	73.29± 5.41	76.46± 4.81
breast-w	65.47± 0.47	65.47± 0.47	88.65± 8.65	65.47± 0.47 ●	90.94± 9.24	65.47± 0.47 ●
ecoli	68.14± 7.72	42.56± 1.21 ●	76.81± 2.85	43.44± 2.56 ●	73.23± 5.32	68.12± 5.09
glass	49.09± 3.53	50.04± 9.11	61.06± 8.38	48.59± 5.80 ●	71.04± 7.88	46.71±11.90 ●
heart-h	77.52± 5.29	63.95± 1.43 ●	82.00± 6.34	63.95± 1.43 ●	84.38± 7.28	63.95± 1.43 ●
iris	90.67± 9.00	90.00±12.67	91.33± 9.45	92.67± 7.34	91.33± 9.45	92.00± 7.57
labor	80.67±15.46	64.67± 3.22 ●	86.67±15.32	73.33±15.07 ●	90.00±14.05	81.00±17.36
led7	27.60± 2.59	23.90± 2.51 ●	56.20± 4.57	56.10± 3.78	67.40± 5.19	64.40± 4.88 ●
lenses	70.00±28.11	63.33±32.20	68.33±33.75	68.33±33.75	68.33±33.75	68.33±33.75
pima	65.89± 1.71	65.11± 0.36	73.96± 4.18	65.11± 0.36 ●	72.93± 5.31	67.18± 4.76 ●
tic-tac-toe	74.21± 2.48	65.34± 0.43 ●	92.48± 3.09	65.34± 0.43 ●	85.38± 3.86	67.22± 1.51 ●
wine	65.36±21.14	65.23±10.53	77.71±26.79	66.90± 7.90	78.82±27.65	52.97±13.27 ●
Data Set	**Pred. Apriori $n = 500$**	**Apriori $n = 500$**	**Pred. Apriori $n = 700$**	**Apriori $n = 700$**	**Pred. Apriori $n = 1000$**	**Apriori $n = 1000$**
balance	73.29± 5.41	76.46± 4.81	73.29± 5.41	76.46± 4.81	73.29± 5.41	76.46± 4.81
breast-w	92.51± 9.95	65.90± 1.51 ●	91.65± 9.83	85.11± 3.76	90.36± 9.69	88.55± 4.03
ecoli	77.08± 6.04	74.98± 4.75	77.08± 6.04	79.18± 4.09	77.08± 6.04	79.77± 2.25
glass	63.9o± 7.56	60.24± 7.30	61.15± 4.94	63.10± 6.30	61.15± 4.94	63.48± 7.62
heart-h	82.34± 8.54	71.85±10.34 ●	82.34± 8.54	79.99± 5.42	82.34± 8.54	83.02± 6.35
iris	91.33± 9.45	92.00± 7.57	91.33± 9.45	92.00± 7.57	91.33± 9.45	92.00± 7.57
labor	80.67±17.34	76.00±20.05	80.67±17.34	77.67±24.09	80.67±17.34	76.00±17.69
led7	69.80± 4.42	72.40± 4.38 ○	74.60± 5.36	72.40± 4.50 ●	74.20± 5.07	72.40± 4.50
lenses	68.33±33.75	68.33±33.75	68.33±33.75	68.33±33.75	68.33±33.75	68.33±33.75
pima	74.49± 6.30	72.14± 4.51	74.49± 6.30	72.53± 4.91	74.49± 6.30	70.71± 3.59
tic-tac-toe	76.62± 2.39	97.91± 1.64 ○	75.79± 2.26	97.39± 1.80 ○	72.55± 1.49	98.01± 1.67 ○
wine	77.71±26.92	65.13±11.47	73.79±24.64	66.34±11.89	70.39±22.11	69.64±18.40

○, ● statistically significant improvement or degradation

Table 3. The accuracies of different mining algorithms using a majority vote classifier. The number n of mined rules was varied. The information about statistical significance is calculated separately for each n.

classifier with Apriori's rules the majority vote classifier needs more classification rules. A compact rule set built by predictive Apriori is better than a compact one built with Apriori. Apriori is able to catch up when the number of rules is increased. Using the best 1000 rules, Apriori has one significant win over predictive Apriori and no significant losses.

We also used a CBA decision list instead of majority voting. In this case—when the main focus is on different mining algorithms—we used CBA without error-based pruning. We observed the same behaviour as for the majority vote algorithm. For a small number of rules predictive Apriori outperforms Apriori as shown in Table 4.

3.4 Comparison with standard classification techniques

We compared CBA to three standard techniques—one decision tree learner and two rule learners. The standard decision tree inducer C4.5 [8] was used. Every path in the tree from the root to a leaf node is considered as a rule. We also included JRip, WEKA's implementation of Cohen's RIPPER algorithm [3], and the PART algorithm for constructing decision lists from partial trees [5]. All three algorithms used their default parameters. In this section we compare the compactness and the accuracy of class association rules to that of the standard techniques.

Table 5 shows the results for the compactness, i.e. the number of rules used for classification. Even though using predictive Apriori results in a slightly smaller set of rules,

Data Set	Pred. Apriori	Apriori	Pred. Apriori	Apriori	Pred. Apriori	Apriori
	$n = 10$		$n = 100$		$n = 200$	
balance	66.73± 7.77	68.15± 4.74	71.50± 5.97	71.50± 5.97	71.50± 5.97	71.50± 5.97
breast-w	93.97± 3.40	89.40± 3.51•	94.25± 3.55	90.10± 4.32•	93.11± 3.20	94.97± 3.29
ecoli	68.72± 8.98	61.91± 2.63•	80.37± 2.43	62.79± 3.54•	80.07± 2.72	74.99± 5.31•
glass	47.66± 5.24	51.41± 8.60	70.56± 7.94	51.45± 5.79•	71.97± 8.77	49.11±11.21•
heart-h	77.86± 4.47	64.98± 1.19•	81.66± 7.03	68.36± 5.39•	79.94± 7.12	65.67± 3.86•
iris	90.67±10.04	93.33± 5.44	92.67± 6.63	92.67± 6.63	92.67± 6.63	92.67± 6.63
labor	67.00±12.81	70.00 ±11.86	81.00±20.61	74.00 ±19.93	79.00±19.50	72.67±16.54
led7	27.50± 2.59	23.90± 2.51•	62.40± 4.01	61.10± 4.72•	71.40± 5.50	68.70± 5.21•
lenses	68.33±33.75	61.67±28.38	63.33±32.20	66.67±30.43	63.33±32.20	66.67±30.43
pima	65.89± 1.71	65.11± 0.36	73.44± 4.81	65.11± 0.36•	74.09± 4.18	68.23± 5.62•
tic-tac-toe	100.00± 0.00	69.84± 2.49•	100.00± 0.00	74.85± 5.31•	100.00± 0.00	100.00± 0.00
wine	76.05±19.91	78.10±12.45	76.60±20.88	87.03± 7.61	79.38±21.25	86.05±10.21
Data Set	Pred. Apriori	Apriori	Pred. Apriori	Apriori	Pred. Apriori	Apriori
	$n = 500$		$n = 1000$		n not restricted	
balance	71.50± 5.97	71.50± 5.97	71.50± 5.97	71.50± 5.97	71.50± 5.97	71.50± 5.97
breast-w	93.11± 3.26	95.40± 3.53○	93.97± 3.55	95.55± 2.92○	93.97± 3.55	95.13± 3.03
ecoli	80.07± 2.72	79.47± 7.12	80.07± 2.72	80.65± 3.24	80.07± 2.72	80.65± 3.24
glass	71.97± 8.77	57.08± 7.77•	71.97± 8.77	61.26±10.88•	71.97± 8.77	71.97± 8.77
heart-h	80.29± 6.39	76.60± 9.10	80.29± 6.39	80.97± 7.51	80.29± 6.39	80.63± 7.20
iris	92.67± 6.63	92.67± 6.63	92.67± 6.63	92.67± 6.63	92.67± 6.63	92.67± 6.63
labor	79.00±19.50	68.00± 8.64	79.00±19.50	77.67±19.88	79.00 ±19.50	79.00±19.50
led7	72.00± 5.16	72.30± 5.10	72.00± 5.16	72.30± 5.10	72.00± 5.16	72.30± 5.10
lenses	63.33±32.20	66.67±30.43	63.33±32.20	66.67±30.43	63.33±32.20	66.67±30.43
pima	74.09± 4.18	71.23± 3.83	74.09± 4.18	74.09± 4.81	74.09± 4.18	74.10± 4.48
tic-tac-toe	100.00± 0.00	100.00± 0.00	100.00± 0.00	100.00± 0.00	98.85± 1.44	99.06± 1.25
wine	79.38±21.25	87.12± 9.05	79.38±21.25	91.60± 7.08	79.38±21.25	93.82± 6.71

○, • statistically significant improvement or degradation

Table 4. The accuracies of different mining algorithms using CBA without error-based pruning. The number n of mined rules was varied. The information about statistical significance is calculated separately for each n.

the result of the comparison to the standard techniques is clear. The three algorithms C4.5, JRip and PART generate fewer rules than CBA. However, with respect to accuracy, Table 6 shows that CBA achieves results comparable to the standard rule learners.

3.5 Time complexity

Pruning a rule set output by predictive Apriori takes less time than pruning one output by Apriori, because predictive Apriori prunes away many rules during the search using its inherent pruning strategy (see Section 2.1). Nonetheless, predictive Apriori is much slower during mining. Table 7 shows the results. In comparison to standard learners, it also shows that learning classifiers based on class association rules involves considerably more computation. The numbers are seconds of runtime on a 2.60 GHz Pentium(R) 4 with 1 GB of memory.

4 Conclusions

This paper has provided a comparative study of classification using association rules. In particular we have shown how to use this classification approach to evaluate the quality of a set of association rules generated by a confidence-based miner.

Dataset	CBA + Apriori	C4.5	JRip	PART	CBA + pred. Apriori
balance	14.10±0.57	39.60±6.00 ●	12.20±2.53	37.69±4.17 ●	14.10± 0.57
breast-w	103.50±4.84	28.00±6.00 ○	12.60±1.35 ○	10.10±3.03 ○	37.80±11.28 ○
ecoli	19.80±2.35	18.30±1.89	9.00±1.41 ○	13.60±2.17 ○	20.30± 1.83
glass	27.30±2.45	22.60±2.76 ○	8.20±0.92 ○	14.50±1.27 ○	22.70± 2.87 ○
heart-h	38.80±3.39	5.40±2.55 ○	3.30±0.67 ○	7.20±1.55 ○	33.10± 2.85 ○
iris	5.60±0.84	4.70±0.48 ○	3.90±0.74 ○	3.80±1.32 ○	5.60± 0.70
labor	26.70±2.45	3.60±1.43 ○	3.60±0.52 ○	3.40±0.84 ○	8.70± 1.83 ○
led7	60.40±3.24	29.10±2.51 ○	15.20±1.55 ○	28.60±1.65 ○	62.00± 2.91 ●
lenses	20.20±1.03	3.60±0.52 ○	3.00±0.00 ○	3.60±0.52 ○	2.60± 2.76 ○
pima	39.70±6.62	19.20±6.53 ○	3.30±0.67 ○	7.50±1.18 ○	39.00± 6.63
tic-tac-toe	155.80±5.92	90.60±4.50 ○	10.60±1.43 ○	40.20±3.77 ○	166.60± 6.52 ●
wine	30.70±5.06	5.40±0.70 ○	3.90±0.57 ○	4.60±0.70 ○	7.60± 5.27 ○

○, ● statistically significant improvement or degradation

Table 5. The compactness of CBA compared to standard techniques. CBA performed error-based pruning and did not restrict the number of rules output by the association rule miner.

Dataset	CBA + Apriori	C4.5	JRip	PART	CBA + pred. Apriori
balance	71.50± 5.97	76.65± 3.77 ○	80.80± 3.56 ○	83.54± 4.93 ○	71.50± 5.97
breast-w	94.12± 3.55	94.69± 2.46	94.13± 2.73	94.26± 3.79	92.25± 3.92 ●
ecoli	81.26± 3.08	84.23± 7.51	82.16± 6.63	83.60± 6.36	80.96± 2.78
glass	70.13± 9.19	66.75± 7.94	68.66± 8.74	68.14± 7.21	68.25± 8.02
heart-h	79.98± 8.26	81.07±11.22	78.95± 9.30	81.02± 7.55	79.97± 9.03
iris	94.00± 5.84	96.00± 5.62	94.67± 6.13	94.00± 5.84	94.00± 5.84
labor	81.33±21.44	73.67±22.52	77.00±19.53	78.67±17.58	79.33±18.84
led7	72.30± 5.10	72.40± 4.55	71.90± 5.04	72.80± 4.85	72.20± 5.18
lenses	50.00±26.06	81.67±33.75 ○	75.00±32.63 ○	81.67±33.75 ○	66.67±38.49
pima	74.36± 4.83	73.83± 5.66	75.14± 3.68	75.27± 3.93	72.79± 3.78
tic-tac-toe	79.85± 1.84	85.07± 4.49 ○	97.81± 1.81 ○	94.47± 3.15 ○	78.70± 2.01 ●
wine	94.44± 5.86	93.86± 5.52	91.57± 9.27	93.27± 5.80	76.11±26.97

○, ● statistically significant improvement or degradation

Table 6. The accuracy of CBA compared to standard techniques. CBA performed error-based pruning and did not restrict the number of rules output by the association rule miner.

More specifically, we applied this methodology to the comparison of Apriori with predictive Apriori. Concerning the quality of the mined rule set, predictive Apriori is able to mine a high quality set of association rules. Its ranking metric—the expected predictive accuracy—makes sure that high-quality rules are ranked closer to the top. The rule ordering induced by the predictive accuracy outperforms Apriori's confidence-based ordering. The results also show that the rule set needed to build an accurate classifier is smaller when predictive Apriori is used. However, our experiments have shown that the time complexity of predictive Apriori is worse.

In addition, we have provided some benchmarks for the standard classifier in this research area—the CBA algorithm. Liu et al. [6] compare their algorithm to C4.5. We extended the comparison by including other state-of-the-art rule learners. In our opinion, CBA (in its standard combination with Apriori as class association rule mining algorithm) has comparable accuracy to standard techniques. However, CBA needs more rules and is slower than the standard techniques. We also found that CBA with predictive Apriori is slightly less accurate than CBA with Apriori. However, this is only

Dataset	CBA + Apriori	C4.5	JRip	PART	CBA + pred. Apriori
balance	0.28(0.23)	0.05 $\circ$	0.48	0.09 $\circ$	2.42(0.24) $\bullet$
breast-w	28.90(27.06)	0.01 $\circ$	0.31 $\circ$	0.02 $\circ$	283.56(1.40) $\bullet$
ecoli	0.85(0.62)	0.02 $\circ$	0.10 $\circ$	0.04 $\circ$	8.66(0.31) $\bullet$
glass	5.82(2.93)	0.03 $\circ$	0.10 $\circ$	0.05 $\circ$	46.81(0.34) $\bullet$
heart-h	29.28(14.04)	0.02 $\circ$	0.04 $\circ$	0.04 $\circ$	297.26(1.02) $\bullet$
iris	0.03(0.02)	0.00 $\circ$	0.01 $\circ$	0.00 $\circ$	0.78(0.03) $\bullet$
labor	344.49(198.14)	0.00 $\circ$	0.00 $\circ$	0.00 $\circ$	370.19(0.06) $\bullet$
led7	5.23(4.21)	0.03 $\circ$	0.22 $\circ$	0.05 $\circ$	50.05(5.37) $\bullet$
lenses	0.02(0.02)	0.00 $\circ$	0.00 $\circ$	0.00 $\circ$	0.26(0.00) $\bullet$
pima	6.54(4.90)	0.05 $\circ$	0.19 $\circ$	0.06 $\circ$	50.16(1.59) $\bullet$
tic-tac-toe	23.89(15.60)	0.02 $\circ$	0.44 $\circ$	0.06 $\circ$	2342.79(17.31) $\bullet$
wine	170.23(48.30)	0.01 $\circ$	0.03 $\circ$	0.01 $\circ$	2908.09(0.21) $\bullet$

$\circ$, $\bullet$ statistically significant improvement or degradation

Table 7. The time complexity of the three standard machine learning techniques and CBA during training. CBA performed error-based pruning and did not restrict the number of rules. The times in brackets are the times for CBA's pruning method including error-based pruning.

true if a large number of association rules is generated. Predictive Apriori can improve classification using association rules when it is used to generate a small set of rules.

References

1. Agrawal R. and Srikant R. Fast Algorithms for Mining Association Rules. In *Proc. of the 20th Int. Conf. on Very Large Data Bases*, pages 475–486. Morgan Kaufmann, 1994.
2. Blake C. and Merz C. UCI Repository of machine learning databases. http://www.ics.uci.edu/∼mlearn/MLRepository.html, 1998.
3. Cohen W. Fast Effective Rule Induction. In *Machine Learning: Proc. of the 12th Int. Conf.*, pages 115–123. Morgan Kaufmann, 1998.
4. Fayyad U. and Irani K. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proc. of the 13th Int. Joint Conf. on Artificial Intelligence*, pages 1022–1027. Morgan Kaufmann, 1993.
5. Frank E. and Witten I. Generating Accurate Rule Sets Without Global Optimization. In *Machine Learning: Proc. of the 15th Int. Conf.*, pages 152–160. Morgan Kaufmann, 1998.
6. Liu B., Hsu W. and Ma Y. Integrating Classification and Association Rule Mining. In *Proc. of the 4th Int. Conf. on Knowledge Discovery and Data Mining*, pages 80–86. The AAAI Press, 1998.
7. Nadeau C. and Bengio Y. Inference for the generalization error. *Advances in Neural Information Processing Systems*, 12:307–313, 1999.
8. Quinlan J. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.
9. Salzberg S. On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach. *Data Mining and Knowledge Discovery*, 1(3):317–327, 1997.
10. Scheffer T. Finding Association Rules That Trade Support Optimally against Confidence. Unpublished manuscript.
11. Scheffer T. Finding Association Rules That Trade Support Optimally against Confidence. In *Proc. of the 5th European Conf. on Principles and Practice of Knowledge Discovery in Databases*, pages 424–435. Springer-Verlag, 2001.
12. Witten I. and Frank E. *Data Mining: Practical machine learning tools and techniques with Java implementations*. Morgan Kaufmann, 2000.