Construct robust rule sets for classification

Jiuyong Li
Department of Mathematics and Computing
The University of Southern Queensland
Australia, 4350
jiuyong@usq.edu.au

Rodney Topor
School of Computing and Information Technology
Griffith University
Australia, 4111
rwt@cit.gu.edu.au

Hong Shen
Graduate School of Information Science
Japan Advanced Institute of Science and Technology
Japan, 923-1292
shen@jaist.ac.jp

ABSTRACT

We study the problem of computing classification rule sets from relational databases so that accurate predictions can be made on test data with missing attribute values. Traditional classifiers perform badly when test data are not as complete as the training data because they tailor a training database too much. We introduce the concept of one rule set being more robust than another, that is, able to make more accurate predictions on test data with missing attribute values. We show that the optimal class association rule set is as robust as the complete class association rule set. We then introduce the *k-optimal* rule set, which provides predictions exactly the same as the optimal class association rule set on test data with up to k missing attribute values. This leads to a hierarchy of k-optimal rule sets in which decreasing size corresponds to decreasing robustness, and they all more robust than a traditional classification rule set. We introduce two methods to find k-optimal rule sets, i.e. an optimal association rule mining approach and a heuristic approximate approach. We show experimentally that a k-optimal rule set generated by the optimal association rule mining approach performs better than that by the heuristic approximate approach and both rule sets perform significantly better than a typical classification rule set (C4.5Rules) on incomplete test data.

Keywords

Data mining, association rule, classification rule.

1. INTRODUCTION

1.1 Motivation

Automatic classification has been a goal for machine learning and data mining, and rule based methods are widely accepted due to their understandability and explanatory. Rule based classification usually involves two stages, learning and testing. Consider a relational database where each record is assigned a category (class), called a training database. In the learning stage, we generate a rule set where each rule associates a pattern with a class. Then in the test stage, we apply this rule set to test data without class information, and to predict the class that a record in the test database belongs to. If the predictive class is the class that the record supposed to belong to, then the prediction is correct. Otherwise, it is wrong. The proportional of correct predictions from test data is accuracy and surely a high accuracy is preferred.

In the machine learning community, many classification rule systems have been proposed, and they produce satisfactory accuracy in many applications. However, when the test data is not as complete as the training data, a classification rule set may perform poorly because it tailors the training data too much. We will give the following example to show this.

EXAMPLE 1. *Given a well-known data set listed in Table 1, a decision tree (e.g. ID3 [18] can be constructed as in Figure 1.*

NO.	Outlook	Temperature	Humidity	Wind	Play
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Table 1: A training data set

The following 5 rules are from the decision tree.

1. *If outlook is sunny and humidity is high, then do not play tennis.*
2. *If outlook is sunny and humidity is normal, then play tennis.*
3. *If outlook is overcast, then play tennis.*
4. *If outlook is rain and wind is strong, then do not play*

SIGKDD '02 Edmonton, Alberta, Canada

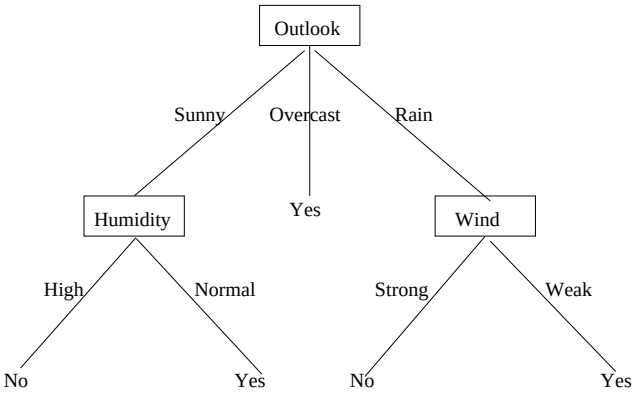

Figure 1: A decision tree from the training data set

tennis.

5. *If outlook is rain and wind is weak, then play tennis.*

We note that all rules include the attribute outlook. Suppose that we have a test data set in which outlook information is unknown. Then these rules cannot make any predictions. Hence, this rule set is not robust at all. However, we may have another rule set that can make some predictions in the presence of missing i.e. outlook information in test data.

In real world applications, missing data in a database is very common, especially in a test database. For example, we may generate a diagnostic rule set from records with the complete check results. But when we apply the rule set, some records may miss one or more check results for some reasons. Hence, a rule set that can make reasonably accurate predictions in the presence of missing attribute values in test data is highly desirable in practice. We say that a rule set is more robust than another rule set if it can make more accurate predictions on incomplete test data than the other rule set. In the following, we will explore this problem and its solutions.

1.2 Related work

Classification rule mining algorithms have been mainly developed for category prediction by the machine learning community. They use heuristic methods to find simple rule sets to explain training data well, and are generally categorized into two groups [17], simultaneously covering algorithms namely C4.5 [18], and sequential covering algorithms such as AQ15 [15] and CN2 [6]. Most algorithms generate simple and accurate rule sets that cover all training data, but these rule sets may not be robust in the presence of missing values as we will discuss in this paper.

There are many proposals for improving predictive accuracy of traditional classifiers, among which Bagging [5] and Boosting [8, 9] are significant in reducing predictive errors. Both techniques utilize voting (weights are involved in Boosting) from a set of classifiers obtained by sampling the training database. However, Bagging and Boosting make predictions hard to understand by users. In this paper, we also consider multiple rule sets, but we disturb a database systematically and use the union of all rule sets.

The proposal for association rule mining first appeared in [1]. Most research work has been on how to generate frequent itemsets efficiently since this may be the bottleneck of association rule mining. Apriori [2] is a widely accepted algorithm.

The traditional goal of association rule mining is to solve market basket problems. However, it can also be used to solve the classification problems, for example, CBA [12], CMAR [11], CAEP [7] and LB [14]. However, all previous proposals are on building accurate classifiers, and none of them relates to robust prediction.

In mining association rule based classification rules, some techniques have been previously developed. Using multiple supports can restrict many uninteresting rules whose consequences are frequently occurring in a database [13]. We proposed the optimal class association rule set [10] as a shortcut to generate association based classification rule sets efficiently.

There is some previous work on handling missing attribute values in training data [16, 18], but to the best of our knowledge there is no report on how to deal with the missing attribute values on test data.

2. ROBUSTNESS OF THE OPTIMAL CLASS ASSOCIATION RULE SET

2.1 The class association rule set

Given a relational database D with n attributes, a record of D is a n-tuple. For convenience of description, we consider a record as a set of attribute-value pairs, denoted by T. A *pattern* is a set of attribute-value pairs. The *support* of a pattern P is the ratio of the number of records containing P to the number of records in the database, denoted by $sup(P)$. An *implication* is a formula $P \Rightarrow c$, where P is a pattern and c is a class. The support of the implication $P \Rightarrow c$ is $sup(P \cup c)$. The confidence of the implication is $sup(P \cup c)/sup(P)$, denoted by $conf(P \Rightarrow c)$. The *covered set* of the rule is the set of all records containing the antecedent of the rule, denoted by $cov(P \Rightarrow c)$. We say $A \Rightarrow c$ is a *class association rule* if $sup(A \Rightarrow c) \geq \sigma$ and $conf(A \Rightarrow c) \geq \phi$, where σ and ϕ are user specified minimum support and confidence respectively. A class association rule set is a set of association rule set with classes as their consequences.

DEFINITION 1. *The complete class association rule set is the set of all class association rules wrt a database, the minimum support and the minimum confidence.*

Given a database D, the minimum support σ and the minimum confidence ϕ, the complete (class association) rule set[1] is denoted by $R_c(\sigma, \phi)$, or simply R_c.

We notice that the goal of classification rule generation is for prediction, so confidence, a training accuracy, does not suit for this goal. It is necessary to have a statistical true accuracy as a replacement.

By using a result from [17], we adopt the lower bound of test (true) accuracy of a hypothesis as the accuracy of a the hypothesis. Hence, we define the *accuracy* of a rule to be

$$acc(A \Rightarrow c) = conf(A \Rightarrow c) - z_N \sqrt{\frac{conf(A \Rightarrow c)(1 - conf(A \Rightarrow c))}{|cov(A \Rightarrow c)|}}$$

where z_N is a constant related with a statistical confidence interval, for example, $z_N = 1.96$ when the confidence interval is 95%.

[1]In the rest of this paper, we consistently discuss class association rules, so we omit words "class association" afterwards.

The requirement of using this accuracy is that the support of a rule is not too small, for example, the absolute support number is not less than 30. If the support of a rule is too small, we need another estimation of test accuracy on very small sample data. Laplace accuracy can then be used instead [3]. It is $acc(A \Rightarrow c) = \frac{|cov(Ac)|+1}{|cov(A)|+|C|}$ where $|C|$ is the number of all classes.

Usually, the minimum confidence requirement of a class association rule is very high so it is natural to exclude conflicting rules, such as $A \Rightarrow c_1$ and $A \Rightarrow c_2$, in a complete rule set.

2.2 The optimal class association rule set

In the practice of rule set based classification, a set of rules is usually sorted by decreasing accuracy, and tailed by a default prediction. This ordered rule set is called a rule based classifier. In classifying an unseen test record (an input record without class attribute information), the first rule that matches the case classifies it. If no rule matches the record, the default prediction is used. In this paper, we ignore the effect of the default prediction since we will concentrate on the predictive power of a rule set. We formulise this procedure in the following.

For a rule r, we use $cond(r)$ to represent its antecedent (conditions), and $cons(r)$ to denote its consequence. Given a test record T, we say rule r covers T if $cond(r) \subseteq T$. A rule can make a prediction on its covered record, denoted by $r(T) \rightarrow cons(r)$. If $cons(r)$ is the class of T, then the rule makes a correct prediction. Otherwise, it makes a wrong prediction. We let the accuracy of a prediction equals the accuracy of the rule making the prediction, denoted by $acc(r(T) \rightarrow c)$. If a rule gives the correct prediction on a record, then we say the rule *identifies* the record.

DEFINITION 2. *Let T be a record in database D and R a rule set for D. A rule r in R is predictive for T wrt R if r covers T. If two rules cover T we choose the one with the greater accuracy. If two rules have the same accuracy we choose the one with higher support. If two rules have the same support we choose the one with the shorter antecedent.*

Please note that in the above definition, we take the support and the length of antecedent of a rule into consideration. This is because they have been minor criteria for sorting rules in a rule based classifier in previous practice, such as in [12]. It is easy to understand the preference of the highest support rule among a number of rules with the same accuracy. The preference for a short rule is consistent with the preference for a simple rule in traditional classification rule generation practice.

As both accuracy and support are real numbers, in a large database it is very unlikely that a record supports two rules with the same accuracy and support. Therefore, we suppose that each record has a unique predictive rule for a given database and rule set in the rest of paper.

The *prediction of rule set* R on record T is the same as that of the predictive rule of R on T with the same accuracy.

Now we consider how to compare predictive power of rules. We use $r_2 \subset r_1$ to represent $cond(r_2) \subset cond(r_1)$ and $cons(r_2) = cons(r_1)$. We call r_2 is more general than r_1, or r_1 is more specific than r_2.

DEFINITION 3. *Given two rules r_1 and r_2, we say that r_2 is stronger than r_1 iff $r_2 \subset r_1 \wedge acc(r_2) \geq acc(r_1)$. We denote rule r_2 is stronger than rule r_1 by $r_2 > r_1$. In a complete rule set R_c, we say a rule in R is (maximally) strong. if there is no other rule in R that is stronger than it. Otherwise, the rule is weak.*

It is clear that only a (maximally) strong rule can make a prediction in the complete rule set. Thus, we have the following definition and an immediate result.

DEFINITION 4. *We call the set of all (maximally) strong rules the optimal rule set wrt the complete class association rule set.*

LEMMA 1. *The optimal rule set wrt R_c is the set all potentially predictive rules in R_c.*

2.3 Robustness of the optimal class association rule set

Suppose that we have an incomplete test record, i.e. some attribute-values are missing. It is clear that we prefer a rule set that can make correct prediction on these incomplete records. More formally, we will give a definition for the robustness as the following. Note that we say that a rule set gives any prediction on a record with accuracy of zero when it cannot provide a prediction on the record.

DEFINITION 5. *Let D be a database, T be a record of D, and R_1 and R_2 be two rule sets for D. Rule set R_1 is more robust than R_2 if, for all $T' \subseteq T$, predictions made by R_1 are at least as accurate as those by R_2.*

Suppose that R_1 is more robust than R_2. For test data that are as complete as the training data both rule sets give the same number of correct predictions with the same accuracy. For test data that are not as complete as the training data, rule set R_1 can provide at least the same number of correct predictions as rule set R_2 with at least the same accuracy. Hence, a robust rule set has more predictive power when test data are not as complete as the training data.

Naturally, more rules will enhance the robustness of a rule set, and the complete rule set is the most robust rule set. However, this rule set is usually too large and includes many rules without predictive power. Hence, we can go further to simplify it.

Clearly there are natural connections between strong rules and predictive rules since a strong rule is a potentially predictive rule. Hence, we have,

THEOREM 1. *For every rule set $R \subseteq R_c$ for database D, the optimal class association rule set R_o is the smallest rule set that is as robust as the complete class association rule set.*

This means that no matter what an input record is (complete or incomplete), that the optimal rule set gives exact the same prediction on the record at the same accuracy as the complete rule set.

Though the optimal rule set is much smaller than the complete rule set, it is still much larger than a traditional classification rule set. Some rules in the optimal rule set may be unnecessary when the number of missing attribute values is limited. Hence, we may further simplify the optimal rule set. Besides, we are interested in the relationships between the optimal rule set and a traditional classification rule set. These are goals in the next section.

3. ROBUSTNESS OF K-OPTIMAL CLASS ASSOCIATION RULE SETS

In this section, we have a default rule set, namely the complete rule set, from the training database. When we say a predictive rule without mentioning a rule set, then it is with respect to the complete rule set. The test database is the same as the training database without class information.

Robustness mainly concerns missing attribute values in test databases, and hence we first define a k-incomplete database to be a new database with exactly k missing values from every record of the test database.

DEFINITION 6. *Let D be the test database and $k \geq 0$. The k-incomplete database $D_k = \{ T' \mid T' \subset T, T \in D, |T| - |T'| = k\}$.*

For convenience of discussion, we consider all k-incomplete databases of D as a set of $\binom{n}{k}$ (n is the number of attributes for D) databases in which each omit exactly k attribute (column) information from D. For example, all 1-incomplete databases contains a set of n databases where each omits one attribute (column) information from D. We note that the 0-incomplete database of D is D itself.

Let us represent the optimal rule set in terms of incomplete databases.

LEMMA 2. *R_o is the set of predictive rules for records in k-incomplete databases wrt R_c where $0 \leq k \leq n$.*

The optimal rule set preserves all potentially predictive rules from a training database for all incomplete databases. Now we consider how to preserve all potentially predictive rules for some incomplete test databases.

DEFINITION 7. *The k-optimal rule set ($k \geq 0$) over a database is all predictive rules on all k-incomplete databases.*

We have the following result.

LEMMA 3. *The k-optimal rule set provides the same predictions as the optimal rule set on all p-incomplete databases for $0 \leq p \leq k$.*

We can understand a k-optimal rule set in the following way. A k-optimal rule set is a subset of the optimal rule set that makes prediction as well as the optimal rule set on a test database with k missing attribute value per record. As a special case, 0-optimal rule set makes predictions as well as the optimal rule set on a complete test database.

THEOREM 2. *The $(k+1)$-optimal rule set ($k \geq 0$) is at least as robust as the k-optimal rule set.*

Clearly, a k-optimal rule set is a subset of the optimal rule set.

The k-optimal rule sets form a hierarchy.

LEMMA 4. *Let R^k and $R^{(k+1)}$ be the k-optimal and the $(k+1)$-optimal rule sets for D and R_c. Then $R^k \subseteq R^{k+1}$.*

Till now, we have introduced the set of optimal rule sets, and we observe that the following chain always holds these optimal rule sets.

$$R_c \supseteq R_o \supseteq \cdots \supseteq R^{k+1} \supseteq R^k \supseteq \cdots \supseteq R^0$$

From this relation, we can see that the robustness of a k-optimal rule set for $k \geq 0$ is due to that it preserves more potentially predictive rules in case that some rules are paralysed by missing values in a test database.

Usually, a traditional classification rule set is smaller than a 0-complete rule set, since most post pruning algorithms of traditional classification systems work in a way to reduce the size of an output rule set. Because of the heuristic trait of traditional classification rule generation algorithms, we cannot characterize the exact relationship between a traditional classification rule set and a k-optimal rule set. From our observations, most rules in a traditional classification rule set are in the 0-optimal rule set. For example, the rule set from the decision tree on the tennis database is a subset of 0-optimal rule set. Generally, a traditional classification rule set is less robust than a 0-optimal rule set.

Finally, we will consider a property that will help us to find k-optimal rule sets. We can interpret the k-optimal rule set through a set of 0-optimal rule sets.

LEMMA 5. *The union of all 0-optimal rule sets over all k-incomplete databases is the k-optimal rule set.*

This lemma suggests that we can generate a k-optimal rule set by generating 0-optimal rule sets on a set of incomplete databases of the training database.

4. PRELIMINARY EXPERIMENTS

We implemented two algorithms to generate k-optimal robust rule sets. One is an approximate method extended from C4.5 [18], called multiple C4.5rules. The other is a precise method extended from the optimal class association rule set [10], called optimal rule set approach. For more details and software, please send an email to jiuyong@usq.edu.au.

We use four databases from UCI ML Repository [4] in our experiments and a brief summary of the databases is listed in Table 2. Our experiments were conducted on a Sun server with two 200 MHz UltraSPARC CPUs. In the experiment, we use *local support* of rule r, which is $sup(r)/sup(cons(r))$, to avoid too many rules in the large distributed classes and too few rules in the small distributed class . For example, in database Hypothyroid, 95.2% records belong to class Negative and only 4.8 % records belong to class Hypothyroid. So, 5% (global) support is very small for class Negative class, but is too large for class Hypothyroid.

Database	Size	Attr num	Values per attr	class num
Anneal	899	38	2-9	5
Congressional Voting	435	16	3	2
Hypothyroid	3164	25	2 - 4	2
Mushrooms	8124	22	2 - 12	2

Table 2: A brief description of databases

The experimental settings is listed in Table 3. Min Sup is the minimum local support, Min Acc is the minimum accuracy and Max Len is the length of antecedent for the longest rule in an optimal rule set. In database Hypothyroid we stopped executing the program before it found more longer rules.

Sizes and generation time of different rule sets are listed in Table 4. It is clear that the complete rule set is much larger

Database	Min Sup	Min Acc	Max Len
Anneal	0.05	0.95	7
Congressional Voting	0.1	0.95	9
Hypothyroid	0.1	0.95	4
Mushrooms	0.2	0.95	6

Table 3: The experimental setting

than the optimal rule set and more expensive to generate. The size of a k-optimal rule set is much smaller than that of the optimal rule set and is a little larger than that of a traditional classification rule set.

	Mushrooms		Voting	
Rule set	Size	T(sec)	Size	T(sec)
complete	49599	657	151374	1387
optimal	312	18	1133	6
1-optimal	67	18	127	6
0-optimal	39	18	57	6
multiple C4.5Rules ($k = 1$)	46	195	32	1
single C4.5Rules	16	9	7	<1
	Anneal		Hypothyroid	
Rule set	Size	T(sec)	Size	T(sec)
complete	87247	857	11878	112
optimal	219	2	146	44
1-optimal	70	2	56	44
0-optimal	44	2	32	44
multiple C4.5Rules ($k = 1$)	70	20	21	8
single C4.5Rules	22	<1	7	<1

Table 4: Size and generation time of different rule sets

In our experiments, all rule sets are tested without default prediction. This is because the default prediction may disguise the true accuracy. Consider database Hypothyroid: if we set the default prediction as Negative, then a classifier without any rule will give 95.2% accuracy. Clearly, this accuracy is misleading.

We evaluated the predictive power of a rule set by the identification accuracy, which is the accuracy without default prediction.

Identification accuracy = (the number of identified records) / (the number of all records in a database)

The identification accuracy is the proportion of identified records by the rule set in a database. The higher the accuracy, the better the predictive power. Its range is between 0 to 100%.

We tested all generated rule sets on l-incomplete test databases ($0 \leq l \leq 6$) of four databases, and reported their identification accuracy in Figure 2. In our experiment, the number of missing values is compared with the training data. When a training database already has missing values, then the missing values are additional. Each point in Figure 2 is the average of ten trials.

From Figure 2, we can see that when test data is incomplete, a rule set from single C4.5Rules performs poorly while both "precise" (the optimal rule set approach) and approx-

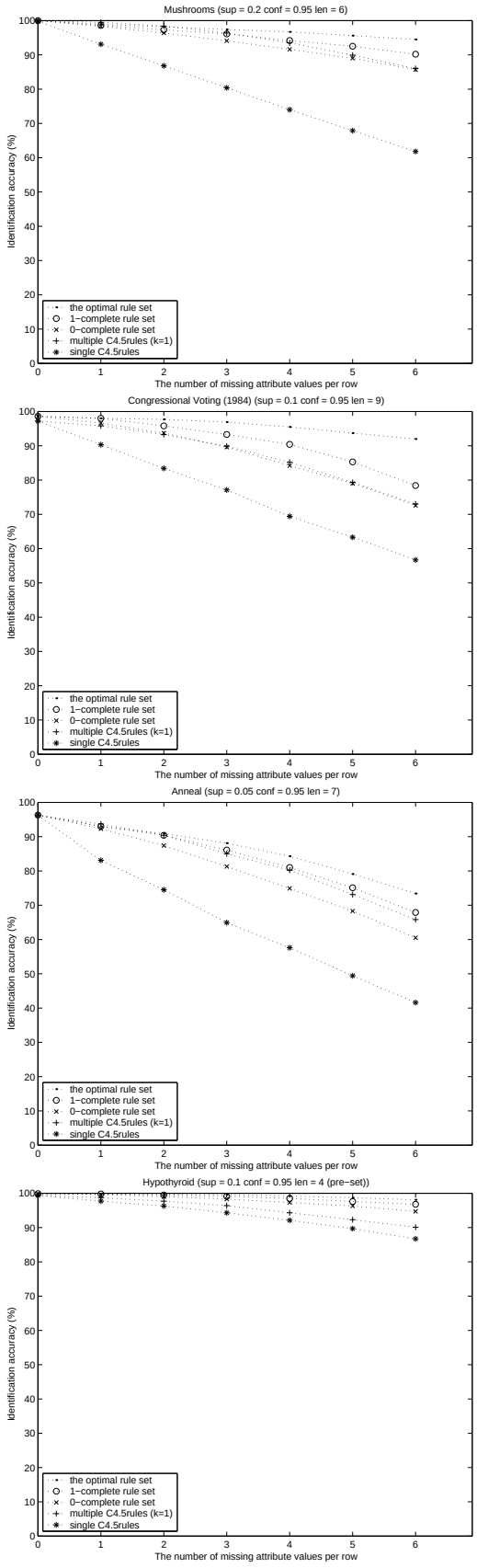

Figure 2: The robustness of different rule sets

imate (the multiple C4.5rules) k-optimal rule sets perform significantly better. In all cases, the optimal rule set performs best, and the 1-optimal rule set the second best. These results are consistent with Theorems 1 and 2. Rule sets from multiple C4.5rules perform better than those from single C4.5Rules but worse than 1-optimal rule sets. We also note that all "precise" 1-optimal rule sets perform exactly the same as the optimal rule sets when the number of missing values is not more than 1 per record as stated by Lemma 3. We also note that a "precise" 1-optimal rule set performs better on incomplete test databases than an approximate 1-optimal rule set from the multiple C4.5rules. Further, approximate k-optimal rule sets (from multiple C4.5rules) perform unstably: they sometimes perform better than 0-optimal rule set, but sometimes do not.

5. CONCLUSION

In this paper, we discussed a new problem, finding robust rule sets to predict on a test database that is not as complete as the training database. We defined a criterion to compare the robustness for different rule sets from a database. We revealed that the optimal rule set is as robust as the complete rule set with the smallest size, and defined k-optimal rule sets for test databases with limited missing attribute values to obtain simple rule sets. We characterized the relationships among k-optimal rule sets and a traditional classification rule set. We proposed a method to find k-optimal sets through the optimal association rule approach. We showed experimentally that a k-optimal rule set generated from the proposed algorithm performs better than a k-optimal rule set generated by an extension of C4.5Rules on incomplete test databases, and that both rule sets perform significantly better than a traditional classification rule set on incomplete test databases. Given the frequent missing value in real world databases, the k-optimal rule sets have significant potential in future applications.

6. REFERENCES

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining associations between sets of items in massive databases. In *Proc. of the ACM SIGMOD Int'l Conference on Management of Data*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the Twentieth International Conference on Very Large Databases*, pages 487–499, Santiago, Chile, 1994.

[3] E. A. Bender. *Mathematical Methods in Artificial Intelligence*. IEEE Computer Society Press, 1996.

[4] E. K. C. Blake and C. J. Merz. UCI repository of machine learning databases, http://www.ics.uci.edu/∼mlearn/MLRepository.html, 1998.

[5] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[6] P. Clark and R. Boswell. Rule induction with CN2: Some recent improvements. In *Machine Learning - EWSL-91*, pages 151–163, 1991.

[7] G. Dong, X. Zhang, L. Wong, and J. Li. CAEP: Classification by aggregating emerging patterns. In *Proceedings of the 2nd International Conference on Discovery Science (DS-99)*, volume 1721 of *LNAI*, pages 30–42, Berlin, 1999. Springer.

[8] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156, 1996.

[9] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[10] J. Li, H. Shen, and R. Topor. Mining optimal class association rule set. In *Proceedings of the 5th Pacific-Asia Conference on Methodologies for Knowledge Discovery and Data Mining (PAKDD 2001)*, pages 364–375. Springer, 2001.

[11] W. Li, J. Han, and J. Pei. CMAR: Accurate and efficient classification based on multiple class-association rules. In *Proceedings 2001 IEEE International Conference on Data Mining (ICDM 2001)*, pages 369–376. IEEE Computer Society Press, 2001.

[12] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*, pages 27–31, 1998.

[13] B. Liu, W. Hsu, and Y. Ma. Mining association rules with multiple minimum supports. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 337–341, N.Y., 1999. ACM Press.

[14] D. Meretakis and B. Wüthrich. Extending naive Bayes classifiers using long itemsets. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 165–174, N.Y., 1999. ACM Press.

[15] R. Michalski, I. Mozetic, J. Hong, and N. Lavrac. The AQ15 inductive learning system: an overview and experiments. In *Proceedings of IMAL 1986*, Orsay, 1986. Université de Paris-Sud.

[16] J. Mingers. An empirical comparison of selection measures for decision tree induction. *Machine Learning*, 3:319–342, 1989.

[17] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[18] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.