New results for a Hybrid Decision Tree/Genetic Algorithm for Data Mining

Deborah R. Carvalho
Computer Science Dept.
Universidade Tuiuti do Parana
Av. Comendador Franco, 1860
Curitiba PR. 80215-090. Brazil
Deborah@utp.com.br

Alex A. Freitas
Computing Laboratory
University of Kent at Canterbury
Canterbury, Kent, CT2 7NF, UK
A.A.Freitas@ukc.ac.uk
http://www.cs.ukc.ac.uk/people/staff/aaf

Keywords*: data mining, classification, genetic algorithm, rule discovery, decision trees.*

1. Introduction

This paper addresses the well-known data mining task of discovering classification rules [5]. A classification rule is a prediction rule of the form: IF <conditions> THEN <prediction (class)>. An example of a classification rule is: "IF (*Age* > 25) AND (*Salary* > $50,000) THEN (*Credit* = good)". Classification rules in this format have the advantage of being intuitively comprehensible for the user, which is important from a data mining viewpoint. Classification rules are normally expressed in disjunctive normal form, where each rule represents a disjunct and each rule condition represents a conjunct – i.e., the conditions in a rule antecedent are connected by a logical conjunction (AND) operator. In this context, a small disjunct can be defined as a rule which covers a small number of training examples (records, or cases) [6].

At first glance small disjuncts seem to have a small impact on classification accuracy. However, it is important to note that, although each small disjunct covers a small number of examples, the set of all small disjuncts can cover a large number of examples [10].

We have recently proposed a hybrid decision tree/genetic algorithm (GA) method to cope with the problem of small disjuncts [1]. The main contribution of this paper is that it extends the computational results presented in that work along two important directions, namely:

(a) In [1] we reported results for only 8 data sets, whereas in this paper we report computational results for 22 data sets, so that the evaluation of our method is more robust;
(b) In [1] we reported results concerning only the predictive accuracy of the hybrid method, whereas in this paper we report results concerning not only the predictive accuracy, but also the simplicity (comprehensibility) of the discovered rules.

2. The Hybrid Decision-Tree/Genetic Algorithm (GA)

Our hybrid decision tree/GA method discovers classification rules in two training phases. First, it runs C4.5, a well-known decision-tree induction algorithm [8]. The induced, pruned tree is transformed into a set of rules (or disjuncts). Each of these rules is considered either as a "small" or a "large" disjunct, depending on whether or not the number of examples covered by the rule is smaller than or equal to a given threshold. Second, it uses a GA to discover rules covering the examples belonging to small disjuncts. Examples belonging to large disjuncts are classified by the decision tree produced by C4.5.

The basic idea of the method can be justified as follows. Decision-tree algorithms have a bias towards generality that is well suited for large disjuncts, but not for small disjuncts. On the other hand, GAs tend to cope with attribute interaction better than most greedy rule induction algorithms [4]. This makes them a promising solution for the problem of small disjuncts, since attribute interactions are believed to be one of the causes of small disjuncts [3].

The GA component of the method was specifically designed for discovering small disjunct rules. The first step consists in creating a suitable training set for this GA, as follows. All the examples belonging to all the leaf nodes considered small disjuncts are grouped into a single training set, called the "second training set" (to distinguish it from the original training set used to build the decision tree). This second training set is provided as input data for the GA. The basic idea of this process is illustrated in Figure 1, where the leaf nodes considered small disjuncts are denoted by squares.

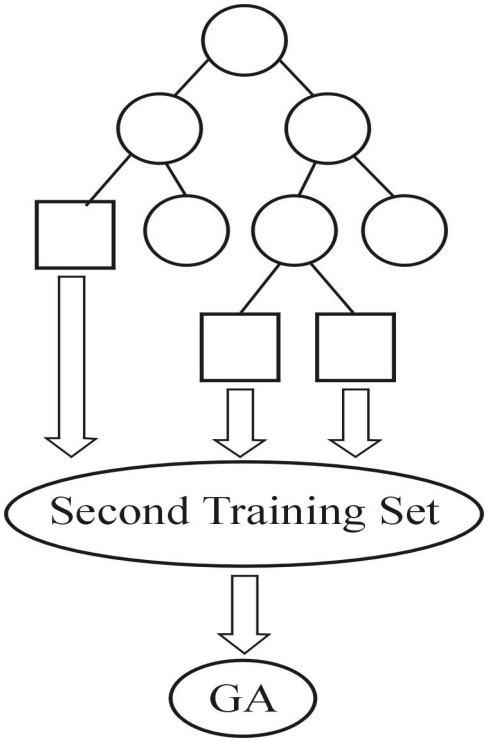

Figure 1: Construction of the second training set to be given as input to the GA

Each individual (candidate solution) represents the antecedent (IF part) of a small-disjunct rule. The antecedent of a rule consists of a conjunction of conditions, where each condition is an attribute-value pair [1]. Note that the consequent (THEN part) of each rule is not represented in the genome of individuals. Rather, the consequent of each rule is dynamically chosen as the most frequent class in the set of examples covered by that rule's antecedent.

In addition to standard crossover and mutation operators, the GA uses a new task-dependent, heuristic rule-pruning operator specifically designed for pruning classification rules. Each condition in the genome is associated with a flag, called the active bit, which takes the value 1 or 0 to indicate whether or not, respectively, the associated condition occurs in the decoded rule antecedent. This allows the GA to use a fixed-length genome (for the sake simplicity) to represent a variable-length antecedent. The heuristic for rule pruning is based on the idea of using the decision tree built by C4.5 to compute a classification accuracy rate for each attribute, according to how accurate were the classifications performed by the decision tree paths in which that attribute occurs. That is, the more accurate were the classifications performed by the decision tree paths in which a given attribute occurs, the higher the accuracy rate associated with that attribute, and the smaller the probability of removing a condition with that attribute form a rule. (See [1] for more details.)

In order to discover a diverse set of rules the GA uses an iterative niching method which is similar to the sequential covering approach used by some rule induction algorithms. The basic idea is that the GA is iteratively applied to the second training set in such a way that each iteration (i.e., each GA run) discovers a single rule, and each iteration discovers a rule covering examples which are different from examples covered by rules discovered in previous iterations. In the first iteration the GA has access to

all the examples in the second training set, and it discovers a single rule. Then the examples correctly covered by this rule are removed from the second training set. An example is "correctly covered" by a rule if the example's attribute values satisfy all the conditions in the rule antecedent and the example belongs to the same class as predicted by the rule. This process is iteratively performed while the cardinality of (number of examples in) the second training set is greater than 5. (It is assumed that a cardinality smaller than 5 means that there are too few examples to allow the discovery of a reliable classification rule.)

The fitness function is given by the formula: Fitness = (TP / (TP + FN)) * (TN / (FP + TN)), where TP, FN, TN and FP stand for the number of true positives, false negatives, true negatives and false positives [5].

3. Computational Results

We have evaluated the performance of our hybrid decision tree/GA method across 22 data sets, out of which 12 are public domain data sets (available from the University of California at Irvine (UCI)'s data repository at: http://www.ics.uci.edu/~mlearn/MLRepository.html). The other 10 are proprietary data sets contain data about the scientific production of researchers of the Brazilian government's National Council of Scientific and Technological Development (CNPq) [9]. These 10 data sets will be simply denoted by DS-1 through DS-10 in the table containing the computational results, to be shown later. These 22 data sets represent a wide range of different classification problems, with considerable variation in the number of examples and attributes across the data sets. All examples with missing values were removed from the data sets as a preprocessing step.

As mentioned in the Introduction, although each small disjunct covers a small number of examples, it is possible that the set of all small disjuncts covers a large number of examples. Indeed, this was confirmed in several data sets used in our experiments, as can be seen in Figure 2. This figure shows the percentage of training examples belonging to small disjuncts (out of the total set of examples) for each data set. Note that the percentage of small-disjunct examples was greater than 10% in about half of the data sets and it was greater than 40% in Wave data set.

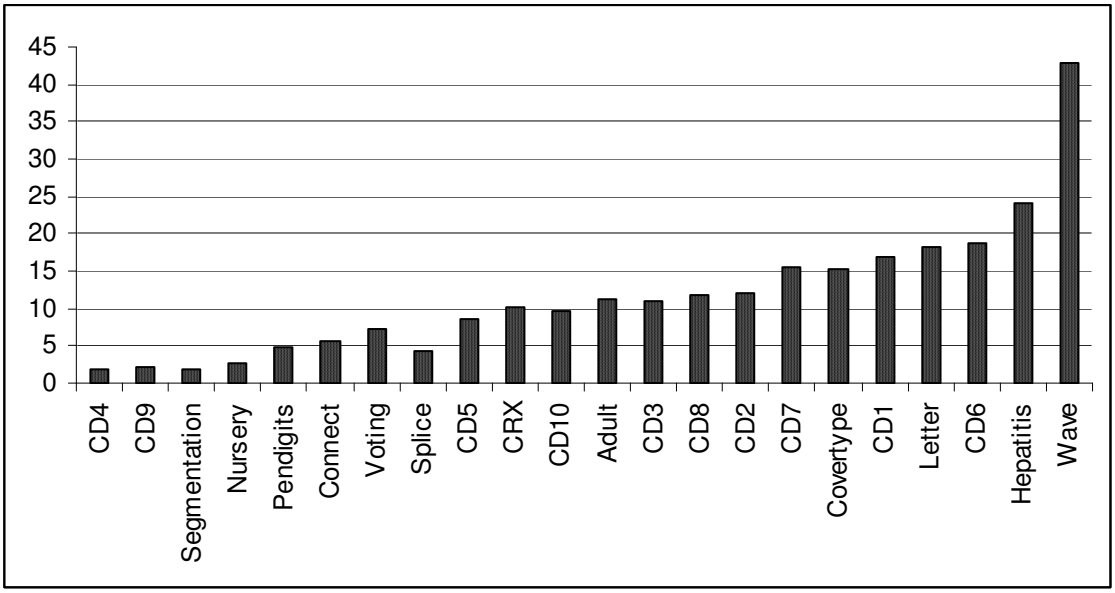

Figure 2: Relative frequency of small-disjunct examples found in each data set

The experiments used C4.5 [8] as the decision-tree component of our hybrid method. The performance of the hybrid decision tree/GA method was compared with the performance of two versions of C4.5 alone, as follows. The first version consists of running C4.5 with its standard set of parameters, using

the tree constructed by C4.5 to classify all test examples – both large-disjunct and small-disjunct examples. The second version is a variant of C4.5, which we call double C4.5, specifically designed for coping with small disjuncts. This variant consists of two runs of C4.5. The first run considers all examples in the original training set, producing a first decision-tree. Once all the examples belonging to small disjuncts have been identified by this decision tree, C4.5 is run again on the "second training set" (the same training set used as input to the GA), producing a second decision tree. A test example is classified as follows. If it belongs to a large disjunct of the first decision tree (i.e., the tree built using the entire training set), it is classified by the corresponding leaf node of that tree. Otherwise – i.e., it belongs to a small disjunct of the first tree – it is classified by the second decision tree (i.e., the tree built using the second training set). Note that this makes the comparison between our hybrid method and double C4.5 fair, since: (a) both the hybrid C4.5/GA and double C4.5 use the same second training set to discover rules belonging to small disjuncts; (b) both methods use the tree built by standard C4.5 (using the entire training set) to classify test examples belonging to large disjuncts; and (c) both methods use the rules or tree built from the second training set to classify test examples belonging to small disjuncts.

In our experiments we have used a commonplace definition of small disjuncts, based on a fixed threshold of the number of examples covered by the disjunct. The general definition is: "A decision-tree leaf if considered a small disjunct if and only if the number of examples belonging to that leaf is smaller than or equal to a fixed size S". We did experiments with two values of S, namely S = 10 and S = 15. Due to limitation of space, we report only the results for S = 10 here. The results for S = 15 are qualitatively similar.

In three data sets – Adult, Connect and Letter - we used a single partitioning of the data into a training and test sets, since these data sets are relatively large. In the other data sets, since the number of examples was not so large, we used a well-known 10-fold stratified cross-validation procedure [5], in order to make the results more reliable from a statistical viewpoint. In the case of the Adult, Connect and Letter data sets, all the predictive accuracy results reported here are the results on the test set (which was not seen during training). In the case of the other data sets, all the predictive accuracy results are the average results, on the test set, over the ten iterations of the cross-validation procedure. In the case of the hybrid C4.5/GA, the GA was executed ten times, with a different random seed each time. Hence, in the Adult, Connect and Letter data sets the results of C4.5/GA represent an average over the 10 executions of the GA. In the other data sets the results of C4.5/GA represents an average over 100 executions (10 iterations of cross validation times 10 different random seeds). The GA parameters were set as follows: population size: 200 individuals, number of generations: 50, one-point crossover probability: 80%, mutation probability: 1%, tournament selection with tournament size of 2. We made no attempt to optimize these parameters, in order to make a fair comparison with C4.5 and double C4.5, since we are using standard, non-optimized parameters for C4.5 as well.

Table 1 reports the accuracy rate of standard C4.5, double C4.5 and our hybrid C4.5/GA in the 22 data sets. For each data set, the highest accuracy rate among the three methods is shown in bold. The numbers after the "±" symbol denote standard deviations. In the columns referring to double C4.5 and C4.5/GA, the cells where the corresponding method achieved a significantly larger (smaller) accuracy rate than C.45 (considered a baseline method) are indicated by the symbol "+" ("–") after the standard deviations. The accuracy rate of double C4.5 or C4.5/GA is considered significantly larger or smaller than the accuracy rate of standard C4.5 if the corresponding accuracy rate intervals, taking into account the standard deviations, do not overlap. The results are summarized in the last two rows of the table. More precisely, the last but one row shows the number of data sets were double C4.5 and C4.5/GA obtained an accuracy rate significantly better than the baseline standard C4.5, whereas the last row shows the number of data sets were double C4.5 and C4.5/GA obtained an accuracy rate significantly worse than standard C4.5.

As can be seen in the table, the hybrid C4.5/GA outperforms the two other algorithms in 14 of the 22 data sets – i.e., in 63% of the cases. Double C4.5 outperforms the other two algorithms in only 3 data sets. Note that not all results are statistically significant. The hybrid C4.5/GA is significantly better than C4.5 in 9 data sets and the reverse is true in only 2 data sets, which indicates that the hybrid

method is a good solution for the problem of small disjuncts. Double C4.5 is significantly better than C4.5 in 4 data sets and reverse is true in 4 data sets as well, which indicates that double C4.5 does not seem to be an improvement over standard C4.5, with respect to predictive accuracy. Overall, taking into account the 22 data sets, the results indicate that the hybrid C4.5/GA achieved considerably better results than both standard C4.5 and double C4.5, with respect to predictive accuracy.

TABLE 1: ACCURACY RATE (%) ON THE TEST SET

Data Set	C4.5	Double C4.5	C4.5/AG
Connect	72,60 ± 0,5	76,19 ± 0,6 **+**	**76,95 ± 0,1 +**
Adult	78,62 ± 0,5	76,06 ± 0,5 **–**	**80,04 ± 0,1 +**
Crx	**91,79 ± 2,1**	90,78 ± 1,2	91,66 ± 1,8
Hepatitis	80,78± 13,3	82,36 ± 18,7	**95,05 ± 7,2**
House	93,62 ± 3,2	89,16 ± 8,0	**97,65 ± 2,0**
Segmentat.	**96,86 ± 1,1**	72,93 ± 5,5 **–**	78,68 ± 1,1 **–**
Wave	75,78 ± 1,9	64,93 ± 3,9 **–**	**83,95 ± 3,0 +**
Splice	65,68 ± 1,3	61,51 ± 6,6	**70,70 ± 6,3**
Covertype	**71,61 ± 1,9**	68,64 ± 14,8	68,71 ± 1,3
Letter	**86,40 ± 1,1**	82,77 ± 1,0 **–**	79,24 ± 0,2 **–**
Nursery	95,40 ± 1,2	**97,23 ± 1,0**	96,77 ± 0,7
Pendigits	96,39 ± 0,2	**96,86 ± 0,4**	95,72 ± 0,9
DS-1	60,71 ± 3,0	**63,82 ± 5,2**	63,43 ± 1,4
DS-2	65,55 ± 1,5	72,52 ± 5,9	**73,77 ± 2,5 +**
DS-3	75,65 ± 2,4	82,27 ± 1,3 **+**	**84,15 ± 0,9 +**
DS-4	**92,97 ± 0,9**	92,58 ± 1,0	92,72 ± 1,0
DS-5	82,7 ± 2,8	83,01 ± 1,9	**83,36 ± 2,1**
DS-6	57,78 ± 2,1	60,68 ± 3,2	**61,69 ± 1,6 +**
DS-7	65,18 ± 1,0	70,29 ± 2,4 **+**	**71,27 ± 1,6 +**
DS-8	75,57 ± 1,4	81,03 ± 1,9 **+**	**82,63 ± 1,9 +**
DS-9	93,00 ± 0,5	93,72 ± 1,2	**93,80 ± 1,4**
DS-10	82,80 ± 1,7	85,60 ± 1,4	**86,88 ± 1,6 +**
Number of significantly better results		4	9
Number of significantly Worse results		4	2

With respect to simplicity (number of discovered rules and number of conditions per rule), in almost all the 22 data sets the hybrid C4.5/GA discovered a rule set considerably simpler (smaller) than the rule set discovered by standard C4.5. (Each path from the root to a leaf node of the decision tree is considered a rule.) More precisely, the number of rules discovered by C4.5/GA was significantly smaller (taking into account the standard deviations) than the number of rules discovered by standard C4.5 in 21 of the 22 data sets, i.e., in approximately 95% of the cases. The number of rules discovered by double C4.5 was significantly smaller than the number of rules discovered by standard C4.5 in 18 of the 22 data sets, but the former was significantly larger than the latter in 2 data sets.

With respect to the average number of conditions per discovered rule, once again the hybrid C4.5/GA obtained the best results. The number of conditions per rule discovered by C4.5/GA was significantly smaller (taking into account the standard deviations) than the number of conditions per rule discovered by standard C4.5 in 16 of the 22 data sets, i.e., in 72% of the cases. The former was significantly larger than the latter in only one data set (Hepatitis). The number of conditions per rule discovered by double C4.5 was significantly smaller than the number of conditions per rule discovered by standard C4.5 in 8 of the 22 data sets. The former was significantly larger than the latter in only one data set (Hepatitis).

4. Conclusions and Future Work

The computational results reported in the previous section constitute significant evidence that, with respect to predictive accuracy, the hybrid C4.5/GA can be considered a good solution for the problem of small disjuncts. Overall, it obtained results considerably better than both standard C4.5 and double C4.5. Another advantage of the hybrid C4.5/GA is that in general it discovers a rule set considerably simpler (smaller) than the rule set discovered by standard C4.5. Hence, C4.5/GA seems a good choice for coping with the problem of small disjuncts, particularly when one wants to maximize both the predictive accuracy and the simplicity of the discovered rule set.

One possible research direction consists of performing a "meta-learning" [7] on the computational results obtained in our experiments. That is, one could use a classification algorithm to discover rules predicting, for each data set, which method designed for coping with small disjuncts will obtain the best result. In this case different methods for coping with small disjuncts (such as C4.5/GA and double C4.5) would correspond to the classes to be predicted, and the predictor attributes would be characteristics of the data sets.

Another possible research direction consists of developing a fitness function that takes into account both the predictive accuracy and the simplicity of the candidate rules being evolved by the GA. (Note that in the current GA the issue of rule simplicity is dealt with by using a rule pruning operator, but this issue is ignored by the fitness function.) This can be done in at least two ways. First, one could use a weighted formula, where predictive accuracy and simplicity have different user-specified weights. This approach has the advantage of simplicity, but it has the disadvantage of introducing subjective user-specified parameters. Another approach would be to develop a multi-objective fitness function, in the Pareto sense [2]. This avoids the need for a subjective specification of weights, but it increases the complexity of the GA.

References

[1] Carvalho, D.R; Freitas, A.A. A genetic algorithm with sequential niching for discovering small-disjunct rules. *Proc. 2002 Genetic and Evolutionary Computation Conf. (Gecco-2002),*pp. 1035-1042. Morgan Kaufmann, 2002.

[2] Deb, K. Introduction to Selection, in Back, T, Fogel, D.B., Michalewicz (Eds), *Evolutionary Computation 1,*Philadelphia: Institute of Physics Publishing. 2000, pp. 166-171.

[3] Freitas, A.A. Understanding the crucial role of attribute interaction in data mining. *Artificial Intelligence Review 16(3)*, Nov. 2001, pp.177-199.

[4] Freitas, A.A. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer-Verlag, 2002.

[5] Hand, D.J. *Construction and Assessment of Classification Rules*, John Wiley & Sons. 1997.

[6] Holte, R.C.; Acker, L.E.; Porter, B.W. Concept Learning and the Problem of Small Disjuncts, *Proc. IJCAI – 89*, pp.813-818. 1989.

[7] Michie, D; Spiegelhalter, D.J. and Taylor, C.C. (Eds). *Machine Learning, Neural and Statistical Classification*, Ellis-Horwood, 1994.

[8] Quinlan, J.R. *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publisher. 1993.

[9] Romao, W.; Freitas, A.A. and Pacheco, R.C.S. A Genetic Algorithm for Discovering Interesting Fuzzy Prediction Rules: applications to science and technology data. *Proc. Genetic and Evolutionary Computation Conf. (GECCO-2002)*, pp. 1188-1195. Morgan Kaufmann, 2002.

[10] Weiss, G.M.; Hirsh, H. A Quantitative Study of Small Disjuncts, *Proc. of Seventeenth National Conference on Artificial Intelligence.* Austin, Texas, pp.665-670. 2000.