

CC4.5: cost-sensitive decision tree pruning

J. Cai¹, J. Durkin¹ & Q. Cai²

¹*Department of Electrical and Computer Engineering,
University of Akron, U.S.A.*

²*Department of Electrical Engineering and Computer Science,
Case Western Reserve University, U.S.A.*

Abstract

There are many methods to prune decision trees, but the idea of cost-sensitive pruning has received much less investigation even though additional flexibility and increased performance can be obtained from this method. In this paper, we introduce a cost-sensitive decision tree pruning algorithm called CC4.5 based on the C4.5 algorithm. This algorithm uses the same method as C4.5 to construct the original decision tree, but the pruning methods in CC4.5 are different from that in C4.5. CC4.5 includes three cost-sensitive pruning methods to deal with misclassification cost in the decision tree. Unlike many other pruning algorithms, CC4.5 uses intelligent inexact classification to consider both error and cost when pruning. Moreover, experiments show that CC4.5 results in improved decision trees with respect to the cost and its comprehensibility and accuracy are also satisfactory.

Keywords: decision tree pruning, cost-sensitive, C4.5, CC4.5, intelligent inexact classification.

1 Introduction

Decision tree technology has been proven to be a valuable way of capturing human decision making within a computer. But it often suffers the disadvantage of developing very large trees and making them incomprehensible to experts [1]. To solve this problem, researchers in the field have much interest in tree pruning [1, 2, 3]. Tree pruning methods transform a large tree into a small tree and make it easily understood. But one main problem for many traditional decision tree pruning methods is that when we prune a decision tree, we always assume that all mis-



classifications are equally probable and equally serious. However, in a real world classification problem, there is also a cost associated with misclassifying examples from each class. Cost-sensitive classification allows one to assign different costs to different types of misclassifications. Recently, some work has focused on the use of cost in decision tree pruning [4, 5, 6]. In this paper, a cost-sensitive decision tree pruning algorithm called CC4.5 which is based on the C4.5 algorithm is introduced. Also, three different cost-sensitive pruning methods in CC4.5 to deal with misclassification cost are proposed and analyzed. This paper is outlined as follows: Section 2 illustrates three different cost-sensitive pruning methods included in CC4.5; In section 3, we introduce CC4.5; Comparative analysis for these three pruning methods and evaluation between CC4.5 and other well-known pruning methods are described in section 4. Finally, the conclusion is given in section 5.

2 Cost-sensitive decision tree pruning

There are many decision tree pruning methods, but most of them assume that all misclassification costs are equally probable and equally important, so their goal is only to minimize the number of errors made when predicting the classification of unseen examples [7]. Unfortunately, it is not enough in reality. For example, a diagnosis mistake made for an old person may be much more serious than that made for a young person. Cost-sensitive decision tree pruning methods attempt to reduce the cost of misclassification when considering whether pruning or not. A cost matrix is used to define the misclassification costs [4]. Some methods have been proposed to minimize the misclassification cost [4, 5, 6]. In the following subsections, the descriptions of three cost-sensitive decision tree pruning methods are reported.

2.1 Intelligent inexact classification in expert system

We use the intelligent inexact classification technique for cost-sensitive pruning [8]. This technique identifies an unknown object by comparing information about the unknown object with information about a library of N known objects. The result of the comparison process forms a level of belief, F_n , for each object n out of N objects, that represents the degree of match between object n and the unknown object.

This technique relies on selecting a list of I object features to consider in the comparison process, determining the importance, α_i , of each i feature of I to the process, and the degree of similarity, β_{in} , between object n and the unknown object based on feature i . The level of belief that the library object n matches the unknown object is then found from

$$F_n = \frac{\sum_{i=1}^n \alpha_i \beta_{in}}{\sum_{i=1}^n \alpha_i} \quad (1)$$

where



α_i is the importance of factor i

β_{in} is a measure of the user's desire for factor-value i of product n .

We assume that both α and β are bound by 0 and 1, so F_n is also bound by 0 and 1.

$$0 \leq \alpha \leq 1, 0 \leq \beta \leq 1, 0 \leq F_n \leq 1 \quad (2)$$

where when

$\alpha = 0$ priority is definitely not true

$\alpha = 1$ priority is definitely true

$\beta = 0$ match is definitely not true

$\beta = 1$ match is definitely true

Object n with the highest F value is considered the best match for the unknown object.

It is clear the decision which we make from the above function is context-based. The parameters α_i and β_{in} reflect the context because, for different customers, the importance of a specific factor, α , and the measure of desire for that factor-value, β , may be changed. Even the same customers possibly change these two parameters under different conditions.

2.2 Using intelligent inexact classification in cost-sensitive pruning

Just as intelligent inexact classification is used in expert systems, we can use it when the task is to decide which of several errors of a given object are the least cost. The only difference for the intelligent inexact classification used in expert system and that used in cost-sensitive pruning is that the expert system needs the importance of factors, while the cost-sensitive technique needs the seriousness of errors. Then for decision tree pruning we can evaluate the error cost, c , from

$$C' = \frac{\sum_{i=1}^n \alpha_i p_i}{\sum_{i=1}^n \alpha_i} \quad (3)$$

where:

α_i is the seriousness of error i ,

p_i is a measure of the error possibility of error i if the tree is pruned.

where, when,

$\alpha = 0$ error is definitely not serious,

$\alpha = 1$ error is definitely serious.

We integrate the cost evaluation function in equation (3) with reduced error pruning. In this method, for every no-leaf subtree S of the original decision tree, we examine the change in misclassification cost over the pruning set that would occur if this subtree were replaced by the best possible leaf. If the cost of the new tree would be equal to or be smaller than that of the original tree and that subtree S contains no subtree with the same property, S is replaced by the leaf. Otherwise, stop the process. This method is called CC4.5-1. We rely upon the expert to set the values of α_i in the cost matrix.



2.3 Integrating cost and error rate in decision tree pruning

Most pruning techniques only consider the error rate while cost-sensitive pruning only considers the cost. But we can't say it is a good system if only its cost is low but its error rate is too high or its error rate is low but its cost is too high. So we consider using intelligent inexact classification to prune the decision tree based on not only considering error rate but also considering the cost of error. We propose two methods to deal with it:

1. The first method uses the following equation to decide whether to prune or not.

$$F = I_1 * C + I_2 * E \quad (4)$$

where:

I_1 is the weight of cost,

I_2 is the weight of error rate,

C is the error cost which we get from equation (3),

E is the error rate.

We will select to prune the decision tree if the result of equation (4) is decreased after pruning, otherwise we don't prune the tree. We call this pruning method CC4.5-2.

2. The second method sets the threshold values to make the pruning decision. In this method, threshold means the point at which a subtree can be pruned. We can get advice from experts to set a threshold value for cost which we get from equation (3) and another threshold value for error rate. Let c represent the threshold value for cost and e represent the threshold value for error rate. We can set the criteria as follows:

- (1) IF error rate decreases and cost decreases THEN pruning
- (2) IF error rate increases and cost increases THEN no pruning
- (3) IF error rate decreases and cost increases THEN:
 - (a) IF cost is equal to or less than c THEN pruning
 - (b) IF cost is larger than c THEN no pruning
- (4) IF error rate increases and cost decreases THEN:
 - (a) IF error rate is equal to or less than e THEN pruning
 - (b) IF error rate is larger than e THEN no pruning

We call this method CC4.5-3 and we also rely upon the expert to set the threshold values.

3 CC4.5

CC4.5 is a cost-sensitive decision tree pruning algorithm based on C4.5. CC4.5 integrates intelligent inexact classification, which we discussed previously, with C4.5 to deal with cost-sensitive pruning.

CC4.5 uses the same method as that of C4.5 to construct the original decision tree. But its pruning method is different from that of C4.5. CC4.5 includes

CC4.5-1, CC4.5-2 and CC4.5-3 three cost-sensitive pruning methods to deal with misclassification cost in the decision tree. Users can select one of three methods to prune the decision tree. CC4.5 uses separated pruning set to prune the original tree and considers cost during pruning. When we get decision tree, we can also get rules from trees as does C4.5. CC4.5 has more files than C4.5. The names file filestem.cost is a matrix defining costs of different misclassification errors. Each cost is separated by a space. In the same way, error and cost weights and thresholds can be set by users for a particular application.

4 Empirical comparison

We made a comparative study between some of the well known decision tree pruning techniques and the cost-sensitive pruning method in CC4.5. From the comparison, we identified the strengths and weaknesses of CC4.5.

Ideally, the pruned decision tree should be much more comprehensible than the original decision tree but should not be significantly less accurate when classifying unseen cases[1]. These are two important criteria, size and accuracy, to test how well a pruning method works. We add cost as the third criterion to test the pruning methods. The behavior of our pruning method was analyzed in a series of experiments. The databases which we used to test the pruning methods are available in the UCI Machine Learning Repository[9]. They are also very common in decision tree pruning experiments[1, 2, 3]. The data sets chosen are Pima, Hepatitis, Cleveland, Vote, Iris, and Glass. For the experiments, each data set was randomly divided into three parts: a growing set(49 percent), a pruning set(21 percent) and a test set(30 percent). Each database was run by each pruning method 10 times and the results have been summarized in tables 1-3. In these tables, REP represents reduced error pruning and PEP represents pessimistic pruning.

Table 1: Results of the tests on cost between cost-sensitive pruning method and other pruning methods.

database	REP	PEP	C4.5	CC4.5-1	CC4.5-2	CC4.5-3
Pima	0.6667	0.6759	0.6691	0.5076	0.5446	0.4934
Hepatitis	0.1408	0.1393	0.1502	0.146	0.1463	0.1373
Cleveland	0.2605	0.2699	0.2768	0.2621	0.2271	0.2426
Vote	0.0287	0.0270	0.0356	0.0237	0.0237	0.0206
Iris	0.0367	0.0321	0.0310	0.0363	0.0319	0.0213
Glass	0.2399	0.2074	0.2018	0.2537	0.2328	0.2319

We used equation (3) to calculate the cost and compared the cost which we got from each pruning method in table 1. It shows that most cost-sensitive pruning

Table 2: Results of the tests on error rate between cost-sensitive pruning method and other pruning methods (percent).

database	REP	PEP	C4.5	CC4.5-1	CC4.5-2	CC4.5-3
Pima	26.8	28.39	27.34	42.45	38.36	35.76
Hepatitis	18.47	21.52	21.3	23.92	23.69	25.87
Cleveland	46.22	44.82	49.11	48.01	46.01	45.67
Vote	5.22	4.21	5.55	3.84	3.85	3.44
Iris	7.72	6.66	6.67	7.95	4.99	3.94
Glass	40.47	34.36	34.21	48.28	39.84	44.98

Table 3: Results of the tests on size(number of leaves) between cost-sensitive pruning method and other pruning methods.

database	REP	PEP	C4.5	CC4.5-1	CC4.5-2	CC4.5-3
Pima	14.6	44.2	33.8	9.2	11	14.2
Hepatitis	3	17.8	9.6	6.4	7.8	7.4
Cleveland	41.8	75.7	64.4	31.6	35.6	34.6
Vote	4	4.3	4.6	4	3.7	4.6
Iris	5	7.2	6.6	5.2	11	14.6
Glass	14.6	39.4	35	15.8	15	13.8

methods get the lower cost than other non cost-sensitive pruning methods. The cost-sensitive pruning method which sets values of threshold(CC4.5-3) achieves the lowest cost. From the results of these experiments we also notice that pessimistic pruning, PEP, and C4.5 sometimes have a very good performance in cost. The reason is that pessimistic pruning and C4.5 does not require a pruning set separate from the cases in the training set from which the tree was constructed. The disadvantage of pessimistic pruning and C4.5 is that they tend to produce bigger decision trees than other pruning, especially than cost-sensitive pruning(table 3).

We compared results of the tests on error rate between cost-sensitive pruning methods and other pruning methods in table 2. From table 2, we find that when we only consider cost in pruning(CC4.5-1), error rate appears greater than the cost-sensitive pruning methods which consider both cost and error rate(CC4.5-2 and CC4.5-3). In most times, other well-known pruning methods are of superior accuracy to the cost-sensitive pruning. But, at least, we get lower cost from cost-sensitive pruning.

The results of the tests on size are reported in table 3. We use the number of leaves to represent the size of the decision tree. It is clear that cost-sensitive pruning tends to produce smaller decision trees than the other well-known pruning methods we tested.

5 Summary

In this paper, a cost-sensitive decision tree pruning algorithm called CC4.5 based on the C4.5 algorithm has been introduced. In particular, its performance has been compared with those of several well-known pruning methods. The strength and weakness of CC4.5 has also been pointed out.

It is clear that our cost-sensitive pruning methods require pruning sets separated from the training sets from which the tree was constructed. Our future work will focus on the integration of cost-sensitive pruning methods in other pruning methods, e.g. cost-complexity pruning, or pessimistic pruning, which do not require separate pruning sets. We also plan to compare CC4.5 pruning methods to other pruning methods in the near future.

References

- [1] Quinlan, J.R., Simplifying decision trees. *Int J Human-Computer Studies*, **51(2)**, pp. 497–510, 1999.
- [2] Mingers, J., An empirical comparison of selection measures for decision tree induction. *Machine Learning*, **3(3)**, pp. 319–342, 1989.
- [3] F. Esposito, D.M. & Semeraro, G., A comparative analysis of methods for pruning decision trees. *IEEE transactions on pattern analysis and machine intelligence*, **19(5)**, pp. 476–491, 1997.
- [4] U. Knoll, G.N. & Tausend, B., Cost sensitive pruning of decision trees. *Proceedings of ECML-94*, eds. F. Bergadano & L. De Raedt, Springer: Berlin, Heidelberg, pp. 383–386, 1994.
- [5] Andrew, B. & C, L.B., Cost-sensitive decision tree pruning use of the roc curve. *Proc. of the 8th Australian Joint Conference on Artificial Intelligence*, ed. X. Yao, World Scientific Publ. Co.: Singapore, pp. 1–8, 1995.
- [6] Turney, P., Cost-sensitive classification: empirical evaluation of a hybrid genetic decision tree induction algorithm. *Journal of Artificial Intelligence Research*, **2**, pp. 369–409, 1995.
- [7] Pazzani, M. & Merz, C., Reducing misclassification costs. *Proc. of the 11th International Conference on Machine Learning*, eds. W. Cohen & H. Hirsh, Morgan Kaufmann: San Francisco, pp. 217–225, 1994.
- [8] Durkin, J., Intelligent inexact classification. *Expert System Note*, ed. J. Durkin, University of Akron, 1990.
- [9] Blake, C. & Merz, C., <http://www.ics.uci.edu/~mllearn/mlrepository.html>. *UCI Repository of Machine Learning Databases*, University of California, Irvine, 1998.

