Cost-Sensitive Test Strategies

Victor S. Sheng, Charles X. Ling
Department of Computer Science
The University of Western Ontario
London, Ontario N6A 5B7, Canada
{ssheng, cling}@csd.uwo.ca

Ailing Ni
School of Computer Science
Anhui University of Technology
China

Shichao Zhang
Department of Automatic Control
Beijing University of Aeronautics
and Astronautics, China

Abstract

In medical diagnosis doctors must often determine what medical tests (e.g., X-ray, blood tests) should be ordered for a patient to minimize the total cost of medical tests and misdiagnosis. In this paper, we design cost-sensitive machine learning algorithms to model this learning and diagnosis process. Medical tests are like attributes in machine learning whose values may be obtained at cost (attribute cost), and misdiagnoses are like misclassifications which may also incur a cost (misclassification cost). We first propose an improved decision tree learning algorithm that minimizes the sum of attribute costs and misclassification costs. Then we design several novel "test strategies" that may request to obtain values of unknown attributes at cost (similar to doctors' ordering of medical tests at cost) in order to minimize the total cost for test examples (new patients). We empirically evaluate and compare these test strategies, and show that they are effective and that they outperform previous methods. A case study on heart disease is given.

Introduction

Inductive learning techniques have had great success in building classifiers and classifying test examples into classes with a high accuracy or low error rate. However, in many real-world applications, reducing misclassification errors is not the final objective, since different error can cost quite differently. This type of learning is called cost-sensitive learning. (Turney 2002) surveys a whole range of costs in cost-sensitive learning, among which two types of costs are most important: misclassification costs and attribute costs. For example, in a binary classification task, the cost of false positive (*FP*) and the cost of false negative (*FN*) are often very different. In addition, attributes (similar to medical tests) may have different costs, and acquiring values of attributes may also incur costs. The goal of learning in this paper is to minimize the sum of the misclassification costs and the attribute costs.

Tasks involving both misclassification and attribute costs are abundant in real-world applications. In medical diagnosis, medical tests are like attributes in machine learning whose values may be obtained at cost (attribute cost), and misdiagnoses are like misclassifications which may also bear a cost (misclassification cost). When building a classification model for medical diagnosis from the training data, we must consider both the attribute costs (medical tests such as blood tests) and misclassification costs (errors in the diagnosis). Further, when a doctor sees a new patient (a test example), additional medical tests may be ordered at cost to better diagnose or predict the disease of the patient (i.e., reducing the misclassification cost). We use the term "test strategy" to describe a process that allows the learning algorithm to obtain attribute values at cost when classifying test examples. The goal of the test strategies in this paper is to minimize the sum of attribute costs and misclassification costs, similar to doctors' goal to minimize the total cost to the patients (or the whole medical system). A case study on heart disease is given in the paper.

In this paper, we first propose an improved decision tree learning that minimizes the sum of misclassification costs and attribute costs. We then describe several novel test strategies to determine what values of unknown attributes should be obtained, and at what order, such that the total expected cost is minimum. Extensive experiments have been conducted to show the effectiveness of our tree building algorithms and the new test strategies compared to previous methods.

Review of Previous Work

(Gorry and Barnett 1968) suggested a "myopic approach" to request more information during decision making, but the method focuses only on prediction accuracy; there is no cost involved in getting the information. Cost-sensitive learning has received extensive attentions in recent years. Much work has been done in considering non-uniform misclassification costs (alone), such as (Elkan 2001). Those works can often be used to solve problem of learning with imbalanced datasets (Blake and Merz 1998). Some previous work, such as (Nunez, 1991, Tan 1993), considers the attribute costs alone without incorporating misclassification costs. As pointed out by (Turney 2000) it is obviously an oversight. As far as we know, the only work considering both misclassification and attribute costs includes (Turney 1995, Zubek and Dietterich 2002, Greiner et al. 2002, Ling et al. 2004, Chai et al. 2004). We discuss these works in detail below.

In (Zubek and Dieterrich 2002), the cost-sensitive learning problem is cast as a Markov Decision Process (MDP). They adopt an optimal search strategy, which may incur a high computational cost. In contrast, we adopt the local search similar to C4.5 (Quinlan 1993), which is very efficient. (Greiner et al. 2002) studied the theoretical aspects of active learning with attribute costs using a PAC learning framework, which models how to use a budget to collect the relevant information for the real-world

applications with no actual data at beginning. (Turney 1995) presents a system called ICET, which uses a genetic algorithm to build a decision tree to minimize the cost of tests and misclassifications. Our algorithm is expected to be more efficient than Turney's genetic algorithm.

(Chai et al. 2004) proposed a naïve Bayesian based cost-sensitive learning algorithm, called CSNB, which reduces the total cost of attributes and misclassifications. Their test strategies are quite simple. We propose an improved decision tree algorithm that uses minimum total cost of tests and misclassifications as the attribute split criterion. We also incorporate discounts in attribute costs when tests are performed together. Most important, we propose a novel single batch strategy and a Multiple Batch strategy that, as far as we know, have not been published previously. Experiments show that our tree-based test strategies outperform naïve Bayes strategies in terms of the total cost in many cases (see later sections).

Our work is also significantly different from the previous work in attribute value acquisition. (Melville et al., 2004; 2005) proposed attribute value acquisition during training, instead of testing as in our paper. Their algorithm is sequential in nature in requesting the missing values, instead of in batches. In addition, their goal is to reduce misclassification errors, not the total cost as in our paper.

Minimum Cost-Sensitive Decision Trees

We assume that we are given a set of training data, the misclassification costs, and test costs for each attribute. (Ling et al, 2004) propose a new cost-sensitive decision tree algorithm that uses a new splitting criterion of *cost reduction* on training data, instead of minimal entropy (as in *C4.5*), to build decision trees. The cost-sensitive decision tree is similar to C4.5 (Quinlan, 1993), except that it uses total cost reduction, instead of entropy reduction, as the attribute split criterion. More specifically, the total cost before and after splitting on an attribute can be calculated, and the difference is the cost reduction "produced" by this attribute. The total cost before split is simply the misclassification cost of the set of examples. The total cost after split on an attribute is the sum of misclassification cost of subsets of examples split by the attribute values, plus the attribute cost of these examples. The attribute with the maximal positive cost reduction is chosen as the root, and the procedure recursively applies to the subsets of examples split by the attribute values.

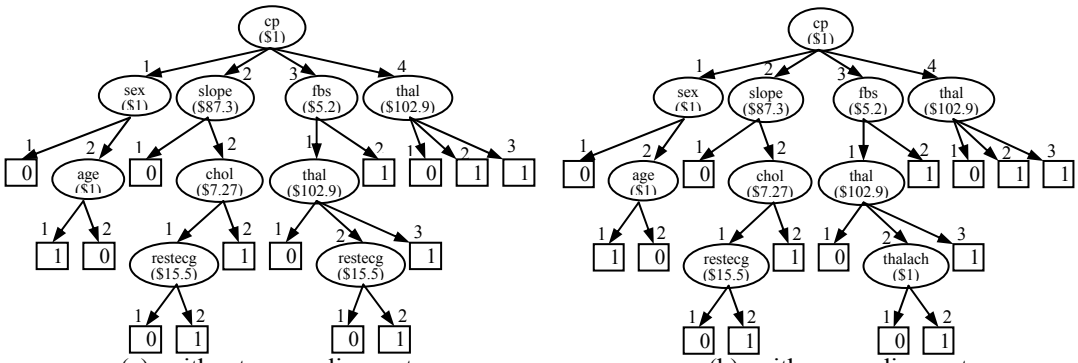
We improve (Ling et al, 2004)'s algorithm by incorporating possible *discounts* when obtaining values of a group of attributes with missing values in the tree building algorithm. This is a special case of conditional attribute costs (Turney 1995), which allows attribute costs to vary with the choice of prior tests. Often medical tests are not performed independently, and when certain medical tests are performed together, it is cheaper to do these tests in group than individually. For example, if both tests are blood tests, it would be cheaper to do both tests together than individually. In our case we assume that attributes can be partitioned into groups, and each group has a particular discount amount. When the first attribute in a group is requested for testing, the attribute cost is its original cost. However, if any additional attributes in the same group are requested for their values, their costs would be the original costs minus the discounted cost. In implementing the cost-sensitive decision-tree building process, if an attribute in a group is selected as a split attribute, the costs of other attributes in the group are simultaneously reduced by the discount amount for the future tree-building process. As the attribute costs are discounted, the tests in the same group would more likely be picked as the next node in the future tree building.

A Case Study on Heart Disease

We apply our cost-sensitive decision-tree learning on a real application example that involves the diagnosis of the Heart Disease, where the attribute costs are obtained from medical experts and insurance programs. The dataset was used in the cost-sensitive genetic algorithm by (Turney 1995). The learning problem is to predict the coronary artery disease from the 13 non-invasive tests on patients, as listed in Table 1. The attributes on patients profile, such age, sex, etc., are also regarded as "tests" with a very low cost (such as $1) to obtain their values. The costs of the 13 non-invasive tests are in Canadian dollars ($), and were obtained from the Ontario Health Insurance Program's fee schedule (Turney 1995). These individual tests and their costs are also listed in Table 1. Tests such as exang, oldpeak, and slope are electrocardiography results when the patient runs on a treadmill, and are usually performed as a group. Tests done in a group are discounted in costs, and Table 1 also lists these groups and the discount amount of each group. Each patient can be classified into two classes: the class label 0 or negative class indicates a less than 50% of artery narrowing; and 1 indicates more than 50%. There are a total of 294 cases in the dataset, with 36.1% positive cases (106 positive cases).

Table 1: Attribute costs (in $) and group discounts for Heart disease.

Tests	Description	Individual Costs	Group A discount	Group B discount	Group C discount
age	age of the patient	$1.00			
sex	sex	$1.00			
cp	chest pain type	$1.00			
trestbps	resting blood pressure	$1.00			
chol	serum cholesterol in mg/dl	$7.27	$2.10		
fbs	fasting blood sugar	$5.20	$2.10		
restecg	resting electrocardiography results	$15.50			
thalach	maximum heart rate achieved	$102.90		$101.90	
exang	exercise induced angina	$87.30			$86.30
oldpeak	ST depression induced by exercise	$87.30			$86.30
slope	slope of the peak exercise ST segment	$87.30			$86.30
Ca	number of major vessels colored by fluoroscopy	$100.90			
thal	finishing heart rate	$102.90		$101.90	



(a): without group discount (b): with group discount

Figure 1: Cost-sensitive decision trees. Continuous attributes have been discretized and are represented by 1, 2, and so on.

However, no information about misclassification costs was given. After consulting a researcher in the Heart-Failure Research Group in the local medical school, a positive prediction normally entails a more expensive and invasive test, the angiographic test, to be performed, which more accurately measures the percentage of artery narrowing. A negative prediction may prompt doctors to prescribe medicines, but the angiographic test may still be ordered if other diseases (such as diabetes) exist. An angiographic attribute costs about $600. Thus, it seems reasonable to assign false positive and false negative to be $600 and $1,000 respectively.

When applying the cost-sensitive decision tree algorithm on the Heart dataset, we obtain the trees as shown in Figure 1. In the tree, the branch labels (1, 2, …) are discretized values: the numerical attributes in the dataset are discretized into integers (1, 2, …) using the minimal entropy method of (Fayyad and Irani 1993). Figure 1 (a) is the tree without considering group discounts, while Figure 1 (b) is the tree considering group discounts (as in Table 1). We can easily see that the two trees are very similar, except for one test in Figure 1 (b), thalach, which replaces the test restecg in Figure 1 (a). Since tests thal and thalach belong to the same group, after thal is tested at the full cost, thalach costs only $1 and is selected after thal when group discounts are considered. This would clearly reduce the total cost when the tree in Figure 1 (b) classifies test examples (see later).

Also we can see that in general less expensive tests (attributes) are used in the top part of the trees. The splitting criterion selects attributes according to their relative merit of reducing the total cost. When these trees are presented to the Heart-Failure researcher, he thinks that they are quite reasonable in predicting artery narrowing. We will revisit the case study with each novel test strategy proposed in following sections.

Three Categories of Test Strategies

A "test strategy" is a method to classify a test example from a cost-sensitive decision tree during which it is possible to "perform tests", that is, to obtain values of unknown attributes, at cost (attribute costs). It mimics the diagnosis process of doctors in which additional medical tests may be requested at cost to help their diagnoses (predicting the disease of the patients).

In this paper we define and study three categories of test strategies representing three different cases in diagnosis: Sequential Test, Single Batch Test, and Multiple Batch Test. For a given test example (a new patient) with unknown attributes, the Sequential Test can request only one test (attribute value) at a time, and wait for the test result to decide which attribute to be tested next, or if a final prediction is made. The Single Batch Test, on the other hand, can request one set (batch) of one or many tests to be done simultaneously before a final prediction is made. The Multiple Batch Test can request a series of batches of tests, each after the results of the previous batch are known, before making a final prediction. Clearly, the Multiple Batch Test is most general, as the other two are special cases of it – Sequential Test is when each batch only contains one test, and Single Batch is when the number of batches can only be one.

The three categories of test strategies correspond well to different cases of diagnosis in the real world. In medical diagnoses, for example, doctors normally request one set of tests (at cost) to be done. This is the case of the Single Batch Test. Sometimes doctors may order a second (and a third) set of tests to be done, based on the results of the previous set of tests. This is the case of the Multiple Batch Test. If doctors only order one test at a time (this can happen if tests are very expensive and/or risky), this is the case of the Sequential Test. The goal of these test strategies again is to minimize the total cost of tests (attributes) requested and the misclassification costs associated with the final prediction. In the next three subsections the three types of test strategies will be discussed in great detail.

Strategy 1: Sequential Test

We propose a simple Sequential Test strategy (Ling et al. 2004) that directly utilizes the decision tree built to guide the sequence of tests to be performed in the following way: when the test example is classified by the tree, and is stopped by an attribute whose value is unknown, a test of that attribute is requested and made at cost. This process continues until the test case reaches a leaf of the tree.

According to the leaf reached, a prediction is made, which may incur a misclassification cost if the prediction is wrong. Clearly the time complexity of the strategy is only linear to the depth of the tree.

Note that Sequential Test is near "optimal" by the nature of the decision tree built to minimize the total cost; that is, subtrees are built because there is a cost reduction in the training data. Thus, the tree's suggestions for tests should also result in near minimum total cost in the test case.

Case Study on Heart Disease Continued. We choose a test example with most attribute values known from the dataset, as the known values serve as the test results. We apply Sequential Test on the tree in Figure 1(b) which considers the group discount to the test case. Assuming all values are unknown, Sequential Test requests the sequence of tests as: cp, fbs, thal, and thalach, with a total attribute cost of $110.10, while the misclassification cost is $0. Therefore, the total cost for this test case is $110.10.

Strategy 2: Single Batch Test

In Sequential Test described earlier one must wait for the result of each test to determine which test will be the next one. Waiting not only agonizes patients in medical diagnosis, it may also be life threatening if the disease is not diagnosed and treated promptly. Thus doctors normally order one set of tests to be done at once. This is the case of the Single Batch Test. Note that results of the tests in the batch can only be obtained simultaneously after the batch is determined.

In this section we propose a novel Single Batch strategy. The Single Batch seeks a set of tests to be performed such that the sum of the attribute costs and expected misclassification cost after those tests are done is optimal (minimal). Intuitively, it finds the expected cost reduction for each unknown attribute (test), and adds a test to the batch if the expected cost reduction is positive and maximum (among other tests). This process is continued until the maximum cost reduction is no longer greater than 0, or there is no reachable unknown attributes. The batch of tests is then discovered. However, Single Batch is a guess work. Often some tests requested are wasted, and the test example may not be classified by a leaf node (in this case it is classified by an internal node in the tree). The pseudo-code of the Single Batch is shown here.

In the pseudo-code, *misc(.)* is the expected misclassification cost of a node, *c(.)* is the attribute cost, *R(.)* is all reachable unknown nodes and leaves under a node, and *p(.)* is the probability (estimated by ratios in the training data) that a node is reached. Therefore, the formula *E(i)* in the pseudo-code calculates the cost difference between no test at *i* (so only misclassification cost at *i*) and after testing *i* (the attribute cost plus the weighted sum of misclassification costs of reachable nodes under *i*). That is, *E(i)* is the expected cost reduction if *i* is tested. Then the node *t* with the maximum cost reduction is found, and if such reduction is positive, *t* should be tested in the batch. Thus, *t* is removed from *L* and added into the batch list *B*, and all reachable unknown nodes or leaves of *t*, represented by the function *R(t)*, is added into *L* for further consideration. This process continues until there is no positive cost reduction or there is no unknown nodes to be considered (i.e., *L* is empty). The time complexity of the Single Batch is linear to the size of the tree, as each node is considered only once.

L = empty /* list of reachable and unknown attributes */
B = empty /* the batch of tests */
u = the first unknown attribute when classifying a test case
Add *u* into *L*
Loop
  For each $i \in L$, calculate *E(i)*:
    $E(i) = misc(i) - [c(i) + \sum p(R(i)) \times misc(R(i))]$
  E(t) = max E(i) /* *t* has the maximum cost reduction */
  If *E(t)* > 0 then add *t* into *B*, delete *t* from *L*, add *R(t)* into *L*
  else exit Loop /* No positive cost reduction */
Until *L* is empty
Output *B* as the batch of tests

Case Study on Heart Disease Continued. We choose the same test example to study the Single Batch with the decision tree in Figure 1(b). The Single Batch suggests a single batch of (cp, sex, slope, fbs, thal, age, chol, and restecg) to be tested. The test example did not go into a leaf, and some tests are wasted. The total attribute cost for this case is $221.17, while the misclassification cost is 0. Thus, the total cost for the test case is $221.17.

Strategy 3: Multiple Batch Test

The Multiple Batch Test naturally combines the Sequential Test and the Single Batch, in that multiple batches of tests can be requested in sequence. To make the Multiple Batch Test meaningful, one must assume and provide a "batch cost", the extra cost of each additional batch test (there is no batch cost for the first batch). When the batch cost is set as 0, then Multiple Batch should become Sequential Test, as it is always better to request one test at a time before the next request. In other words, if waiting costs nothing, one should never request multiple tests at the same time, as some tests may be wasted (as in Single Batch), thus increasing the total cost. If the batch cost is infinitely large, then one can only request one batch of tests, thus Multiple Batch becomes Single Batch.

Here we extend Single Batch described in the last subsection to Multiple Batch. Recall that in Single Batch, an unknown attribute is added into the batch if the successive cost reduction of testing it is positive and maximum among the current reachable unknown attributes. In Multiple Batch, we include an additional constraint: attributes added in the current batch must improve the accumulative ROI (return on investment), which considers the batch cost. The ROI is defined as $ROI = \frac{\sum Costreduction}{BatchCost + \sum AttributeCost}$.

The rationale behind this (heuristic) strategy is that attributes that bring a larger ROI should be worth including in the same batch test. After the current batch of tests is determined and tested with values revealed, the test example can be classified further down in the tree

according to the test results until it is stopped by another unknown attribute. The same process then applies, until no more batches of tests are required. The time complexity of this strategy is linear to the size of the tree, as each node in the tree would be considered at most once.

The algorithm described above is heuristic but it is close to the ideal one: if the batch cost is $0, then usually only one test will be added in the batch, and the strategy is very similar to the Sequential Test. On the other hand, if the batch cost is very large, the current batch will grow until the cost reduction of the remaining unknown attributes is no longer greater than 0, and the strategy is similar to the Single Batch. See experimental comparison later.

Case Study on Heart Disease Continued. We apply Multiple Batch on the same test example with the tree in Figure 1 (b). Assuming the batch cost is $50.00, the strategy decides that two batches of tests are needed for the test case. The first batch has just two tests, cp and fbs. After the values of cp and fbs are obtained, the second batch also contains two tests, thal and thalach. The misclassification cost is 0, while the total attributes costs for the test case is $161.1 (including the batch cost of $50.00). Thus the total cost for the test case is also $161.1.

Note that based on this single test case, we cannot simply conclude that in general Sequential Test is best, Multiple Batch is second and Single Batch is worst. The experimental comparison in the following section will answer this question.

Experimental Comparisons

To compare the overall performance of the three test strategies, we choose 10 real-world datasets, listed in Table 2, from the UCI Machine Learning Repository (Blake and Merz 1998). These datasets are chosen because they are binary class, have at least some discrete attributes, and have a good number of examples. To create datasets with more imbalanced class distribution, two datasets (thyroid and kr-vs-kp) are resampled to create a small percentage of positive examples. They are called thyroid_i and kr-vs-kp_i respectively. Each dataset is split into two parts: the training set (60%) and the test set (40%). Unlike the case study of heart disease, the attribute costs and misclassification costs of these datasets are unknown. To make the comparison possible, we simply assign certain values for these costs. We assign random values between $0 and $100 as attribute costs for all attributes. This is reasonable because we compare the relative performance of all test strategies under the same assigned costs. The misclassification cost FP/FN is set to $2,000/$6,000 ($2,000 for false positive and $6,000 for false negative) for the more balanced datasets (the minority class is greater than 10%) and $2,000/$30,000 for the imbalanced datasets (the minority class is less than 10% as in thyroid_i and kr-vs-kp_i). The group discount of attributes is not considered. To make the comparison complete, the heart disease dataset used in the case study (called Heart-D at Table 2) is also added in the comparison (with its own attribute costs). For test examples, a certain ratio of attributes (0.2, 0.4, 0.6, 0.8, and 1) are randomly selected and marked as unknown to simulate test cases with various degrees of missing values.

Table 2: The features of 13 Datasets.

	No. of Attributes	No. of Examples	Class dist. (N/P)
Ecoli	6	332	230/102
Breast	9	683	444/239
Heart	8	161	98/163
Thyroid	24	2000	1762/238
Australia	15	653	296/357
Tic-tac-	9	958	332/626
Mushroo	21	8124	4208/3916
Kr-vs-kp	36	3196	1527/1669
Voting	16	232	108/124
Cars	6	446	328/118
Thyroid_i	24	1939	1762/167
Kr-vs-	36	1661	1527/134
Heart-D	13	294	188/106

Comparing the Three New Test Strategies

We compare Sequential Test, Single Batch, and Multiple Batch with the batch cost to be $0, $100 and $200 on the 13 datasets listed in Table 2. The results are presented in Figure 2. From the figure we can draw several interesting conclusions. First, when the ratio of missing values increases, the total cost of all the three strategies also increases, as expected. This is because it costs more when requesting more missing values in the test examples. Second, the total cost of Sequential Test is lowest, followed closely by Multiple Batch with 0 batch cost (B=0). As we discussed earlier, Multiple Batch should become Sequential Test if the batch cost is 0. The small difference between the two is probably due to the heuristics used in Multiple Batch. Third, Single Batch is worse than Sequential Test (and Multiple Batch with B=0). This is because requesting multiple tests in a batch is a guess work. Often some tests are requested but wasted, while other useful tests are not requested, increasing the total cost. Fourth, Single Batch is better than Multiple Batch with B=100 and 200. This might be a bit surprising at first glance. As we discussed earlier, Multiple Batch should become Single Batch as the worst case when the batch cost is infinitely large. However, when the batch cost is large but not infinitely large, usually more than one batch is requested by Multiple Batch, in which case the batch cost is added to the total cost. This batch cost is "extra" to test examples when comparing to Single Batch (and Sequential Test).

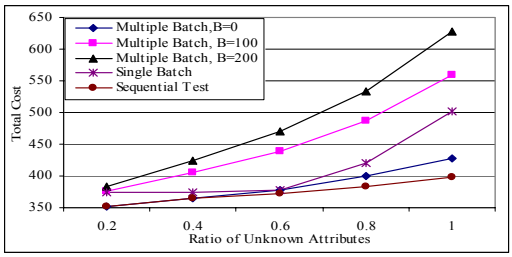


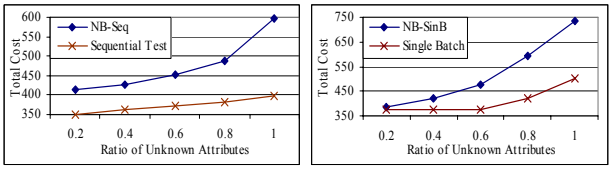
Figure 2: Comparing the total costs for the three new test strategies. The smaller the total cost, the better.

To conclude, the experiments in this section confirm our expectations on the three new test strategies: Sequential Test is best as it requests only one test at a time before requesting next. Multiple Batch resembles closely to

Sequential Test when the batch cost is zero, and it becomes worse when the batch cost is large.

Comparing with CSNB Strategies

We first compare our Sequential Test with the sequential test strategy in naïve Bayes (called NB-Seq in short), proposed in (Chai et al. 2004), under the same experimental setting. The average total costs (in $) for the 13 datasets are plotted in Figure 3 (a). We can clearly see that tree-based Sequential Test outperforms NB-Seq on average on the 13 datasets (balanced or imbalanced) under all unknown attribute ratios. This is because Sequential Test takes advantages of the minimum cost decision tree, but NB-Seq does not.



(a): Sequential Test. (b): Single Batch.

Figure 3 (a): Comparing tree-based Sequential Test with NB-Seq. Figure 3 (b): Comparing tree-based Single Batch with NB-SinB. The smaller the total cost, the better.

Next we compare our Single Batch with the naïve Bayes single batch (called NB-SinB in short) proposed in (Chai et al. 2004). The average total costs (in $) for the 13 datasets are plotted in Figure 3(b). From the figure we can see again that our tree-based Single Batch outperforms naïve Bayes NB-SinB. The reason is again that the minimum cost decision tree is utilized when deciding the single batch, while naïve Bayes has no such structure to rely on. Last, our tree-based test strategies are about 60 to 300 times faster than naïve Bayes strategies (details not shown here).

Conclusions and Future Work

In this paper, we present an improved decision tree learning algorithm with cost reduction as the attribute split criterion to minimize the sum of misclassification costs and attribute costs. We then design three categories of test strategies: Sequential Test, Single Batch, and Multiple Batch, to determine which unknown attributes should be tested, and in what order, to minimize the total cost of tests and misclassifications. The three test strategies correspond well to three different policies in diagnosis. We evaluate the performance of the three test strategies (in terms of the total cost) empirically, and compare them to previous methods using naïve Bayes. The results show that the new test strategies perform well. The time complexity of these new test strategies is linear to the tree depth or the tree size, making them efficient for testing a large number of test cases. These strategies can be readily applied to large datasets in the real world. A detailed case study on heart disease is given in the paper.

In our future work, we plan to continue to work with medical doctors to apply our algorithms to medical data.

References

Blake, C.L., and Merz, C.J. 1998. *UCI Repository of machine learning databases (website)*. Irvine, CA: University of California, Department of Information and Computer Science.

Chai, X., Deng, L., Yang, Q., and Ling,C.X.. 2004. Test-Cost Sensitive Naïve Bayesian Classification. *In Proceedings of the Fourth IEEE International Conference on Data Mining*. UK : IEEE Computer Society Press.

Elkan, C. 2001. The Foundations of Cost-Sensitive Learning. *In Proceedings of the 17th International Joint Conference of Artificial Intelligence*, 973-978. Seattle.

Fayyad, U.M., and Irani, K.B. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. *In Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1022-1027. France: Morgan Kaufmann.

Gorry, G. and Barnett, G. 1968. "Experience with a model of sequential diagnosis", *Computers and Biomedical Research*.

Greiner, R., Grove, A., and Roth, D. 2002. Learning cost-sensitive active classifiers. *Artificial Intelligence,* 139(2): 137-174.

Ling, C.X., Yang, Q., Wang, J., and Zhang, S. 2004. Decision Trees with Minimal Costs. *In Proceedings of the Twenty-First International Conference on Machine Learning,* Banff, Alberta: Morgan Kaufmann.

Melville, P., Saar-Tsechansky, M., Provost, F., and Mooney, R.J. 2004. Active Feature Acquisition for Classifier Induction. *In Proceedings of the Fourth International Conference on Data Mining*. Brighton, UK.

Melville, P., Saar-Tsechansky, M., Provost, F., and Mooney, R.J. 2005. Economical Active Feature-value Acquisition through Expected Utility Estimation. *UBDM Workshop, KDD 2005*.

Nunez, M. 1991. The use of background knowledge in decision tree induction. *Machine learning*, 6:231-250.

Quinlan, J.R. eds. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Tan, M. 1993. Cost-sensitive learning of classification knowledge and its applications in robotics. *Machine Learning Journal,* 13:7-33.

Turney, P.D. 1995. Cost-Sensitive Classification: Empirical Evaluation of a Hybrid Genetic Decision Tree Induction Algorithm. *Journal of Artificial Intelligence Research* 2:369-409.

Turney, P.D. 2000. Types of cost in inductive concept learning. *In Proceedings of the Workshop on Cost-Sensitive Learning at the 17th ICML*, California.

Zubek, V.B., and Dietterich, T. 2002. Pruning improves heuristic search for cost-sensitive learning. *In Proceedings of the Nineteenth International Conference of Machine Learning,* 27-35, Sydney, Australia: Morgan Kaufmann.