論 文

Technical Papers

# An Interactive-Graphic System for Decision Tree Induction

Trong Dung Nguyen* Tu Bao Ho*

Received MM DD, 19YY.

**Keywords:** decision tree induction, attribute selection measure, model selection, interactive learning.

**Summary**

The objectives of this paper are threefold. First to introduce R-measure as an attribute selection measure in decision tree induction (DTI). Second to introduce a model selection tool using a well designed benchmark that allows the user to carry out an experimental comparative evaluation of different DTI methods/models in order to select the appropriate one for a given task. Third to introduce the development of the interactive-graphic system CABRO, particularly its Tree Visualizer and Interactive Learning Mode.

## 1. Introduction

The performance of a decision tree induction (DTI) system depends principally on methods to solve three problems : *attribute selection, pruning*, and *discretization*. Most measures for attribute selection in DTI are either information theory-based such as information gain and gain-ratio [Quinlan 93], or statistics-based such as $\chi^2$, gini-index [Breiman 84], etc. Stemming from the limitation of the deterministic model of rough set theory [Pawlak 91] when dealing with uncertain information, the first objective of our work is to find a rough set-based measure for attribute selection (*R-measure*).

It is commonly agreed that there is no universally superior learning method or model. The problem of *model selection* is that of choosing the appropriate learning method/model for a given application task. Most DTI systems currently do not provide the user support in model selection. This task is difficult for most users as it requires many empirical comparative evaluations and/or meta-knowledge. Recently, there have been many experimental comparative evaluations of DTI methods for attribute selection measures, pruning and discretization, e.g. [Minger 89a], [Minger 89b], [Dougherty 95]. Several authors have warned that experimental studies in machine learning should be done more carefully in order to obtain valid evaluations, e.g. [Salzberg 97]. Though the multiple cross-validation experiments on a large number of datasets provide reliable evaluations, they are costly and not always done well. The second objective of our work is to develop an interactive tool that supports doing model selection and experimental comparative evaluation of DTI methods/models based on a carefully designed benchmark.

Though the decision tree is a simple notion it is difficult to understand a tree of big size. For example, the well-known DTI program C4.5 [Quinlan 93] produces a pruned tree of 2,464 leaf nodes from the census bureau database, introduced recently to the Knowledge Discovery Nuggets Directory, that consists of 199,523 instances described by 40 numeric and symbolic attributes (103 Mbytes). It is very difficult to understand and verify that big tree in the text form (about 18,500 lines, i.e. about

18,500 decision and leaf nodes). In such cases, a graphic visualisation of trees with different ways of view and navigation is of great support for the user. The third objective of this work is to develop an *interactive-graphic* DTI system CABRO for DTI.

The paper is organized as follows. In section 2 we briefly recall some basic notions in rough set theory and DTI. In section 3 we propose R-measure as a DTI attribute selection measure. In section 4 we describe the model selection tool and report experimental comparative studies of R-measure using this tool. In section 5 we describe the current development of CABRO. Finally, section 6 gives the conclusion and what we feel to be worth to pursue.

## 2. Preliminaries

### 2・1 *Attribute dependency measure in rough set theory*

The starting point of rough set theory [Pawlak 91] is the assumption that our "view" on elements of an object set $O$ depends on an indiscernibility relation among them, that means an equivalence relation $E \subseteq O \times O$. Two objects $o_1, o_2 \in O$ are said to be *indiscernible* w.r.t $E$ if $o_1 E o_2$. The *lower* and *upper* approximations of any $X \subseteq O$, w.r.t. an equivalence relation $E$, are defined as

$$E_*(X) = \{o \in O : [o]_E \subseteq X\} \quad (1)$$

$$E^*(X) = \{o \in O : [o]_E \cap X \neq \emptyset\} \quad (2)$$

where $[o]_E$ denotes the equivalence class of objects which are indiscernible with $o$ w.r.t the equivalence relation $E$. A subset $P$ of the set of attributes used to describe objects of $O$ determines an equivalence relation that divides $O$ into equivalence classes each containing objects having the same values on all attributes of $P$. A key concept in the rough set theory is the *degree of dependency* of a set of attributes $Q$ on a set of attributes $P$, denoted by $\mu_P(Q)$ $(0 \leq \mu_P(Q) \leq 1)$, defined as

$$\mu_P(Q) = \frac{|\bigcup_{[o]_Q} P_*([o]_Q)|}{|O|} \quad (3)$$

If $\mu_P(Q) = 1$ then $Q$ totally depends on $P$; if $0 < \mu_P(Q) < 1$ then $Q$ partially depends on $P$; if $\mu_P(Q) = 0$ then $Q$ is independent of $P$. The measure of dependency is fundamental in rough set theory as based on it important notions are defined, such as reducts and minimal sets of attributes, significance of attributes, etc.

Table 1 given by Pawlak consists of eight objects described by two descriptive attributes *Temperature, Headache*, and the class attribute *Flu*, denoted by $T$, $H$, and $F$. From (3) we can calculate $\mu_{\{T,H\}}(F) = 1$, $\mu_T(F) = 5/8$ and $\mu_H(F) = 0$, i.e. according to this measure, *Flu* totally depends on $\{Temperature, Headache\}$, partially depends on $Temperature$ and is independent of $Headache$.

**Table 1** Information table

| | *Temperature (T)* | *Headache (H)* | *Flu (F)* |
|---|---|---|---|
| $e_1$ | *normal* | *yes* | *no* |
| $e_2$ | *high* | *yes* | *yes* |
| $e_3$ | *very_high* | *yes* | *yes* |
| $e_4$ | *normal* | *no* | *no* |
| $e_5$ | *high* | *no* | *no* |
| $e_6$ | *very_high* | *no* | *yes* |
| $e_7$ | *high* | *no* | *no* |
| $e_8$ | *very_high* | *yes* | *yes* |

An interpretation of (3) can be obtained by expressing the causal relation between attributes in the form of rules. For example, in considering how the attribute *Flu* depends on the attribute *Temperature* we can verify that

If *Temperature* = *normal* then *Flu* = *no*
If *Temperature* = *very_high* then *Flu* = *yes*

The number of objects that satisfy these rules is 5 out of 8. In other words, the proportion of objects whose values on *Flu* are correctly predicted by values on *Temperature* is 5/8. This argument is analogous with the definition of the degree of dependency, where each rule corresponds to an equivalent class w.r.t. $P$ which is included in an equivalent class w.r.t. $Q$.

### 2・2 *Decision tree induction*

Decision tree induction (DTI) is a widely used technique in supervised inductive learning and knowledge discovery. From a given set of labelled instances, a DTI system induces a classifier in the form of a decision tree that correctly predicts classes of unknown instances. The common framework for most univariate DTI methods can be briefly described in the following steps.

*S1. Select the "best" attribute by a selection measure*
*S2. Extend tree by adding a new branch for each attribute value*
*S3. Sort training examples to leaf nodes*
*S4. If examples unambiguously classified Then Stop Else Repeat steps 1-4 for leaf nodes*

The attribute selection is carried out by measures that decide which attribute will be selected to branch the tree. Often, the discretization and pruning are done as pre-processing and post-processing steps of the DTI.

## 3. Formation of R-measure

The limitation of the deterministic model of rough set theory when dealing with uncertain information has been recognized and studied, e.g., the probabilistic model [Pawlak 88], the variable precision model [Ziarko 93].

The variable precision model extends rough sets by employing relations named majority inclusion relations. A majority inclusion relation considers a set $A$ is included in a set $B$ if the intersection is a majority of set $A$ w.r.t a threshold. Based on such a relation the model redefines all the notions of rough sets. Although those redefinitions aimed at better handling uncertain and noisy data, they also raised a new problem of specifying appropriate thresholds in a particular application.

In contrast to the variable precision model, the probabilistic model requires no threshold. However, on the one hand, the definitions of the basic notions (e.g. the upper, lower approximations and the boundary) are totally consistent with Bayes' decision procedure. On the other hand, the definitions of the derived notions (e.g. attribute dependency, reduct, core) are based exclusively on information theory. This mixing approach makes the model somehow incoherent and does not directly inherit all useful properties of the original model.

We propose alternative definitions of the derived notions for the probabilistic model that are consistent with Bayes' decision procedure. In short, these variants aim at (1) overcoming the limitations of the original model for noisy data, (2) making the probabilistic model more coherent, and (3) preserving the convenience without requiring any threshold. We describe here only our modifications to $\mu_P(Q)$. Other derived notions of rough sets (attribute significance, reduct, core, superfluous) are definitely based on this key notion, and can be defined accordingly. Return to the Table 1, we can obtain the following probabilistic rules about the relation between *Flu* and *Headache*

*If Headache = yes* then *Flu = yes* (3/4)
*If Headache = no* then *Flu = no* (3/4)

These rules show that *Flu* somehow depends on *Headache*, but formula (3), by its value 0 in this case, says that *Flu* is independent of *Headache*. Let us consider further probabilistic rules. Suppose that the value on *Headache* of a new object is known, and an agent wants to predict the value on *Flu* of this object. For example, if *Headache = yes*, then there are two possibilities: *Flu = yes* (3/4), or *Flu = no* (1/4). To minimize the probability of error, *Flu = yes* is certainly chosen as it is the value with the maximum likelihood of occurrence among all possibilities. Due to the risk of *Flu = no*, this prediction is uncertain and has an estimated accuracy of 3/4. Similarly, the value *Flu = no* will be predicted if *Headache = no* with the estimated accuracy is also 3/4. Denote by $X$ the event that the prediction of the agent is true, we have

$$\begin{aligned} \mathrm{P(X)} &= \mathrm{P}(H = yes) \times \mathrm{P}(\mathrm{X} \mid H = yes) + \\ &\quad \mathrm{P}(H = no) \times \mathrm{P}(\mathrm{X} \mid H = no) \\ &= 1/2 \times 3/4 + 1/2 \times 3/4 = 3/4 \end{aligned}$$

This value can be interpreted as the degree of dependency of *Flu* on *Headache* established by the above argument. This argument can be generalized and formulated for a measure of degree of dependency of an attribute set $Q$ on an attribute set $P$

$$\mu'_P(Q) = \frac{1}{|O|} \sum_{[o]_P} max_{[o]_Q} |[o]_Q \bigcap [o]_P| \quad (4)$$

The degree of dependency *Flu* on *Temperature* calculated by (4) is 3/4.

A property of $\mu'_P(Q)$ has been proved but the proof is skipped here due to the length.

**[Theorem]** *For every sets $P$ and $Q$ of attributes we have*

$$\frac{max_{[o]_Q}|[o]_Q|}{|O|} \leq \mu_P^{'}(Q) \leq 1 \qquad (5)$$

We can define that $Q$ totally depends on $P$ iff $\mu_P^{'}(Q) = 1$; $Q$ partially depends on $P$ iff $max_{[o]_Q}|[o]_Q|/|O| < \mu_P^{'}(Q) < 1$; $Q$ is independent of $P$ iff $\mu_P^{'}(Q) = max_{[o]_Q}|[o]_Q|/|O|$.

Suppose that we are dealing with a problem of learning a classifier with $K$ classes $C_1, C_2, ..., C_K$ from a set of training instances $O$ described by a set of attributes. We assume that all attributes are discrete, each of which is with a finite number of possible values. Let $n_{..}$ denotes the total number of training instances, $n_{i.}$ the number of instances from class $C_i$, $n_{.j}$ the number of instances with the $j$-th value of the given attribute $A$, and $n_{ij}$ the number of instances from class $C_i$ and with the $j$-th value of $A$. Let further

$$p_{ij} = \frac{n_{ij}}{n_{..}}, \quad p_{i.} = \frac{n_{i.}}{n_{..}}, \quad p_{.j} = \frac{n_{.j}}{n_{..}}, \quad p_{i|j} = \frac{n_{ij}}{n_{.j}}$$

denote the approximation of the probabilities from the training set. In (4), if we consider $P$ a descriptive attribute and $Q$ the class attribute, we can rewrite $\mu_P^{'}(Q)$ in the form

$$\mu^{'} = \sum_j p_{.j} max_i p_{i|j} \qquad (6)$$

As this formula describes how much the class attribute depends on a descriptive attribute, we can naturally consider it as a candidate for a new attribute selection measure. However, despite the fact that it shows good results in some datasets, the results become unstable when the dimension of data increases. The fact is that the measure is too greedy in finding "best" attributes for the front step while tree growing is a multistep procedure. An analysis based on the notion of impurity function gives us a clearer view of this phenomenon, and provides a basis to go from $\mu^{'}$ to $\tilde{\mu}$ (R-measure).

Let $O$ be a set of objects each of them belongs to one of the classes $C_1, C_2, ..., C_K$, and vector $PC = \langle q_1, q_2, ..., q_K \rangle$ is the *class probability vector* where each component $q_i$ is the proportion of $i$-class objects [Fayyad 92].

**[Definition]** *An impurity function is a function $\phi$ defined on PC with the properties*

*(i) $\phi$ is maximum only at the point (1/K, 1/K,..., 1/K);*

*(ii) $\phi$ is minimum only at the point (1, 0, ..., 0), (0, 1, ..., 0), ..., (0, ..., 0, 1);*

*(iii) $\phi$ is a symmetric function of $q_1, ..., q_K$.*

Given an impurity function $\phi$, we consider the impurity of the partition of $O$ after splitting it by an attribute $A$ as described by step S1 in the DTI framework. Before splitting, we define the impurity value of $O$ as $I(O) = \phi(p_{1.}, ..., p_{K.})$, and after splitting we define the impurity value of the partition as

$$IP(O, A) = \sum_j p_{.j} \phi(p_{1|j}, ..., p_{K|j}) \qquad (7)$$

We wish to select an attribute $A$ that most reduces the partition impurity after splitting or, equivalently, that maximizes

$$R(O, A) = I(O) - IP(O, A) \qquad (8)$$

or, equivalently, that maximizes the purity of the partition which is defined as

$$P(O, A) = \sum_j p_{.j}(1 - \phi(p_{1|j}, ..., p_{K|j})) \qquad (9)$$

$P(O, A)$ can be considered as an attribute selection measure. In the case of $\mu^{'}$ the corresponding impurity function can be defined as

$$\phi(p_1, ..., p_K) = 1 - max_j p_j \qquad (10)$$

As shown in [Breiman 84], DTI needs another requirement for impurity function, otherwise the corresponding attribute selection measure will have the defects of degeneracy and not adequate for the overall multistep tree growing procedure

*(iv)* $\frac{\partial^2 \phi}{\partial q_i^2} < 0, \quad 0 < q_i < 1.$

Furthermore, we prefer that $\frac{\partial^2 \phi}{\partial p_i^2}$ is not only negative but also a constant as it will make $\frac{\partial \phi}{\partial q_i}$ not only decrease as $q_i$ increases but also decrease linearly. It leads to a significant modification in our impurity function

$$\phi(p_1, ..., p_K) = 1 - (max_j p_j)^2 \qquad (11)$$

and the corresponding attribute selection measure will be

$$\tilde{\mu} = \sum_j p_{.j} (max_i p_{i|j})^2 \qquad (12)$$

We call $\tilde{\mu}$ in (12) *R-measure* and it has the general form for arbitrary attribute sets $P$ and $Q$

$$\tilde{\mu}_P(Q) = \frac{1}{|O|} \sum_{[o]_P} max_{[o]_Q} \frac{|[o]_Q \bigcap [o]_P|^2}{|[o]_P|} (13)$$

## 4. Model Selection and Evaluation of R-measure

The three common criteria for evaluating DTI methods/models are size, accuracy and understandability of induced trees [Minger 89a]. Understandability is difficult to quantify as it is based on the judgments of domain experts or users. Tree size and accuracy can be quantitatively evaluated, and they have been used in the model selection in different ways. In [Kobayashi 96], the authors used a genetic algorithm for generating Pareto optimal decision trees by tradeoffs between accuracy and simplicity (tree size). Most work focused on statistical evaluations of the tree size and accuracy where the accuracy is widely considered to be of primary importance. However, as machine learning has no curriculum as other sciences do, empirical evaluations of learning systems have not always been done well. In [Salzberg 97] the authors warned that if not done carefully, comparative studies of classification algorithms can easily result in statistically invalid conclusions. Recently, *k-fold stratified cross validation* (typically with $k = 10$), has been recommended as a reliable method for real-world datasets similar to those of the UCI repository of machine learning databases.

To support doing model selection we have developed an interactive tool in system CABRO, based on a $k$-fold stratified cross validation, that allows the user to generate different combinations of available methods, to test and estimate which one will offer the best performance in an interactive and visual manner. Generally, independent methods for solving three DTI problems of attribute selection, pruning, and discretization can be combined to form different DTI models. In the current version of CABRO several well known methods have been implemented and other methods can be gradually integrated. For attribute selection we choose the gain-ratio [Quinlan 93], gini-index [Breiman 84], $\chi^2$ [Minger 89a] and R-measure. For pruning we choose the error-complexity [Breiman 84], reduced-error and pessimistic error [Quinlan 93], and for discretization we choose the entropy-based and error-based discretization [Fayyad 92]. Figure 1 illustrates a screen of CABRO and shows how the user can interactively do the model selection. The Model Selection window (upper-left) permits to select and combine methods (by default Measure = R-measure; Pruning = Error-complexity; and Discretization = Entropy-based). After selecting a model, the user can induce the tree with and/or without interaction (subsection 5.2). The induced tree will be displayed in the Tree Visualizer window (subsection 5.1). By clicking each node in the induced tree we can see its information in the Decision/Leaf Node Information window (lower-left). Given a dataset (application task), CABRO first carries out automatically a random shuffle of the dataset then divides the dataset into $k$ mutually exclusive subsets (folds) of approximately equal sizes and the same proportions of labels as in the original dataset.

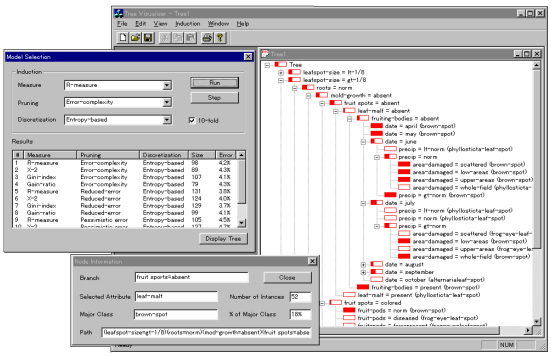

**Fig. 1** Model Selection in CABRO

To estimate the performance of a selected model, CABRO carries out automatically the $k$-fold cross validation by doing $k$ times the procedure: running the corresponding program with training data (the union of $k-1$ folds) for inducing a decision tree, and estimating the model accuracy with testing data (the rest fold). CABRO takes the final estimation of the model accuracy and tree size as the average of those obtained from $k$ runs. When there are different selected models, CABRO carries out the $k$-fold cross validation for each model with the same division of the original dataset into $k$ folds.

**Table 2** Experimental resutls

| data sets | dimension | type | gain-ratio | | gini-index | | $\chi^2$ | | R-measure | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | size | error | size | error | size | error | size | error |
| Vote | 16x300 | sym | 4.0±0.0 | **5.0±2.8** | 7.0±4.2 | 5.9±2.7 | 7.0±4.2 | 5.9±2.7 | 5.8±2.9 | 5.7±2.7 |
| Cancer | 9x700 | sym | 46.1±19.1 | 7.4±2.9 | 36.2±11.9 | 7.4±3.5 | 36.2±11.9 | 7.4±3.5 | 37.3±11.2 | **7.1±3.4** |
| Shuttle | 9x956 | sym | 53.4±15.8 | **0.2±0.1** | 114.2 ±12.2 | **0.2±0.1** | 162.7±30.4 | 0.3±0.1 | 135.3±18.3 | 0.3±0.1 |
| Promoters | 45x105 | sym | 9.8±4.3 | 24.5±7.5 | 9.4±3.7 | **22.7±10.0** | 9.4±3.7 | **22.7±10.0** | 9.4±3.7 | **22.7±10.0** |
| Solar Flare | 12x1286 | sym | 26.8±7.8 | **25.3±1.5** | 54.4±15.0 | 27.8±1.3 | 45.8±18.1 | 26.6±2.0 | 44.6±31.1 | 25.5±1.0 |
| Diabetes | 8x768 | num | 18.2±9.4 | **25.3±2.6** | 22.0±4.8 | 25.6±2.5 | 13.6±6.2 | 25.5±2.5 | 27.8±19.8 | **25.3±2.6** |
| Splice | 45x3189 | num | 17.3±5.5 | **34.5±8.2** | 22.3±6.9 | 36.8±6.8 | 19.1±7.5 | 37.3±6.4 | 18.7±6.8 | 35.9±6.9 |
| Waveform | 36x3195 | sym | 223.9±72.9 | 25.7±1.1 | 244.5±69.5 | **24.4±1.6** | 340.6±191.6 | 26.8±1.3 | 249.6±78.4 | 25.1±1.1 |
| Heart Disease | 13x270 | mix | 8.8±3.8 | 25.6±4.1 | 25.8±2.6 | 25.6±5.6 | 9.0±3.2 | 26.3±4.9 | 8.2±5.1 | **25.2±4.6** |
| Vehicle | 18x846 | num | 131.9±40.7 | 32.7±5.1 | 111.4±37.8 | 32.0±3.7 | 111.4±47.4 | 31.9±3.2 | 18.2±4.9 | **0.9±0.4** |
| Hypothyroid | 25x3163 | num | 131.9±40.7 | 32.7±5.1 | 111.4±37.8 | 32.0±3.7 | 111.4±47.4 | 31.9±3.2 | 18.2±4.9 | **0.9±0.4** |
| Audiology | 70x226 | sym | 28.4±13.3 | 30.9±11.0 | 37.0±16.0 | 30.9±11.9 | 66.9±14.9 | 45.2±8.7 | 41.3±13.4 | **29.1±11.7** |
| Cars | 8x392 | num | 17.1±9.5 | 26.0±2.0 | 21.4±8.5 | 26.8±5.2 | 17.4±9.1 | 26.5±5.2 | 21.8±12.8 | **25.2±4.8** |
| Horse-colic | 28x368 | num | 8.2±4.1 | **14.3±5.1** | 30.9±20.3 | 16.8±3.5 | 18.5±9.0 | 23.5±3.5 | 30.6±17.4 | 23.9±3.2 |
| Pima-diabetes | 8x768 | num | 17.6±5.8 | **23.4±3.6** | 25.4±8.3 | 23.5±3.5 | 18.5±9.0 | 23.5±3.5 | 30.6±17.4 | 23.9±3.2 |
| Segmentation | 19x2310 | num | 236.4±46.5 | 6.2±1.6 | 257.5±81.2 | **6.1±2.0** | 310.7±48.3 | 7.6±2.0 | 272.0±90.4 | **6.1±2.1** |
| Iris | 4x150 | num | 4.0±0.0 | 3.3±3.3 | 4.0±0.0 | **2.7±3.2** | 4.0±0.0 | 4.0±4.0 | 4.0±0.0 | 4.0±4.0 |

In Model Selection window are displayed the selected models, the size and error rate of induced trees. Two main advantages of this tool are (1) it makes the model evaluation much easier and avoids errors caused by manual calculations; (2) it offers a good solution for model selection if meta-knowledge is not available, and if meta-knowledge is available it can be mutually used with experimental results in order to select more reliable models.

R-measure is evaluated by using this tool as follows. We compare experimentally four measures gain-ratio, gini-index, $\chi^2$ and R-measure using 18 datasets from the UCI repository of machine learning databases. Four DTI models are formed using these four measures with the same methods of pruning (error-complexity) and discretization (entropy-based). Table 2 presents experimental results on size and error rate (pruned trees) of four measures estimated with the 90% confidence interval.

Some observations and conclusions can be drawn from these results.

(1) For the error rates, the lowest values (bold numbers) are attained by both the gain-ratio and R-measure at 9 out of 18 datasets, by the gini-index and $\chi^2$ at 5 and 2 out of 18 datasets, respectively. We notice that while on a majority of datasets the error rates of different measures are significantly different, on some datasets all or almost the measures attained the same error rates. If we consider the fact that $\chi^2$ attained no unique lowest value and that it had comparatively high values in general, we can say that this measure showed a poor performance in our evaluation. The gini-index showed to be better as it attained the lowest error rates 5 times and the middle values on almost other datasets. However, it is worth noting that it attained only 1 unique lowest value, compared to 5 lowest values attained by both the gain-ratio and R-measure. Additionally, if we compare only the gini-index and R-measure, we can see the ratio of lower error rate is 5 versus 10 among 15 datasets on those the two measures attained different values. In this evaluation, the gain-ratio and R-measure showed equally low error rates as both attained 9 lowest and 5 unique lowest values. Our evaluation again confirms the fact that there are significant differences between the attribute selection measures and also there is no absolute superior measure.

(2) For the tree size, the gain-ratio is the only measure showed a significant advantage due to the fact that it is designed with the bias of small trees. However, in practice the differences between tree sizes are not very important when the trees are not very large. From the larger datasets Spice, Waveform, and Segmentation we had relatively big trees, the gain-ratio did not show any significant advantage. The gain-ratio showed its advantage of smaller tree when trees are of small or middle-size, but it do not have this advantage when trees become big.

(3) R-measure showed good results and this makes us believe that R-measure is considering as a good alternative measure for DTI attribute selection.

## 5. An interactive-graphic system

To support the understanding and learning decision trees, CABRO has been implemented in a Visual Interactive Model through a rich graphical environment. A Visual Interactive Model (VIM) aims at combining meaningful pictures and easy interactions to stimulate creativity and insight; promoting a process of 'generate and test', it facilitates a rapid cycle of learning. In CABRO, the VIM offers: (1) Tree Visualizer that supports the understanding and analysing decision trees; (2) the Interactive Learning Mode that supports user participation in the model selection and the learning process. CABRO has been implemented in platforms of MS Windows 95 and UNIX workstations under the X Window.

CABRO allows the user to load a dataset, to select DTI models, to induce decision trees, to manipulate generated trees, to test and match decision trees with unknown cases, among other features.

### *5·1 Tree Visualizer*

The *Tree Visualizer* is a graphical interface that displays a tree in graphical form as a set of nodes and connections. Its main contribution is *the transformation of a decision tree from a static form (text) to a dynamic form (graphic) on which a certain number of operations can be done.* The user can dynamically navigate through the tree and has different views on generated trees, switch among several view modes, choosing alternate parts of the tree or focus on the paths to one class.

- *Viewing the tree structure*: The tree can be collapsed or expanded fully from the root or from any decision node. The user can exploit different multiple views on parts of the tree, such as to view a subtree from a decision node, to collapse some nodes and/or expand some other ones, etc.
- *Viewing decision/leaf nodes*: The user can click on a node to see its information (Figure 1, Decision/Leaf Node Information window (lower-left)): branching attribute and branched attribute-value, number of covered cases, the major class, the percentage of major class, the path leading from the root to the decision/leaf node, etc.
- *Viewing a class on the tree*: The user can focus on observing decision paths to leaf nodes of a class. These leaf nodes are highlighted and proportions of cases bearing this class label are indicated approximately in decision nodes.

### *5·2 Interactive Learning*

The *Interactive Learning Mode* allows the user to participate actively in the DTI process, mainly the generating-and-test process in connection to the Model Selection and Tree Visualizer.

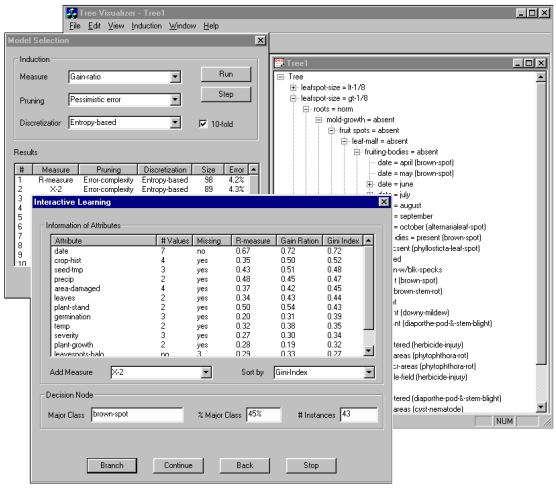


**Fig. 2** Interactive Learning in CABRO

The user can execute a full automatic learning of decision trees without intervention by clicking the Run button, or execute an interactive learning of decision trees by clicking the Step button, or combine these two modes. In interactive learning, CABRO offers the user the chance to participate in selecting attributes to branch decision nodes from evaluated attributes. When choosing the interactive learning mode from a decision node, the user will see the table listing candidate attributes and their corresponding values on every measures, sorted by the selected measure (Figure 2). Depending on the user selection of branching attribute (highlighting the attribute and clicking the Branch button), CABRO will split the decision node and display the corresponding tree, as well related information. The user can do backtracking to regrow the tree at some node

with respect to the induction scheme (Back button) or continue the learning process.

The interactive learning mode is particularly meaningful at some first levels of the decision tree where domain knowledge can play a significant role in generating a desirable tree. The user can switch from the interactive learning mode to the automatic learning mode at any moment.

## 6. Conclusion

We have first presented R-measure for the attribute selection in DTI inspired by rough set theory, then introduced a model selection tool. We have also described the interactive-graphic system CABRO with two main components: Tree Visualizer and Interactive Learning. The high performance of R-measure and the advantages of the interactive-graphic environment of CABRO offer some significant features of a DTI system.

We feel that the following issues are worth pursuing (1) to enrich the interactive-graphic system by implementing other pruning and discretization techniques, (2) to use R-measure to investigate other notions in rough set theory, and (3) to use CABRO in real applications.

***Acknowledgments***

The authors wish to thank Professors M. Kimura, M. Vlach, and H. Shimodaira at JAIST–Hokuriku, and anonymous reviewers for valuable comments and suggestions.

## ◇ References ◇

[Breiman 84] Breiman, L., Friedman, J., Olshen, R. and Stone, C.: *Classification and Regression Trees*, Belmont, CA: Wadsworth (1984).

[Dougherty 95] Dougherty, J., Kohavi, R. and Sahami, M.: Supervised and Unsupervised Discretization of Continuous Features, in *Proceedings 12th International Conference on Machine Learning*, Morgan Kaufmann, pp. 194–202 (1995).

[Fayyad 92] Fayyad, U.M. and Irani K.B.: On the Handling of Continuous-Valued Attributes in Decision Tree Generation, *Machine Learning*, pp. 87–102 (1992).

[Kobayashi 96] Kobayashi, S., Yoshida, K., and Yamamura, M.: Generating a Set of Pareto Optimal Decision Trees by Genetic Algorithms, *Journal of Japanese Society for Artificial Intelligence*, Vol. 11, No. 5, pp. 778–785 (1996).

[Minger 89a] Mingers, J.: An Empirical Comparison of Selection Measures for Decision Tree Induction, *Machine Learning*, 3, pp. 319–342 (1989).

[Minger 89b] Mingers, J.: An Empirical Comparison of Pruning Methods for Decision Tree Induction, *Machine Learning*, 4, pp. 227–243 (1989).

[Pawlak 88] Pawlak, Z., Wong, S.K.M., Ziarko, W.: Rough Sets, Probabilistic versus Deterministic Approach. *International Journal of Man-Machine Studies*, 29, pp. 81–95 (1988).

[Pawlak 91] Pawlak, Z.: *Rough sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic Publishers (1991).

[Quinlan 93] Quinlan, J.R.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann (1993).

[Salzberg 97] Salzberg, S.L.: On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach, *Data Mining and Knowledge Discovery*, 1, 317–327 (1997).

[Ziarko 93] Ziarko, W.: Variable Precision Rough Set Model. *Journal of Computer and System Sciences*, 46, pp. 39–59 (1993).

担当編集委員 ：××○○，査読者 ：○○××〕

## Authors' Profile

**Trong Dung Nguyen**

received a B.A. degreee from Hanoi University in 1990 in computer science and Master's degree from JAIST–Hokuriku in 1997. He is currently undertaking a PhD on knowledge discovery. 〈nguyen@jaist.ac.jp〉

**Tu Bao Ho** (Member)

received a B.Eng. degree from Hanoi University of Technology in 1978, M.S. and Ph.D. degrees from University Paris 6, in 1984 and 1987. He has been with JAIST–Hokuriku as a visiting associate professor at School of Information Science since July 1993, and as a professor at School of Knowledge Science since April 1998. His research interests include machine learning, knowledge-based systems, and knowledge discovery in databases. 〈bao@jaist.ac.jp〉