Fusing Monotonic Decision Trees

Yuhua Qian, *Member, IEEE*, Hang Xu, Jiye Liang, Bing Liu, *Fellow, IEEE*, and Jieting Wang

Abstract—Ordinal classification with a monotonicity constraint is a kind of classification tasks, in which the objects with better attribute values should not be assigned to a worse decision class. Several learning algorithms have been proposed to handle this kind of tasks in recent years. The rank entropy-based monotonic decision tree is very representative thanks to its better robustness and generalization. Ensemble learning is an effective strategy to significantly improve the generalization ability of machine learning systems. The objective of this work is to develop a method of fusing monotonic decision trees. In order to achieve this goal, we take two factors into account: attribute reduction and fusing principle. Through introducing variable dominance rough sets, we firstly propose an attribute reduction approach with rank-preservation for learning base classifiers, which can effectively avoid overfitting and improve classification performance. Then, we establish a fusing principe based on maximal probability through combining the base classifiers, which is used to further improve generalization ability of the learning system. The experimental analysis shows that the proposed fusing method can significantly improve classification performance of the learning system constructed by monotonic decision trees.

Index Terms—Monotonic classification, rough sets, attribute reduction, decision tree, ensemble learning

1 Introduction

Classification model is one of important research issues in machine learning and data mining. A classification task is to learn a classifier from a given trained data set with class labels, which can be used to predict the categories of unlabeled objects. From the viewpoint of constraints among attribute values, classification tasks can be regarded as two types: nominal classification and ordinal classification. Unlike no ordinal structure among different decision values, for an ordinal classification task, the ordinal relationship between different class labels should be taken into account [40], [45]. Monotonic classification is a class of special ordinal classification tasks, where the decision values are ordinal and discrete, and there are a monotonic constraint between attributes and decision classes [32]. A monotonic constraint indicates that the objects with better attribute values should not be assigned to a worse decision class [19]. Monotonic classification is a kind of common tasks, which have attracted increasing attention from domains of data mining, knowledge discovery, pattern recognition, intelligent decision making, and so on.

For a monotonic classification task, from a given training set of objects with a monotonic constraint, its objective is to learn and extract some decision rules for understanding decisions and building an automatic decision model. To address this issue, several relative researches have been reported. These existing works on monotonic classification can be roughly divided into two groups. One is to develop a theoretic framework for monotonic classification, such as the dominance rough set model [14], [15], [16], [21], [24], [37], [38], [42], the qualitative decision theory [10] and the ordinal entropy model [19], [20], and the other is to construct algorithms for learning monotonic decision models from objects [1], [2], [4], [11].

As one of attempts solving monotonic classification, Greco et al. [14], [15], [16] proposed a dominance rough set through introducing dominance relations into rough sets. Rough sets have been proven to be an effective classification method, which can be used to extract some decision rules and construct a rule-based classifier [8], [22], [35], [36], [47], [49]. Unlike other models of rough sets, the model of dominance rough sets is used to extract ordinal decision rules for monotonic classification. Since then, several researches have been reported to generalize or employ this model in monotonic classification. Shao and Zhang [42] extended the dominance rough set to adapt the context of data sets with missing data. Qian et al. [37], [38] addressed versions of dominance rough sets in set-valued ordered information systems and interval ordered information systems. Hu et al. [21] introduced a fuzzy preference into rough sets for monotonic classification with a fuzzy consistent constraint. As the literature [16] reported, dominance rough sets often produce much larger classification boundary on some real-world tasks, which make the decision algorithm constructed by dominance rough sets as no or few consistent rules could be extracted from data.

As to monotonic classification algorithms, some effective results have been reported. Ben-David extended the classical decision tree algorithm to monotonic classification in 1995. Since then, a collection of decision tree algorithms have been developed for this problem [6], [9], [12], [23], [33]. In addition, Ben-David [3] also extended the nearest neighbor classifier to monotonic tasks and designed an ordinal learning model (OLM). In 2003, Cao-Van and Baets [6] introduced ordinal stochastic dominance learner (OSDL) based on associated cumulative distribution. In 2008, Lievens et al. [28] presented a probabilistic framework that served as the

- *Y. H. Qian, H. Xu, J. Y. Liang, and J. T. Wang are with the School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi Province, China. E-mail: {jinchengqyh, xuh102}@126.com, ljy@sxu.edu.cn, jietingw@163.com.*
- *B. Liu is with the Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607. E-mail: liub@cs.uic.edu.*

Manuscript received 3 Apr. 2014; revised 2 Mar. 2015; accepted 11 Mar. 2015. Date of publication 3 May 2015; date of current version 8 Sept. 2015. Recommended for acceptance by J. Bailey. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TKDE.2015.2429133

base of instance-based algorithms to solve the supervised ranking problems. In addition, in 2008, Duivesteijn and Feelders [11] proposed a modified nearest neighbor algorithm for the construction of monotone classifiers from data by monotonizing training data. Xia et al. [46] extended the Gini impurity used in CART to ordinal classification, and called it ranking impurity. Although the mentioned algorithms above improve the performance of extracting ordinal information, they can not ensure a monotonic decision tree is learned from a training data set with a monotonic constraint.

For classification tasks, in fact, we also need to consider robustness of a classification algorithm and sensitivity for noisy data. As we know, noise has great influence on modeling monotonic classification tasks [5]. If the measures used to evaluate quality of attributes in monotonic classification are sensitive to noisy objects, the performance of the trained classifier would be weak. An effective and robust measure of attribute quality is required for monotonic classification. To reduce the influence of noisy data and obtain decision rules with clear semantics, Hu et al. [19] designed a robust and understandable algorithm (a rank entropy based monotonic decision tree, just REMT) for monotonic classification. The theoretic and experimental analysis showed that the REMT algorithm can get monotonically consistent decision rules if objects in a training set are monotonically consistent and its performance is also good when data are contaminated with noise.

It is well known that ensemble learning can great improve the generalization performance of a learning system. Ensemble learning refers to first training a set of base classifiers from data sets and then fusing these classifiers with a fusion strategy for a given classification task or regression task [50], [51]. In fact, fusing a set of the same base classifiers will not yield any enhancement. The improvement comes from the diversity among these base classifiers, which is because that different base classifiers potentially offer complementary information about the objects to be classified. It was reported that the base classifiers should take both accurate and diverse into account together for constructing a good ensemble system [13], [26], [43].

Considering two merits of effectiveness and robustness of REMT algorithm, the objective of this study is to develop a fusing method of monotonic decision trees induced by the REMT algorithm for further enhancing the generalization performance of a monotonic classification system. As we know, attribute reduction plays an important role in improving classification performance and speeding up training [17], [29]. Based on this consideration, we first propose an attribute reduction method with rank-preservation property based on the variable dominance rough sets, which is used to generate some monotonic attribute reducts and learn base classifiers depicted by monotonic decision trees. The size of an monotonic attribute reduct is usually much shorter than that of the original attribute set, and the corresponding monotonic decision tree induced by it may have much better generalization ability. Through adjusting values of the parameter β in the variable dominance rough set, various monotonic attribute reducts from the original data set can be obtained, which are used to learn different base classifiers. This satisfies the diversity among base classifiers in ensemble learning. Then, we propose a fusing principe for combining the base classifiers based on the idea of maximal probability, which is used to further improve generalization ability of the monotonic classification system. The results show the effectiveness of the proposed method from two viewpoints of the classification accuracy and the mean absolute error. The contributions of this work is two folds. One is to propose an attribute reduction method for a monotonic classification task. The other is to develop a fusing strategy for fusing monotonic decision trees induced by the REMT algorithm. These two folds all can improve the performance of a monotonic classification system constructed by monotonic decision trees.

The rest of the paper is organized as follows. The preliminaries on dominance rough sets and monotonic decision trees are introduced in Section 2. In Section 3, we propose an attribute reduction method for monotonic classification and discuss some of its properties. In Section 4, we first give the algorithm of how to generate multiple monotonic attribute reducts (be used to learn multiple base classifiers), and then develop a fusing principle based on maximal probability (FPBMP) to combine the base decision trees. Section 5 gives a series of experimental analyses for showing the performance of the proposed method in this paper. Finally, Section 6 concludes this paper with some remarks and discussions.

2 Preliminaries on Dominance Rough Sets and Monotonic Decision Trees

Let $U = \{x_1, x_2, \ldots, x_n\}$ be a set of objects, A a set of attributes to describe the objects, d is a decision attribute, and D a finite ordinal set of decisions. With every attribute $a \in A$, a set of its values V_a is associated. The value of x_i under an attribute $a \in A$ or d is denoted by $v(x_i, a)$ or $v(x_i, d)$, respectively. The ordinal relation between objects in terms of the attribute a or d is denoted by $\succeq$, and $x \succeq_a y$ or $x \succeq_d y$ means that x is at least as good as (outranks) y with respect to the attribute a or d respectively. In the following, without any loss of generality, we consider a condition attribute having a numerical domain, that is, $V_a \subseteq \mathbf{R}$ ($\mathbf{R}$ denotes the set of real numbers) and being of type gain, that is, $x \succeq_a y \Leftrightarrow v(x,a) \geqslant v(y,a)$ (according to increasing preference) or $x \succeq_a y \Leftrightarrow v(x,a) \leqslant v(y,a)$ (according to decreasing preference), where $a \in A$, $x, y \in U$. For a subset of attributes $B \subseteq A$, we define $x \succeq_B y \Leftrightarrow \forall a \in B, v(x, a) \geqslant v(y,a)$. In other words, x is at least as good as y with respect to all attributes in B.

In a given set of objects, we say that x dominates y with respect to $B \subseteq A$ if $x \succeq_B y$, and denoted by $xR_B^{\geqslant} y$. That is

$$R_B^{\geqslant} = \{(x, y) \in U \times U \mid x \succeq_B y\}.$$

Obviously, if $(x, y) \in R_B^{\geqslant}$, then x dominates y with respect to B. A predicting rule is a function

$$f : U \to D,$$

which assigns a class label in D to each object in U. A monotonically ordinal classification function should satisfy the following constraint:

$$x \succeq y \Rightarrow f(x) \succeq f(y), \ \forall x, y \in U.$$

Definition 1. *Let $DT = (U, A \cup \{d\})$ be a decision table, $B \subseteq A$. If $\forall x, y \in U$, $x \succeq_B y$, then $x \succeq_d y$, we say DT is B-monotonically consistent.*

Dominance rough set is an effective method to deal with monotonic classification, which can extract a family of ordinal decision rules from a given ordinal data set. In the following, we review several notations to be used throughout this paper.

Let $DT = (U, A \cup \{d\})$ be a decision table, $B \subseteq A$, $B = B_1 \cup B_2$, where B_1 be the attribute set according to increasing preference, and B_2 the attribute set according to decreasing preference. The granules of knowledge [31], [39], [48] induced by the dominance relation $R_B^{\geqslant}$ are the set of objects dominating x, i.e.,

$$[x]_B^{\geqslant} = \{y \in U \mid v(y, a_1) \geqslant v(x, a_1)(\forall a_1 \in B_1) \text{ and } v(y, a_2) \leqslant v(x, a_2)(\forall a_2 \in B_2)\} = \{y \in U \mid (y, x) \in R_B^{\geqslant}\}$$

and the set of objects dominated by x,

$$[x]_B^{\leqslant} = \{y \in U \mid v(y, a_1) \leqslant v(x, a_1)(\forall a_1 \in B_1) \text{ and } v(y, a_2) \geqslant v(x, a_2)(\forall a_2 \in B_2)\} = \{y \in U \mid (x, y) \in R_B^{\geqslant}\},$$

which are called the B-dominating set and the B-dominated set with respect to $x \in U$, respectively.

For simplicity and without any loss of generality, in the following we only consider condition attributes with an increasing preference.

The following property can be easily concluded [14], [15], [37], [38].

Property 1. *Let $R_B^{\geqslant}$ be a dominance relation, then*

1) *$R_B^{\geqslant}$ is reflexive, transitive and unsymmetric, so it is not an equivalence relation;*
2) *if $C \subseteq B \subseteq A$, then $R_A^{\geqslant} \subseteq R_B^{\geqslant} \subseteq R_C^{\geqslant}$;*
3) *if $C \subseteq B \subseteq A$, then $[x]_A^{\geqslant} \subseteq [x]_B^{\geqslant} \subseteq [x]_C^{\geqslant}$;*
4) *if $x_j \in [x_i]_B^{\geqslant}$, then $[x_j]_B^{\geqslant} \subseteq [x_i]_B^{\geqslant}$ and $[x_i]_B^{\geqslant} = \bigcup \{[x_j]_B^{\geqslant} : x_j \in [x_i]_B^{\geqslant}\}$;*
5) *$[x_i]_B^{\geqslant} = [x_j]_B^{\geqslant}$ iff $v(x_i, a) = v(x_j, a)(\forall a \in B)$;*
6) *$F = \{[x]_B^{\geqslant} \mid x \in U\}$ constitutes a covering of U.*

For any $X \subseteq U$ and $B \subseteq A$, the lower and upper approximations of X with respect to the dominance relation $R_B^{\geqslant}$ are defined as follows:

$$\underline{R_B^{\geqslant}}(X) = \{x \in U \mid [x]_B^{\geqslant} \subseteq X\},$$
$$\overline{R_B^{\geqslant}}(X) = \{x \in U \mid [x]_B^{\geqslant} \cap X \neq \emptyset\}.$$

The model defined above is called a dominance rough set, introduced by the literature [14]. This model was widely discussed and applied in recent years [21], [37], [38], [42]. Let $d_i^{\geq}$ be a subset of objects whose decisions are equal to or better than d_i, then we say that each object of $\underline{R_B^{\geqslant}}(d_i^{\geq})$ is consistently equal to or better than d_i.

However, the decision boundary regions in some of real applications are often so large that an effective decision model can not be constructed by dominance rough sets, in which there exist too many inconsistent samples in a given data set. In addition, dominance rough sets are heavily sensitive to noisy samples, in which several mislabeled objects might completely change the trained decision models as Hu et al. pointed out in the literature [19]. To address this issue, Hu et al. proposed [19] a rank entropy-based decision tree for monotonic classification. This rank entropy has better robustness than Shannon's information entropy [41] for monotonic classification. In what follows, we review some of its relative concepts.

To characterize the ordinal structure in monotonic classification, Hu et al. [19] introduced a rank entropy method to measure the ordinal consistency between random variables, which includes the following four definitions.

Definition 2. *Given $DT = (U, A \cup \{d\})$, $B \subseteq A$. The rank entropy of the system with respect to B is defined as*

$$RH_B^{\geq}(U) = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{|[x_i]_B^{\geq}|}{n}. \tag{1}$$

Definition 3. *Given $DT = (U, A \cup \{d\})$, $B \subseteq A$, $C \subseteq A$. The rank joint entropy of the set U with respect to B and C is defined as*

$$RH_{B \cup C}^{\geq}(U) = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{|[x_i]_B^{\geq} \cap [x_i]_C^{\geq}|}{n}. \tag{2}$$

Definition 4. *Given $DT = (U, A \cup \{d\})$, $B \subseteq A$, $C \subseteq A$. If C is known, the rank conditional entropy of the set U with respect to B is defined as*

$$RH_{B|C}^{\geq}(U) = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{|[x_i]_B^{\geq} \cap [x_i]_C^{\geq}|}{|[x_i]_C^{\geq}|}. \tag{3}$$

Definition 5. *Let $DT = (U, A \cup \{d\})$ be a decision table, B an arbitrary attribute. The rank mutual information of the set U with respect to B and $\{d\}$ is defined as*

$$RMI^{\geq}(B, \{d\}) = -\frac{1}{n}\sum_{i=1}^{n} \log \frac{|[x_i]_B^{\geq}| \times |[x_i]_{\{d\}}^{\geq}|}{n \times |[x_i]_B^{\geq} \cap [x_i]_{\{d\}}^{\geq}|}. \tag{4}$$

Monotonic decision trees are a class of specific decision trees which assign dominating decision to the objects characterized by better feature values in the case of monotonic classification [4]. Through combining the above rank mutual information and the diagram of the classical decision tree, Hu et al. [19] proposed a rank entropy-based decision tree for monotonic classification, which is better than existing monotonic decision trees in most cases in terms of mean absolute error. The entropy-based decision tree is constructed by the following algorithm.

Regarding labeling rule L, for an unseen object, according to its attribute values, we can match a path from the root node of the monotonic decision tree to a certain leaf node. Through this leaf node, the label of the unseen object can be determined by the following rule [19]:

1) If all of samples in a leaf node belong to the same class, we give the class label of this leaf node to the object;
2) Otherwise, if the samples in the leaf node belong to multiple classes, there are the following two cases:
   2.1) If the number of classes in the leaf node is an odd number, we assign the median class to the object. For example, if the samples in an leaf node come from Classes 1, 2 and 3, respectively, we label the

unseen object with Class 2 if it belongs to this leaf node.

2.2) If the number of classes in the leaf node is an even number, there are two median classes in this leaf node. Then if the current node is a left branch of its parent node, we assign the worse class to this node; otherwise, we assign the better class to it.

Algorithm 1. Rank entropy based monotonic decision tree [19]

Input: criteria: attributes of objects;
decision: class labels of objects;
ε: stopping criterion;
Output: a monotonic decision tree T.
(1) generate the root node.
(2) if the number of objects is 1 or all objects are from the same class, the branch stops growing.
(3) otherwise,
for each attribute a_i,
for each $c_j \in V_{a_i}$,
divide objects into two subsets according to c_j
if $v(a_i, x) \leq c_j$, then put x into one subset,
else put x into the other subset.
denote a_i with respect to c_j by $a_i(c_j)$.
compute $RMI_{c_j} = RMI^{\geq}(\{a_i(c_j)\}, \{d\})$.
end j.
$c_j^* = \arg\max_j RMI_{c_j}$.
end i.
(4) select the best attribute a and the corresponding point: $c^* = \arg\max_i \max_j RMI^{\geq}(\{a_i(c_j)\}, \{d\})$.
(5) if $RMI^{\geq}(\{a\}, \{d\}) < \varepsilon$, then stop.
(6) build a new node and split objects with a and c^*.
(7) Recursively produce new splits according to the above procedure until stopping criterion is satisfied.
(8) end.

3 Attribute Reduction for Monotonic Classification

In machine learning, attribute reduction (feature selection) is an effective method for improving classification performance and avoiding overfitting [7], [20], [27], [34], [44]. For a monotonic classification task, monotonicity constraints between attributes and decisions should be taken into account. However, most of existing techniques are not able to discover and represent the ordinal structures in monotonic data sets. Hence, they can not be well applied for monotonic classification. In this section, we aim to develop an attribute reduction approach to monotonic classification, which will be used to train base monotonic decision trees in next section.

Attribute reduction aims to retain the discriminatory power of original features in rough set theory, which has been proven effective for improving the classification performance of a rough classifier. From this point of view, we want to develop the corresponding attribute reduction method. An ordinal attribute reduct should satisfy the same monotonic constraint as the original set of attributes, with which the rank among objects can be kept unchanged.

Let $DT = (U, A \cup \{d\})$ be an ordinal decision table, where d a decision attribute with an overall preference of objects. Denoted by

$$R^{\geqslant}_{\{d\}} = \{(x, y) : v(x, d) \geqslant v(y, d)\},$$

$R^{\geqslant}_{\{d\}}$ is a dominance relation determined by the decision attribute d. If $R^{\geqslant}_A \subseteq R^{\geqslant}_{\{d\}}$, then DT is called monotonic consistent; otherwise it is monotonic inconsistent. In the following, we give the formal definition of a monotonic attribute reduct.

Definition 6. *Let $DT = (U, A \cup \{d\})$ be an ordinal decision table and $B \subseteq A$. If $R^{\geqslant}_B \subseteq R^{\geqslant}_{\{d\}}$ and $R^{\geqslant}_C \not\subseteq R^{\geqslant}_{\{d\}}$ for any $C \subset B$, then we call B a monotonic attribute reduct of DT.*

We denote by $D^* = \{(x, y) : v(x, d) < v(y, d)\}$, and

$$Dis^*(x, y) = \begin{cases} \{a \in A : (x, y) \notin R^{\geqslant}_{\{a\}}\}, & (x, y) \notin D^*; \\ \emptyset, & (x, y) \in D^*. \end{cases}$$

$Dis^*(x, y)$ is called an ordinal discernibility set between x and y, and $\mathbf{Dis}^* = (Dis^*(x, y) : x, y \in U)$ is called an ordinal discernibility matrix for the decision table.

Similarly to the classical attribute reduction [30], [42], the discernibility matrix based can be employed for obtaining all of ordered attribute reducts from an ordinal decision table.

Although the above method can obtain all monotonic attribute reducts of an ordinal decision table, its time complexity is exponential, which can not be used to learn from large-scale data sets. To solve this problem, in what follows, we develop another ordinal attribute reduction with a heuristic strategy although a reduct obtained by it may be a pseudo monotonic attribute reduct [34].

We continue to use the framework of dominance rough sets in this part. We first introduce a variable parameter β to loosen the condition of a dominance rough set, such that the rough set is less sensitive to noisy objects. Based on this view, for a monotonic classification task, we give an updated version of a dominance rough set, in which a set to be approximated is an upward union $d_i^{\geq} = \bigcup_{j \leq i} D_j$, where $D_j \in U/\{d\} = \{D_1, D_2, \ldots, D_r\}$ that are ordered, that is, for all $i, j \leq r$ if $i \geq j$, then the objects from D_i are preferred to the objects from D_j.

Definition 7. *Given $DT = (U, A \cup \{d\})$ and $B \subseteq A$, d_i is a decision value of d. As to monotonic classification, the variable upward lower and upper approximations of $d_i^{\geq}$ are defined as*

$$\underline{R^{\geqslant}_B}(d_i^{\geq}) = \left\{ x \in U \mid \frac{|[x]^{\geqslant}_B \cap d_i^{\geq}|}{|[x]^{\geqslant}_B|} \geq 1 - \beta \right\},$$

$$\overline{R^{\geqslant}_B}(d_i^{\geq}) = \left\{ x \in U \mid \frac{|[x]^{\geqslant}_B \cap d_i^{\geq}|}{|[x]^{\geqslant}_B|} \geq \beta \right\},$$

where $0 \leq \beta \leq 0.5$.

The following region

$$BND^{\beta}_B(d_i^{\geq}) = \overline{R^{\geqslant}_B}(d_i^{\geq}) - \underline{R^{\geqslant}_B}(d_i^{\geq})$$

is called the upward boundary region of d_i in terms of attribute set B. The monotonic dependency of d with respect to B is formally defined as

$$\gamma_B^{\beta}(d) = \frac{|U - \bigcup_{i=1}^{t} BND_B^{\beta}(d_i^{\geq})|}{|U|}.$$

If the decision table DT is monotonically consistent in terms of B, then $BND_B^{\beta}(d_i^{\geq}) = \emptyset$.

Based on the above monotonic dependency, we can define a coefficient as the significance of attribute a in B relative to the decision attribute d. Given $DT = (U, A \cup \{d\})$, $\forall a \in B \subseteq A$, and d_i a decision value of d, the inner significance of a in B relative to d is formally defined by

$$Sig_{inner}^{\beta}(a, B, d) = \gamma_B^{\beta}(d) - \gamma_{B-\{a\}}^{\beta}(d). \quad (5)$$

This measure can be used to determine the core attributes of A. When $Sig_{inner}^{\beta}(a, B, d) > 0$, as the rough set area classical defined [34], we say a is a core attribute in this decision table. Accordingly, $\forall a \in A - B$, we define the outer significance of a with respect to B as

$$Sig_{outer}^{\beta}(a, B, d) = \gamma_{B\cup\{a\}}^{\beta}(d) - \gamma_B^{\beta}(d). \quad (6)$$

This measure is used to select an attribute in a forward attribute reduction.

Through using these two attribute significance measures, one can design a monotonic attribute reduction approach with a heuristic strategy as follows.

In this algorithm, through introducing a variable parameter β to loosen the condition of a monotonic attribute reduct, such that the searched monotonic attribute reduct has better robustness for noisy objects. In addition, due to the size of a monotonic attribute reduct is much shorter than that of the original attribute set, and hence the base monotonic decision tree induced by the monotonic attribute reduct will have much smaller length and much fewer nodes, which usually possesses much better generalization ability.

4 Fusing Monotonic Decision Trees

The studies have shown that ensemble learning can significantly improve the generalization ability of machine learning systems. In this section, we propose an ensemble strategy by fusing multiple different monotonic decision trees.

In ensemble learning, there are two basic issues [50], [51]: learning multiple classifiers and a fusing strategy, where the former aims to provide some different base classifiers, and the latter is to give an effective ensemble method for obtaining much better generalization performance. In general, diversity among the base classifiers is known to be an important factor for improving generalization performance in ensemble learning.

For fusing monotonic decision trees, we need to learn various base classifiers with much bigger diversity. To solve this problem, we select multiple attribute subsets from the original attribute set of a given data set, in which each attribute subset should be an attribute reduct that preserves the monotonic consistent of the original data set. The diversity in ensemble learning can be satisfied by the corresponding base monotonic decision trees induced by different monotonic attribute reducts.

To obtain multiple monotonic attribute reducts, through loosening the condition of maximal significance in each loop, we can continue to use Algorithm 2 with the second maximal significance. For a given parameter β in variable dominance rough sets, we can generate multiple monotonic attribute reducts from a given date set. Based on the idea of Algorithm 3 in the literature [18], the algorithm can be similarly depicted as follows.

Algorithm 2. Computing a monotonic attribute reduct with a forward searching strategy

Input: a decision table $DT = (U, A \cup \{d\})$ and the parameter β;
Output: a monotonic attribute reduct B of A.
(1) Compute core $Core$ of A using Sig_{inner}^{β} [34].
(2) $B \leftarrow Core$.
(3) $\forall a \in A - B$, compute $SIG_{outer}^{\beta}(a, B, d)$;
  if $SIG_{outer}^{\beta}(a_j, B, d) = \max_i\{SIG_{outer}^{\beta}(a_i, B, d)\}$,
    $B \leftarrow B \cup \{a_j\}$,
  until $\forall a_i$, $SIG_{outer}^{\beta}(a_i, B, d) = 0$.
(4) return B and end.

Algorithm 3. Backward reduction for searching multiple ordinal reducts

Input: a decision table $DT = (U, A \cup \{d\})$ and a value of parameter β;
Output: a set of ordinal reducts.
(1) compute core attributes $Core$ of A using Sig_{inner}^{β}.
(2) $B \leftarrow A - Core$.
(3) $B^* \leftarrow$ sorted B in the ascending order in terms of $g(a) = \gamma_B^{\beta}(d) + \frac{|U/\{a\}|}{|U|}$, where $U/\{a\}$ is a partition of U induced by the attribute a.
(4) $P^* \leftarrow B^* \cup Core$.
(5) find a reduct RED_0 from P^* with Algorithm 2.
(6) $K \leftarrow RED_0 - Core$.
(7) $RED \leftarrow \emptyset$.
(8) for $i = 1$ to $|K|$
    $P^* \leftarrow P^* - \{a_i\}$;
    find a reduct RED_i from P^* by Algorithm 2;
    if $RED_i \notin RED$, then $RED \leftarrow RED + RED_i$;
    $P^* \leftarrow P^* \cup \{a_i\}$.
(9) return $RED = \{RED_0, RED_1, \ldots, RED_N\}$.

The algorithm can produce a set of monotonic attribute reducts satisfying monotonic consistent with respect to the parameter β. These different ordinal reducts lay a foundation for constructing complementary monotonic decision trees.

Let $F = \{T_1, T_2, \ldots, T_N\}$ be a monotonic decision forest learned by $RED = \{RED_1, RED_2, \ldots, RED_N\}$ in the training set, and decisions $w = \{w_1, w_2, \ldots, w_s\}$. Denote the label of an object x obtained by the monotonic decision tree T_i by $T_i(x)$. Given an object x in the test set, we determine the class label of x by the following fusing principle.

Fusing principle based on maximal probability:

1) for every T_i in F, given an object x, if the label $T_i(x)$ is the same decisions w_j for each i, then give x the

label w_j, otherwise compute the probability P_{ij} of x belonging to the decision class w_j;

2) compute $P_j(x) = \frac{1}{N}\sum_{i=1}^{N} P_{ij}$ as the probability of x belonging to the decision class w_j;

3) give x the label w_0, where

$$w_0 : \ P_0(x) = \max\{P_j(x), \ 1 \le j \le s\}.$$

Based on the above fusing principle FPBMP, we give an algorithm of fusing monotonic decision trees, which is as follows.

Now, we explain the working mechanism of the fusing principle for fusing monotonic decision trees. It can be understood by an illustrative example. We generate an artificial data set with three classes, in which there are 39 objects and 12 attributes, as shown in Table 1.

In this data set, objects x_1 to x_{27} are treated as the training set of constructing a monotonic decision tree, and objects x_{28} to x_{39} are looked forward as the test set of evaluating the performance of a monotonic decision tree.

To show the difference between method of signing class labels in REMT and that in monotonic decision trees used in the proposed fusing principle, we first construct a monotonic decision tree using REMT with the parameter $\varepsilon = 0.01$ as Fig. 1.

Now, we learn monotonic decision trees used in the proposed fusing principle. Through Algorithm 3, one can obtain two ordinal attribute reducts:

$$RED_1 = \{g, i, c, a\} \quad \text{and} \quad RED_2 = \{h, j, g, f, c, i\}.$$

Using these two monotonic attribute reducts, we can learn two monotonic decision trees with REMT ($\varepsilon = 0.01$), in which the decision of each leaf node is labeled by a family of probabilities of the node belonging to every class. These two ordinal decision trees are shown as sub-figures (a) and (b) in Fig. 2.

In what follows, we consider the decision of each of objects x_{28}, x_{38} and x_{39} in the test set. Their decisions induced by the monotonic decision tree in Fig. 1 and those induced by the proposed method in this study are listed in Table 2, respectively.

Through computing, we have that:

1) for x_{28}, the output of T_1 and that of T_2 are all $L3$, it is labeled as class $L3$;

TABLE 1
An Artificial Data Set with 12 Features, Where 39 Objects are Divided into Three Classes

Data sets	a	b	c	d	e	f	g	h	i	g	k	l	d
x_1	4	5	2	3	3	3	5	4	5	5	4	5	3
x_2	3	5	1	1	2	2	5	3	5	5	3	5	3
x_3	2	3	2	1	2	4	5	2	5	4	3	4	3
x_4	3	4	3	2	2	2	5	3	5	5	3	5	3
x_5	3	5	2	3	4	4	5	4	4	5	3	5	3
x_6	3	4	3	3	2	4	4	2	4	3	1	3	3
x_7	2	5	1	1	3	4	4	3	4	4	3	4	3
x_8	1	1	2	1	1	3	4	2	4	4	1	4	3
x_9	2	3	2	1	1	2	4	4	4	4	2	5	3
x_{10}	2	3	4	3	1	5	4	2	4	3	2	3	3
x_{11}	2	2	2	1	1	4	4	4	4	4	2	4	3
x_{12}	2	1	3	1	1	3	5	2	4	2	1	3	3
x_{13}	2	1	1	1	1	3	2	2	4	4	2	3	2
x_{14}	2	1	2	1	1	2	4	3	3	2	1	2	2
x_{15}	1	1	1	1	1	1	3	2	4	4	2	3	2
x_{16}	2	2	2	1	1	3	3	2	4	4	2	3	2
x_{17}	2	2	1	1	1	3	2	2	4	4	2	3	2
x_{18}	1	1	4	1	3	1	2	2	3	3	1	2	2
x_{19}	1	1	2	1	1	1	3	3	4	4	2	3	1
x_{20}	3	5	2	1	1	1	3	2	3	4	1	3	1
x_{21}	2	2	1	1	1	1	3	3	3	4	3	4	1
x_{22}	2	1	1	1	1	1	2	2	3	4	3	4	1
x_{23}	1	1	2	1	1	1	2	1	4	3	1	2	1
x_{24}	1	1	3	1	2	1	2	1	3	3	2	3	1
x_{25}	1	1	1	1	1	1	2	2	4	4	2	3	1
x_{26}	1	1	3	1	1	1	1	1	4	3	1	3	1
x_{27}	2	1	1	1	1	1	1	1	2	1	1	2	1
x_{28}	2	2	2	2	1	3	5	3	5	4	2	4	3
x_{29}	3	5	3	3	3	2	5	3	4	4	3	4	3
x_{30}	1	1	4	1	2	3	5	2	4	4	1	4	3
x_{31}	3	4	2	1	2	2	4	2	4	4	1	4	3
x_{32}	3	3	4	4	3	4	4	2	4	4	1	3	3
x_{33}	2	1	1	1	4	3	4	2	4	4	3	3	3
x_{34}	2	1	2	1	1	3	4	2	4	4	2	4	3
x_{35}	2	1	2	1	1	5	4	2	4	4	2	4	3
x_{36}	1	1	3	1	2	1	3	4	4	4	3	4	2
x_{37}	2	1	2	1	1	3	2	2	4	4	2	4	2
x_{38}	3	4	4	3	2	3	3	4	4	4	3	4	2
x_{39}	3	1	3	3	1	2	2	3	4	4	2	3	2

2) for x_{38}, $P_1 = (P_{11} + P_{21})/2 = (0.6 + 0)/2 = 0.3$ and $P_2 = (P_{12} + P_{22})/2 = (0.4 + 1)/2 = 0.7$, it is labeled as class $L2$;

3) for x_{39}, the output of T_1 and that of T_2 are all $L2$, it is labeled as class $L2$;

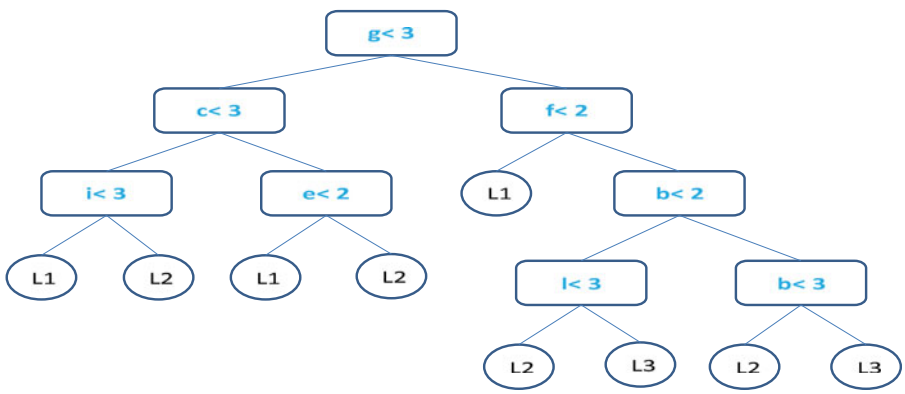
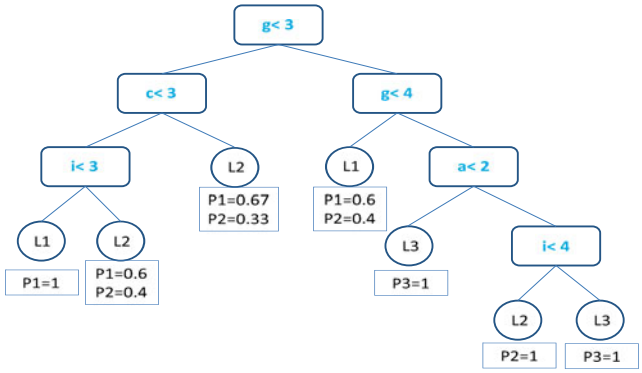
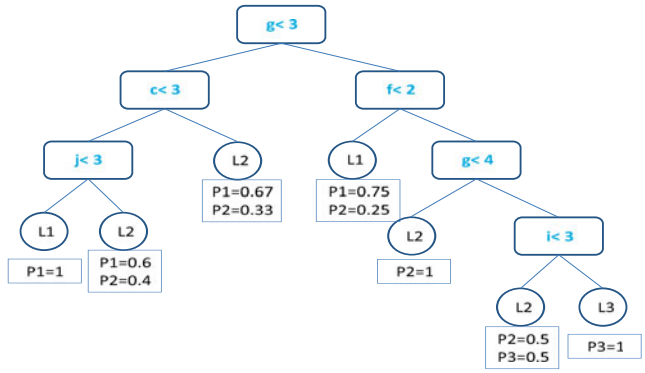


Fig. 1. Monotonic decision tree trained with REMT.



(a) Monotonic decision tree T_1 trained with features $\{g, i, c, a\}$



(b) Monotonic decision tree T_2 trained with features $\{h, j, g, f, c, i\}$

Fig. 2. Monotonic decision trees trained with two feature subsets obtained by feature selection algorithm.

TABLE 2
Difference between REMT and FREMT about the Data Set in Table 1

Objects	Real decisions	Decisons with REMT	Decisions with Reduct 2	Decisons with T_1	Decisons with T_2	Decisons with FREMT
x_{28}	$L3$	$L2$	$L3$	$L3$	$L3$	$L3$
x_{38}	$L2$	$L3$	$L1$	$P_{11} = 0.6, P_{12} = 0.4$	$P_{22} = 1$	$L2$
x_{39}	$L2$	$L1$	$L2$	$L2$	$L2$	$L2$

The corresponding decisions of x_{28}, x_{38}, x_{39} are equivalent to their real decisions $L3$, $L2$ and $L2$. However, their decisions induced by REMT are $L2$, $L3$ and $L1$, respectively. This means that the proposed FREMT has better decision performance than REMT for these two objects. It deserves to point out that decisions of x_{28}, x_{38}, x_{39} induced by $Reduct2$ with REMT are respectively $L3$, $L1$ and $L2$, which are also much closer to real decisions than REMT.

From the above example, we can say that the proposed FREMT may effectively improve the decision performance of the REMT algorithm for monotonic classification. Even if only one ordinal attribute reduct, the monotonic decision tree induced by it with REMT also may have much better generalization.

5 Experimental Analysis

The rank entropy-based decision tree is an effective decision model for monotonic classification. In order to show the effectiveness of the proposed fusing algorithm, in this section, we will compare the proposed algorithm with the rank entropy-based ordinal decision tree on real-world classification tasks.

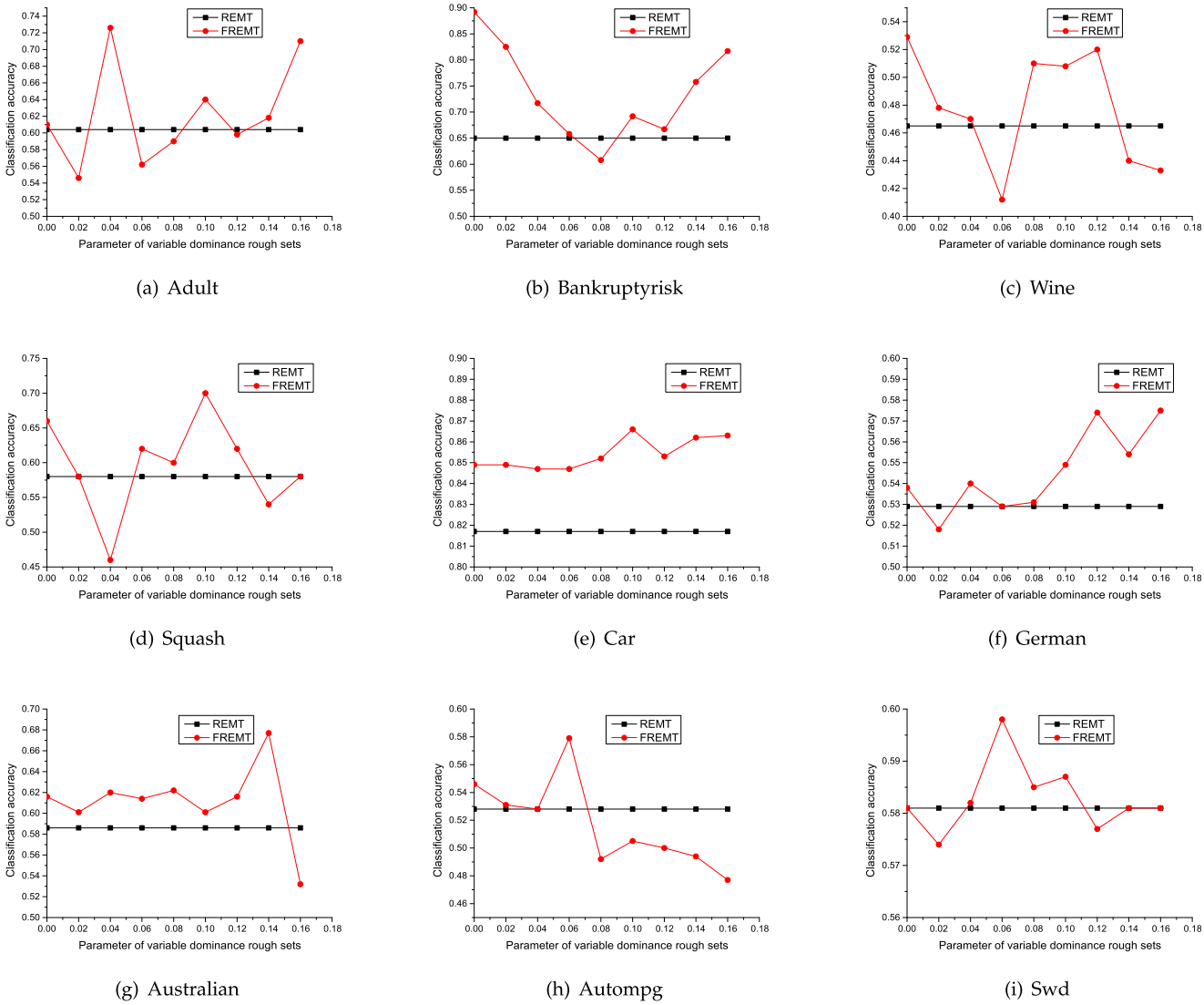
In order to test how our fusing approach behaves in real-world applications, we employed 10 data sets, which are shown as Table 3. In this table, Student score is a real-world data set including 512 students coming from Software Engineering (the class of 2010) in Shanxi University and their scores of 25 courses (features), where 122, 269 and 121 students are evaluated as excellent, good and bad, respectively. Its label distribution is shown as Fig. 5a.

For the student score data set, it is a natural monotonic classification problem. For the first nine data sets, before training the base monotonic decision trees, we need to preprocess these data sets to suit the proposed fusing ordinal decision tree algorithm. As Hu et al. said [19], because we use ascending rank mutual information as the splitting rule, we assume that larger rank value should come from larger feature values, called increasing monotonicity. In practice, we may confront the case that the worse feature value should get the better ranks. This called decreasing monotonicity. To uniformly deal with, we have to transform the problem of decreasing monotonicity to an increasing monotonicity classification task. There are several solutions to this objective. In this experimental analysis, if decreasing monotonicity happens, we will compute reciprocal of attribute values. In order to compare the performance when data sets are monotonic, we relabeled the objects so as to generate monotonic training sets. In this experiment, we revised the labels of some objects and generated monotone training data sets by the monotonization algorithm in the literature [25].

Firstly, we observe the performance of a base monotonic decision tree induced by the proposed attribute reduction algorithm. For each data set, we used 10-fold cross validation technique, in which 90 percent of the data set is used as the training set and the remained objects are used as the test set.

TABLE 3
Nine Data Sets in the Experimental Analysis

Data sets	Num. of objects	Num. of features	Num. of classes
Adult	500	15	2
Bankruptyrisk	39	13	3
Wine	1,599	12	2
Squash	50	25	3
Car	1,727	7	4
German	1,000	21	2
Australian	690	15	2
Autompg	392	8	4
Swd	3,240	11	3
Student score	512	25	3



(a) Adult (b) Bankruptyrisk (c) Wine

(d) Squash (e) Car (f) German

(g) Australian (h) Autompg (i) Swd

Fig. 3. The average value of classification accuracies of every group of base monotonic decision trees.

We here use the classification accuracy and the mean absolute loss to verify the performance of the trained model of each base ordinal decision tree. The classification accuracy for evaluating the performance of a classifier is computed as

$$AC = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y}_i \oplus y_i), \tag{7}$$

in which if $\widehat{y}_i = y_i$, then $\widehat{y}_i \oplus y_i = 1$, otherwise $\widehat{y}_i \oplus y_i = 0$. And, the mean absolute error is calculated as

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y}_i - y_i|, \tag{8}$$

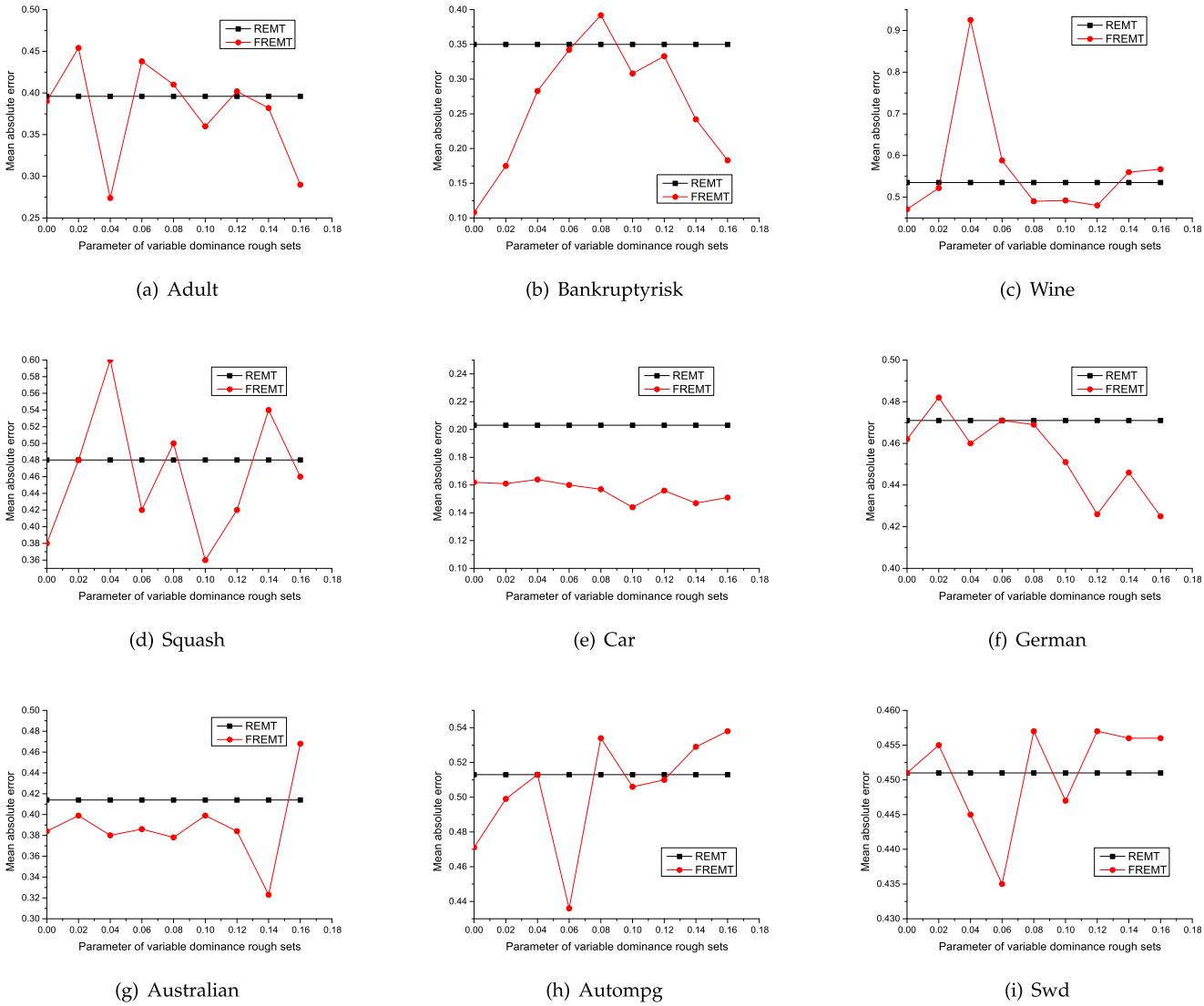
where n is the number of objects in the test set, $\widehat{y}_i$ is the output of the algorithm and y_i is the real output of the ith object.

In the experimental analysis, let $\varepsilon = 0.01$ and β vary from 0.00 to 0.16 with a step length 0.02. Based on 10-fold cross validation technique, the average performances of the classification accuracy and the mean absolute loss are computed and shown in Figs. 3 and 4, respectively.

From the curves in Fig. 3, we see that most of base ordinal decision trees possess much higher classification accuracies than REMT in most cases, except the data set Autompg. Moreover, regarding the curves in Fig. 4, it can be seen that most of base ordinal decision trees have much lower mean absolute loss than REMT in most cases, except the data set Swd. In addition, we also can see the classification accuracies (or the mean absolute loss) of these base monotonic decision trees are often different each other, which can satisfy the diversity constraint in ensemble learning.

In what follows, we verify the performance of the fusing ordinal decision trees induced by sub monotonic decision trees. Given β varying from 0.00 to 0.16 with a step length 0.02, we firstly compute attribute reducts from original data sets, then use these attribute reducts to learn every sub monotonic decision tree with REMT, where $\varepsilon = 0.01$, and finally fusing these decision trees according to Algorithm 4.

For each data set, we still used 10-fold cross validation technique. The classification accuracy and the mean absolute loss are used to verify the performance of the fusing monotonic decision trees. The experimental results are listed in Table 4.



(a) Adult (b) Bankruptyrisk (c) Wine

(d) Squash (e) Car (f) German

(g) Australian (h) Autompg (i) Swd

Fig. 4. The average value of mean absolute losses of every group of base monotonic decision trees.

Algorithm 4. Fusing rank entropy based monotonic decision trees (FREMT)

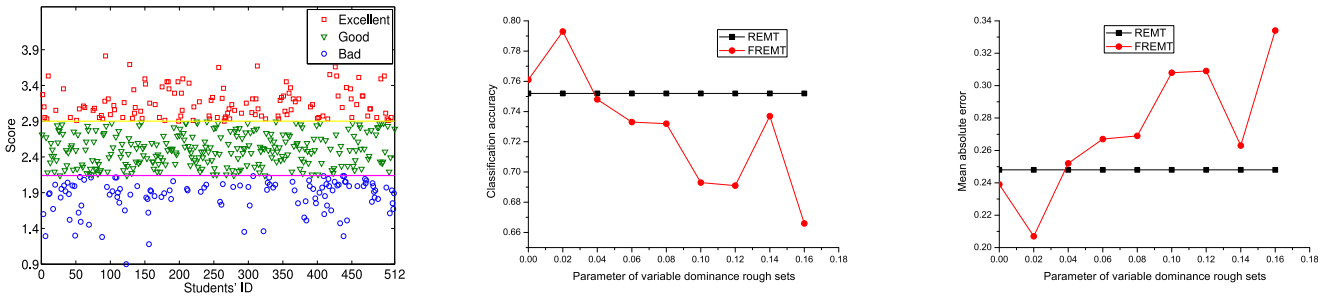
Input: a decision table $DT = (U, A \cup \{d\})$, the parameters $\beta = \{\beta_1, \beta_2, \ldots, \beta_m\}$ and an object x depicted by A;
Output: a decision of object x.
(1) from $i = 1$ to m
finding all ordinal reducts with Algorithm 3
$RED_i = \{RED_0, RED_1, \ldots, RED_{N_i}\}$
(2) $RED \leftarrow \bigcup_{i=1}^{m} RED_i$
(3) for $RED_j \in RED$, learn a tree T_j with REMT.
(4) $F \leftarrow \{T_1, T_2, \ldots, T_N\}$.
(5) determine the class label of x by F with FPBMP.
(6) end.

Table 4 presents the classification accuracy and the mean absolute loss yielded with two learning algorithms FREMT and REMT. It can be seen from Table 4 that for each data set, FREMT are consistently better than REMT for both AC and MAE. It deserves to point out that FREMT can significantly improve the generalization ability of REMT for these nine monotonic classification tasks. If we adopt a selective ensemble strategy, the fusing monotonic decision forest will possess much better generalization performance. We will follow it with interest in further work.

Remark 1. In fact, the parameter β also plays an important role in terms of tuning the performances of the FREMT algorithm. Different choices of β might produce different trees to combine in the ensemble. This can be induced to two reasons: the value of the parameter β itself and its step length. The latter is used to determine the number of different decision trees to fuse. The former is used to loose the condition of a monotonic attribute reduct. For a fixed step, bigger β may mean more decision trees and much better diversity among them. In fact, it is difficult to specify optimal values of β, which might be chosen by cross-validation. However, it is beyond the scope of this paper. We omit its detailed discussion here.

In what follows, we conclude the advantages of the proposed fusing principle FPBMP and analyze their reasons.

- Most of base monotonic decision trees induced by ordinal attribute reducts have better generalization ability themselves.



(a) Student score distribution (b) Classification accuracy (c) Mean absolute loss

Fig. 5. Label distribution and base monotonic decision trees on the student score data set.

As we know, attribute reduction has been proven effective in improving classification performance and avoiding overfitting. A monotonic attribute reduct can satisfy the same monotonic constraint as the original set of attributes, which can preserve the rank among objects unchanged. The size of an obtained monotonic attribute reduct is much smaller than that of the original attribute set, and hence the base monotonic decision tree induced by the ordinal attribute reduct will have much smaller length and much fewer nodes, which usually possesses much better generalization ability. In addition, we set a variable parameter β to loosen the condition of a monotonic attribute reduct, such that the sensitivity of the learned base monotonic decision tree is reduced for noise. Therefore, most of base monotonic decision trees will have better generalization ability than the monotonic decision tree induced by the original attribute set in a given data set.

- The base monotonic decision trees satisfy the diversity requirements in ensemble learning.

  Diversity among the individual learners is deemed to be a key issue in ensemble learning, which can significantly improve the generalization ability of machine learning systems. In this study, the diversity among base monotonic decision trees can be guaranteed by the diversity among monotonic attribute reducts from the original attribute set. In general, different monotonic attribute reducts would learn different monotonic decision trees. This implies that one can learn much more decision rules from the different ordinal decision trees, which would provide much better predictive ability for unlabeled objects.
- The learning system fused by FPBMP significantly improve the generalization ability of monotonic decision trees.

The success of the fusing principle is attributed to two factors. On one hand, when the outputs of a tested object in different base monotonic decision trees are equal to each other, its final label will be assigned as this output. On the other hand, when these outputs of the object are inconsistent each other, we accumulate the evidences that it belongs to different classes in all base ordinal decision trees, and assign it to the class with the maximal probability. In fact, its rationality also can be understood by Example 1. This may be a more reasonable ensemble solution than a simple majority vote strategy, which can be safely used to fuse monotonic decision trees.

Remark 2. From the above discussions, we know that each of base monotonic decision trees is induced on a different subset of features. When the number of features is small, the number of possible different base trees in the forest may be small too. This would affect the performance of the FPBMP algorithm. Nonetheless, this problem might be solved through adding randomness in the decision tree induction procedure. However, it is beyond the scope of this paper, which can be followed with interest in further work.

6 Conclusions and Further Work

Ordinal classification is a kind of special classification tasks, in which a monotonicity constraint is considered as the fundamental assumption. This assumption argues that the objects with better attribute values should not be assigned to a worse decision class. In recent years, several learning

TABLE 4
Comparison on Classification Accuracy and Mean Absolute Loss

Data sets	Number of base decision trees	AC of FREMT	AC of REMT	MAE of FREMT	MAE of REMT
Adult	36	0.774 ± 0.001	0.604 ± 0.016	0.226 ± 0.001	0.396 ± 0.016
Bankruptyrisk	27	0.858 ± 0.025	0.650 ± 0.036	0.142 ± 0.025	0.350 ± 0.036
Wine	14	0.626 ± 0.001	0.465 ± 0.004	0.374 ± 0.001	0.535 ± 0.004
Squash	68	0.740 ± 0.008	0.580 ± 0.060	0.260 ± 0.008	0.480 ± 0.066
Car	12	0.871 ± 0.000	0.817 ± 0.001	0.148 ± 0.000	0.203 ± 0.001
German	45	0.711 ± 0.001	0.529 ± 0.001	0.289 ± 0.001	0.471 ± 0.001
Australian	47	0.735 ± 0.002	0.586 ± 0.003	0.265 ± 0.002	0.414 ± 0.003
Autompg	27	0.594 ± 0.005	0.528 ± 0.003	0.431 ± 0.008	0.513 ± 0.004
Swd	14	0.683 ± 0.001	0.581 ± 0.001	0.341 ± 0.001	0.451 ± 0.001
Student score	77	0.851 ± 0.003	0.752 ± 0.002	0.149 ± 0.003	0.248 ± 0.002

algorithms have been proposed to handle this kind of special tasks. The rank entropy-based ordinal decision tree is very representative thanks to its better ability of robust and generalization. To further improve the generalization ability of a machine learning system based on ordinal decision trees, in this paper, we aim to investigate how to fuse ordinal decision trees from the viewpoint of ensemble learning. To address this issue, we have taken two factors into account: attribute reduction and fusing principle. Firstly, we have introduced an attribute reduction method with rank-preservation property based on the variable dominance rough sets, which is used to generate some monotonic attribute reducts and learn base classifiers depicted by monotonic decision trees. In general, the size of an ordinal attribute reduct is much shorter than that of the original attribute set, and the corresponding monotonic decision tree induced by it may have much better generalization ability. This may ensure that each of these base classifiers is a stronger learner. Through adjusting values of the parameter β in variable dominance rough sets, we can obtain various monotonic attribute reducts from the original data set, which can be used to learn different base classifiers. This can satisfy the diversity among base classifiers in ensemble learning. Then, based on the idea of maximal probability, we have established a fusing principe for combining the base classifiers, which is used to further improve generalization ability of the learning system. Finally, we have verified the performance of the method proposed in this study through employing nine real data sets. The experimental analysis shows that the proposed fusing method can significantly improve classification performance of monotonic decision trees.

It deserves to point out that sometimes the learning system fusing all trained base classifiers may cause overfitting. Selective ensemble learning could further enhance the performance of the learning system. Hence, it is an important problem that how to select base monotonic decision trees so that the fused learning system possesses as well performance as possible. We will work on this problem in the future.

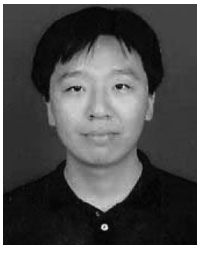
Acknowledgments

This work was supported by National Natural Science Fund of China (Nos. 61322211, 61432011, U1435212), the US National Science Foundation grants (IIS-1111092, IIS-1407927), Program for New Century Excellent Talents in University (No. NCET-12-1031), the Research Fund for the Doctoral Program of Higher Education (No. 20121401110013), and National Key Basic Research and Development Program of China(973) (Nos. 2013CB329404, 2013CB329502).

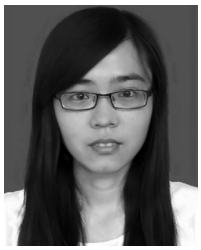
References

[1] N. Barile and A. Feelders, "Nonparametric monotone classification with 628 MOCA," in *Proc. IEEE 8th Int. Conf. Data Mining*, 2008, pp. 731–736.
[2] A. Ben-David, L. Sterling, and Y. H. Pao, "Learning and classification of 620 monotonic ordinal concepts," *Comput. Intell.*, vol. 5, pp. 45–49, 1989.
[3] A. Ben-David, "Automatic generation of symbolic multiattribute ordinal 644 knowledge-based DSSs: Methodology and applications," *Decision Sci.*, vol. 23, pp. 1357–137, 1992.
[4] A. Ben-David, "Monotonicity maintenance in information-theoretic machine learning algorithms," *Mach. Learn.*, vol. 19, pp. 29–43, 1995.
[5] A. Ben-David, L. Sterling, and T. Tran, "Adding monotonicity to learning algorithms may impair their accuracy," *Expert Syst. Appl.*, vol. 36, pp. 6627–6634, 2009.
[6] K. Cao-Van and B. D. Baets, "Growing decision trees in an ordinal setting," *Int. J. Intell. Syst.*, vol. 18, pp. 733–750, 2003.
[7] D. Chen, S. Zhao, L. Zhang, Y. Yang, and X. Zhang, "Sample pair selection for attribute reduction with rough set," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 11, pp. 2080–2093, Nov. 2012.
[8] D. Chen, T. Li, D. Ruan, J. Lin, and C. Hu, "A rough-set-based incremental approach for updating approximations under dynamic maintenance environments," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 2, pp. 274–284, Feb. 2013.
[9] H. A. M. Daniels and M. V. Velikova, "Derivation of monotone decision models from noisy data," *IEEE Trans. Syst., Man Cybern., Part C: Appl. Rev.*, vol. 36, no. 5, pp. 705–710, Sep. 2006.
[10] D. Dubois, H. Fargier, and P. Perny, "Qualitative decision theory with preference relations and comparative uncertainty: An axiomatic approach," *Artif. Intell.*, vol. 148, pp. 219–260, 2003.
[11] W. Duivesteijn and A. Feelders, "Nearest neighbour classification with monotonicity constraints," in *Proc. Eur. Conf. Mach. Learn.*, 2008, pp. 301–316.
[12] A. J. Feelders and M. Pardoel, "Pruning for monotone classification trees," in *Proc. 5th Int. Symp. Intell. Data Anal.*, 2003, pp. 1–12.
[13] P. M. Granitto, P. F. Verdes, and H. A. Ceccatto, "Neural network ensembles: Evaluation of aggregation algorithms," *Artif. Intell.*, vol. 16, pp. 3139–3162, 2005.
[14] S. Greco, B. Matarazzo, and R. Slowinski, "Rough approximation of a preference relation by dominance relations," *Eur. J. Oper. Res.*, vol. 117, pp. 63–83, 1999.
[15] S. Greco, B. Matarazzo, and R. Slowinski, "Rough approximation by dominance relations," *Int. J. Intell. Syst.*, vol. 17, pp. 153–171, 2002.
[16] S. Greco, B. Matarazzo, R. Slowinski, and J. Stefanowski, "Variable consistency model of dominance-based rough sets approach," in *Proc. 2nd Int. Conf. Rough Sets Current Trends Comput.*, 2001, pp. 170–181.
[17] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.
[18] Q. Hu, D. Yu, Z. Xie, and X. Li, "EROS: Ensemble rough subspaces," *Pattern Recognit.*, vol. 40, pp. 3728–3739, 2007.
[19] Q. Hu, X. Che, L. Zhang, D. Zhang, M. Guo, and D. Yu, "Rank entropy based decision trees for monotonic classification," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 11, pp. 2052–2064, Nov. 2012.
[20] Q. Hu, W. Pan, L. Zhang, D. Zhang, Y. Song, M. Guo, and D. Yu, "Feature selection for monotonic classification," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 1, pp. 69–81, Feb. 2012.
[21] Q. Hu, D. Yu, and M. Guo, "Fuzzy preference based rough sets," *Inf. Sci.*, vol. 180, no. 10, pp. 2003–2022, 2010.
[22] Q. H. Hu, D. R. Yu, W. Pedrycz, and D. G. Chen, "Kernelized fuzzy rough sets and their applications," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 11, pp. 1649–1667, Nov. 2011.
[23] R. V. Kamp, A. J. Feelders, and N. Barile, "Isotonic classification trees," in *Proc. 8th Int. Symp. Intell. Data Anal.*, 2009, vol. 5772, pp. 405–416.
[24] W. Kotlowski, K. Dembczynski, S. Greco, and R. Slowinski, "Stochastic dominance-based rough set model for ordinal classification," *Inf. Sci.*, vol. 178, pp. 3989–4204, 2008.
[25] W. Kotlowski and R. Slowinski, "Rule learning with monotonicity constraints," in *Proc. 26th Int. Conf. Mach. Learn.*, 2009, pp. 537–544.
[26] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles," *Mach. Learn.*, vol. 51, pp. 181–207, 2003.
[27] J. Y. Liang, F. Wang, C. Y. Dang, and Y. H. Qian, "A group incremental approach to feature selection applying rough set technique," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 294–308, Feb. 2014.
[28] S. Lievens, B. D. Baets, and K. Cao-Van, "A probabilistic framework for 647 the design of instance-based supervised ranking algorithms in an ordinal setting," *Ann. Oper. Res.*, vol. 163, no. 1, pp. 115–142, 2008.
[29] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005.
[30] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, London, U.K.: Kluwer, 1991.

[31] W. Pedrycz and M. L. Song, "A granulation of linguistic information in AHP decision-making problems," *Inf. Fusion*, vol. 17, pp. 93–101, 2014.
[32] R. Potharst and A. J. Feelders, "Classification trees for problems with monotonicity constraints," *ACM SIGKDD Explorations Newslett.*, vol. 4, no. 1, pp. 1–10, 2002.
[33] R. Potharst and J. C. Bioch, "Decision trees for ordinal classification," *Intell. Data Anal.*, vol. 4, no. 2, pp. 97–112, 2000.
[34] Y. H. Qian, J. Y. Liang, W. Pedrycz, and C. Y. Dang, "Positive approximation: An accelerator for attribute reduction in rough set theory," *Artif. Intell.*, vol. 174, nos. 9/10, pp. 597–618, 2010.
[35] Y. H. Qian, J. Y. Liang, W. Z. Wu, and C. Y. Dang, "Information granularity in fuzzy binary GrC model," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 253–264, Apr. 2011.
[36] Y. H. Qian, J. Y. Liang, and C. Y. Dang, "Incomplete multigranulation rough set," *IEEE Trans. Syst., Man Cybern.-Part A*, vol. 40, no. 2, pp. 420–431, Mar. 2010.
[37] Y. H. Qian, J. Y. Liang, C. Y. Dang, and D. W. Tang, "Set-valued ordered information systems," *Inf. Sci.*, vol. 179, pp. 2809–2832, 2009.
[38] Y. H. Qian, J. Y. Liang, and C. Y. Dang, "Interval ordered information systems," *Comput. Math. Appl.*, vol. 56, pp. 1994–2009, 2008.
[39] Y. H. Qian, H. Zhang, Y. L. Sang, and J. Y. Liang, "Multigranulation decision-theoretic rough sets," *Int. J. Approximate Reasoning*, vol. 55, no. 1, pp. 225–237, 2014.
[40] R. Senge and E. Hullermeier, "Top-down induction of fuzzy pattern trees," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 241–252, Apr. 2011.
[41] C. E. Shannon, "The mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, nos. 3/4, pp. 373–423, 1948.
[42] M. W. Shao and W. X. Zhang, "Dominance relation and rules in an incomplete ordered information system," *Int. J. Intell. Syst.*, vol. 20, pp. 13–27, 2005.
[43] H. W. Shin and S. Y. Sohn, "Selected tree classifier combination based on both accuracy and error diversity," *Pattern Recognit.*, vol. 38, pp. 191–197, 2005.
[44] W. Z. Wu, Y. Leung, and J. S. Mi, "Granular computing and knowledge reduction in formal contexts," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 10, pp. 1461–1474, Oct. 2009.
[45] G. D. Wu, Z. W. Zhu, and P. H. Huang, "A TS-type maximizing discriminability-based recurrent fuzzy network for classification problems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 2, pp. 339–352, Apr. 2011.
[46] F. Xia, W. S. Zhang, F. X. Li, and Y. W. Wang, "Ranking with decision tree," *Knowl. Inf. Syst.*, vol. 17, pp. 381–395, 2008.
[47] Y. Y. Yao, "Probabilistic rough set approximations," *Int. J. Approx. Reasoning*, vol. 49, no. 2, pp. 255–271, 2008.
[48] J. T. Yao, "Granular computing: Perspectives and challenges," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1977–1989, Dec. 2013.
[49] S. Y. Zhao, E. C. C. Tsang, and X. Z. Wang, "Building a rule-based classifier-a fuzzy rough set approach," *IEEE Trans. Knowl. Eng.*, vol. 22, no. 5, pp. 624–638, May 2010.
[50] Z. H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL, USA: CRC, 2012.
[51] Z. H. Zhou, J. X. Wu, and W. Tang, "Ensembling neural networks: Many could be better than all," *Artif. Intell.*, vol. 137, pp. 239–263, 2002.

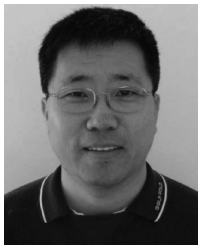


Yuhua Qian received the MS and the PhD degrees in computers with applications at Shanxi University in 2005 and 2011, respectively. He is a professor in the Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, China. He is best known for multigranulation rough sets in learning from categorical data and granular computing. He is actively pursuing research in pattern recognition, feature selection, rough set theory, granular computing, and artificial intelligence. He has published more than 50 articles on these topics in international journals. On professional services, he has served as program chairs or special issue chairs of RSKT, JRS, and ICIC, and PC members of many machine learning, data mining, and granular computing. He also served on the editorial boards of the *International Journal of Knowledge-Based Organizations* and *Artificial Intelligence Research*. He is a member of the IEEE.

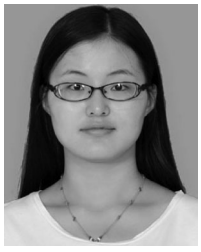


Hang Xu is currently working toward the PhD degree in the School of Computer and Information Technology, Shanxi University. Her research interest includes data mining and knowledge discovery.

Jiye Liang received the BS degree in computational mathematics from Xi'an Jiaotong University. He received the PhD degree in information science from Xi'an Jiaotong University. He is a professor in the School of Computer and Information Technology and Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University. His research interests include artificial intelligence, granular computing, data mining, and knowledge discovery. He has published more than 60 articles in international journals.



Bing Liu received the PhD degree in artificial intelligence from the University of Edinburgh. He is a professor of computer science at the University of Illinois at Chicago (UIC). Before joining UIC, he was with the National University of Singapore. His current research interests include sentiment analysis and opinion mining, opinion spam detection, machine learning, data mining, and natural language processing. He has published extensively on these topics in top conferences and journals. He has also published two books titled *Sentiment Analysis and Opinion Mining* (Morgan and Claypool Publishers) and *Web Data Mining: Exploring Hyperlinks, Contents and Usage Data* (Springer). His work has been widely reported in the international press including a front page article in *The New York Times*. On professional services, he has served as program chairs of KDD, ICDM, CIKM, WSDM, SDM, and PAKDD, and as area/track chairs or senior PC members of many data mining, natural language processing, and web technology conferences. Currently, he serves as the chair of the ACM SIGKDD. He is a fellow of the IEEE.



Jieting Wang is currently working toward the PhD degree in the School of Computer and Information Technology, Shanxi University. Her research interest includes machine learning, data mining, and knowledge discovery.