Maximum Ambiguity-Based Sample Selection in Fuzzy Decision Tree Induction

Xi-Zhao Wang, *Senior Member*, *IEEE*, Ling-Cai Dong, *Student Member*, *IEEE*, and Jian-Hui Yan

Abstract—Sample selection is to select a number of representative samples from a large database such that a learning algorithm can have a reduced computational cost and an improved learning accuracy. This paper gives a new sample selection mechanism, i.e., the maximum ambiguity-based sample selection in fuzzy decision tree induction. Compared with the existing sample selection methods, this mechanism selects the samples based on the principle of maximal classification ambiguity. The major advantage of this mechanism is that the adjustment of the fuzzy decision tree is minimized when adding selected samples to the training set. This advantage is confirmed via the theoretical analysis of the leaf-nodes' frequency in the decision trees. The decision tree generated from the selected samples usually has a better performance than that from the original database. Furthermore, experimental results show that generalization ability of the tree based on our selection mechanism is far more superior to that based on random selection mechanism.

Index Terms—Learning, uncertainty, sample selection, fuzzy decision tree.

1 Introduction

The primary goal of machine learning is to derive general patterns from a limited amount of data. With the development of digital technology, more and more data are produced and stored. But not all these data are useful for machine learning because they usually contain the following three types of data:

- **Noise data.** Noise could reduce the performance of the learner. Generally, it contains some wrong data and outliers.
- **Redundant data.** Different than the duplicated data, redundant data are those which could not affect the performance of the learner.
- **Incomplete data.** Incomplete data are those in which there exist one or some missing values. Specially, the incomplete data are confined to those with missing labels in this paper.

As we known, training set is very important in the learning task. The size of the training set directly affects the performance of the learner. Thus, the research on the acquisition of a high-quality and compact (i.e., small sized) training set such as the sample selection is very significant. Different than a training set randomly drawn from the original data set as usual, sample selection aims to select a representative subset from the original data set as training set on the premise that the performance of the learner generated from the selected subset will not worse or even higher than that of the one trained from the original data set. In the following, we briefly introduce several existing representative sample selection algorithms.

Generally, sample selection can be roughly classified into two categories: data condense and active learning. The former is mainly used to process the first two types of data. It aims to condense the data set by filtering the noises and redundant data. Some representative algorithms are Condensed Nearest Neighbor rule (CNN) serial [3], [14], [15], [26], [37], Instance-Based Learning (IBL) serial [2], [34], [38], and others [6], [8], [39]. The latter is to process the third type of data, i.e., how to select a few representative instances from unlabeled set and thus reduce the labeling cost. Generally, active learning contains three types of queries: uncertainty-based query [7], [18], [20], [28], [31], [32], [33], [44], [45], version space-based query [1], [10], [23], [29], and expect error-based query [5], [9], [24], [27].

In this paper, our study is based on the uncertainty-based query. Clearly, for a certain active learning algorithm, there is always a learning algorithm associated with it. From references, one can see that the associated algorithms are k-Nearest Neighbor (kNN) and Bayesian model in most situations. In this paper, we consider the associated algorithm as the fuzzy decision tree. A new sample selection mechanism has been initially proposed by Wang et al. [36] for the fuzzy decision tree induction. Following the work of Wang et al. [36], this paper makes an attempt to develop a maximum uncertainty-based sample selection mechanism and then to apply it to the fuzzy decision tree learning. Compared with the existing sample selection methods, this mechanism selects the samples based on the principle of maximal classification ambiguity. The major advantage of this mechanism is that the adjustment of the fuzzy decision tree is minimized when adding the selected

- *X.-Z. Wang is with the Machine Learning Center, Department of Mathematics and Computer Science, Hebei University, Baoding 071002, China. E-mail: xizhaowang@ieee.org.*
- *L.-C. Dong and J.-H. Yan are with the Machine Learning Center, Key Laboratory of Machine Learning and Computational Intelligence, College of Mathematics and Computer Science, Hebei University, Baoding 071002, China. E-mail: dawingle@live.cn, JianhYan@163.com.*

Manuscript received 23 Oct. 2009; revised 4 June 2010; accepted 25 Nov. 2010; published online 7 Mar. 2011.
Recommended for acceptance by V. Kumar.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2009-10-0731.
Digital Object Identifier no. 10.1109/TKDE.2011.67.

samples to the training set. This advantage is confirmed via the theoretical analysis of leaf-nodes' frequency in decision trees. The decision tree generated from the selected samples usually has a better performance than that from the original database. Furthermore, experimental results show that the generalization ability of the tree based on our selection mechanism is far more superior to that based on random selection mechanism.

The rest of this paper is organized as follows: Section 2 introduces some uncertainties, presents the relationships between these uncertainties, and gives a brief review on the various uncertainties in different sample selection methodologies. Section 3 presents our proposed sample selection method based on maximum ambiguity in fuzzy decision tree induction and shows the experimental results on some UCI [46] databases. Section 4 gives a further theoretical analysis to illustrate the reasonableness of our proposed sample selection algorithm, i.e., the newly added samples will minimize the modification of the decision tree. Finally, we draw the conclusions in Section 5.

2 Some Uncertainties

Usually uncertainty includes randomness, fuzziness, rough-degree, nonspecificity, etc., where the first two types of uncertainties are considered as the most important. Randomness which is based on the probability distribution was first measured via Shannon's entropy [30]. Fuzziness refers to the unclear boundary between two linguistic terms such as old and young, which is based on the membership functions of fuzzy sets first proposed by Zadeh [41] in 1960s. Rough degree is a type of uncertainty based on partition, which is first given in 1980s by Pawlak [25] in the study of the knowledge representation based on rough sets. Nonspecificity is to describe the uncertainty unfolded in the process of selecting one from two or more than two cases, which was first proposed by Hartley [16] in 1940s. In addition, there exist a number of uncertainties which are different from above mentioned, for example, the U-uncertainty proposed by Higashi and Klir [17] and the credit-based uncertainty given by Liu [22]. The study on relationships among these uncertainties had ever been the focus in uncertainty area. One example is that, through the probability distribution of fuzzy systems, Li [21] established the relationship between randomness and fuzziness. Another example is the study on fuzzy rough sets and rough fuzzy sets [13]. During the recent two decades, the study on uncertainty with domain knowledge has attracted more and more scholars both in the uncertainty mathematics and in machine learning field. Next, we review the different measures of the three uncertainties: randomness, fuzziness, and ambiguity and analyze the relationships between them.

We first list a number of denotations.

$E = \{e_1, e_2, \ldots, e_n\}$ is the instance universe of discourse. Every instance is represented as a vector of which components are values of the same set of attributes.

$C = \{C_1, C_2, \ldots, C_L\}$ is the class label set.

$B = \{\mu_1/e_1, \mu_2/e_2, \ldots, \mu_n/e_n\}$ denotes a fuzzy subset, where μ_i is the membership degree of element e_i belonging to fuzzy subset B. Usually, we denote B in short as $\{\mu_1, \mu_2, \ldots, \mu_n\}$.

$S = \{s_1, s_2, \ldots, s_n\}$ is a sample set. A sample $s_i = (e_i, c_i)$ is a labeled instance, where c_i is a vector $c_i = (c_{i1}, c_{i2}, \ldots, c_{iL})$, each component of which represents the membership degree of the instance e_i belonging to the corresponding class, respectively. For a sample set with crisp labels, the values of $c_{ij}(1 \le i \le n, 1 \le j \le L)$ have two alternatives: "1" and "0." "1" implies that sample s_i belongs to class C_j, and "0" implies that sample s_i does not belong to class C_j. But for a sample set with fuzzy labels, the values of $c_{ij}(1 \le i \le n, 1 \le j \le L)$ can take any value in the interval [0, 1].

Now, we briefly review three kinds of uncertainties and their relationships.

2.1 Entropy

Shannon's entropy is used to measure the impurity of a crisp set. It is proposed by Shannon [30] in 1948, which can be defined as follows:

For a crisp set $E_0 \subseteq E$, the entropy of E_0 can be defined as

$$Entropy(E_0) = -\sum_{i=1}^{L} p_i \ln p_i, \quad (1)$$

where p_i is the proportion of the number of elements that belong to class C_i to the total number of all the elements in E_0. Sometimes, E_0 is represented as $(p_1, p_2, \ldots, p_L)$, where $\sum_{i=1}^{L} p_i = 1, 0 \le p_i \le 1$ for each $i(1 \le i \le L)$.

Clearly, the entropy gets bigger as the proportion of every class gets to equivalent. When all the elements in E_0 belong to the same class, the entropy is minimal; when the elements from every class have the same proportion, the entropy attains its maximum.

2.2 Fuzziness

Fuzziness is a type of cognitive uncertainty. It is caused by the uncertainty transition area from one linguistic term to another, where a linguistic term is a value of linguistic variable. A linguistic variable is a word or a phrase which could take linguistic values. Such as Temperature is a linguistic variable, which can take the linguistic term/values: hot, cool, or middle, etc. Here, hot is a linguistic term, and cool and mild are linguistic terms, too. Essentially, a linguistic term is a fuzzy set. For a fuzzy set $B = \{\mu_1, \mu_2, \ldots, \mu_n\}$, its fuzziness [12] is defined as

$$Fuzziness(B) = -\frac{1}{n}\sum_{i=1}^{n}(\mu_i \ln \mu_i + (1-\mu_i)\ln(1-\mu_i)). \quad (2)$$

Obviously, the fuzziness of a fuzzy set is minimal when every element absolutely belongs to the fuzzy set or absolutely not, i.e., every $\mu_i = 1$ or $\mu_i = 0$ for each $i(1 \le i \le n)$; the fuzziness attains its maximum when the membership degrees of all the elements equal 0.5, i.e., $\mu_i = 0.5$ for every $i = 1, 2, \ldots, n$.

2.3 Ambiguity

Ambiguity is known as the nonspecificity, which is the other type of cognitive uncertainty. It results from choosing one from two or more choices. For example, an interesting film and an expected concert are holding at the same time. In this situation, it is hard for us to decide which one we should attend. This uncertainty associated with the situation is ambiguity. Initially, the concept of ambiguity is

resulting from Hartley Measure [42] in order to measure the nonspecificity of a set. Hartley used the following guideline for a set of classes: the more elements, the bigger ambiguity. Hartley then applied the logarithm function as the measure. Higashi and Klir [17] gave an extending measure of ambiguity to measure the nonspecificity of a fuzzy set. Below is the definition of ambiguity of a fuzzy set B which is defined before Section 2.1

$$Ambiguity(B) = \frac{1}{n}\sum_{i=1}^{n}(\mu_i^* - \mu_{i+1}^*)\ln i, \tag{3}$$

where $\mu_{n+1}^* = 0$ and $(\mu_1^*, \mu_2^*, \ldots, \mu_n^*)$ is the normalization of $(\mu_1, \mu_2, \ldots, \mu_n)$ with $1 = \mu_1^* \geq \mu_2^* \geq \cdots \geq \mu_n^* \geq \mu_{n+1}^* = 0$.

Ambiguity expresses the possible uncertainty of choosing one from many available choices. Obviously, the larger the set of possible alternatives is, the bigger the ambiguity is. The ambiguity will attain its maximum when all the μ_i are equivalent, that is all the $\mu_i^* = 1 (1 \leq i \leq n)$ but $\mu_{n+1}^* = 0$. It will be full specificity, i.e., no ambiguity exists, when only one alternative is possible, i.e., only one μ_i is equal to 1 but all others equal to zero.

2.4 Some Relationships among Three Uncertainties

Shannon's entropy is to measure the uncertainty caused by randomness; fuzziness is to measure the uncertainty of a linguistic term, which is usually represented by a fuzzy set; ambiguity is to measure the nonspecific when choosing one from many available choices.

Specifically, for a set of samples referred in a classification problem, entropy measures the impurity of a crisp set; fuzziness measures the distinction between the set and its complement; ambiguity measures the amount of uncertainty associated with the set of possible alternatives. Clearly, the significance of the three uncertainties is different. The entropy of the sample set is associated with the proportions of the number of samples from every class, which denotes the class impurity of the set. Usually, entropy is restricted to be used in a crisp set. The fuzziness of a sample set is associated with the membership degree of every sample to every class. Fuzziness denotes the sharpness of the border of every fuzzy subset which is formed according to the partition based on classes. Particularly, there is no fuzziness for a crisp set. The ambiguity of a sample set, similar to fuzziness, is associated with the membership degree of every sample to every class. But ambiguity describes the uncertainty associated with the situation resulting from the lack of specificity in describing the best or the most representative one. It emphasizes the uncertainty of possibility of choosing one from many available choices. Next, we discuss the relationships between the three types of uncertainties in detail.

Entropy and fuzziness. Entropy is defined on a probability distribution E_0, i.e., $E_0 = (p_1, p_2, \ldots, p_n)$ where $\sum_{i=1}^{n} p_i = 1$ and $0 \leq p_i \leq 1$ for each $i (1 \leq i \leq n)$. Fuzziness is defined on a possibility distribution B, i.e., a fuzzy set $B = (\mu_1, \mu_2, \ldots, \mu_n)$ where $0 \leq \mu_i \leq 1$ for each $i (1 \leq i \leq n)$. Since a probability distribution is a possibility distribution, we can consider the fuzziness of a probability distribution E_0. We now analyze the relationships between entropy and fuzziness based on a probability distribution $E_0 = (p_1, p_2, \ldots, p_n)$.

As a special case, when E_0 is a two-dimensional probability distribution, i.e., $E_0 = (p_1, p_2)$, then

$$\begin{aligned} Entropy(E_0) &= -p_1 \ln p_1 - p_2 \ln p_2 \\ &= -p_1 \ln p_1 - (1 - p_1)\ln(1 - p_1) \end{aligned} \tag{4}$$

$$Fuzziness(E_0) = 2(-p_1 \ln p_1 - (1 - p_1)\ln(1 - p_1)). \tag{5}$$

Obviously,

$$Fuzziness(E_0) = 2Entropy(E_0). \tag{6}$$

Suppose that $E_0 = (p_1, p_2, \ldots, p_n)$ is a n-dimensional probability distribution. The entropy and fuzziness of E_0 are defined in (1) and (2), respectively. We now discuss their monotonic properties and extreme-value points.

Noting that $p_1 + p_2 + \cdots + p_n = 1$, we assume that, without losing generality, p_1 is the single variable and $p_2, \ldots, p_{n-1}$ are $n - 2$ constants with $p_2 + \cdots + p_{n-1} = c$ and $p_n = 1 - c - p_1$. Then, (1) and (2) degenerate, respectively, to

$$Entropy(E_0) = -p_1 \ln p_1 - (1 - p_1 - c)\ln(1 - p_1 - c) + A \tag{7}$$

and

$$\begin{aligned} Fuzziness(E_0) = Entropy(E_0) &- (1 - p_1)\ln(1 - p_1) \\ &- (p_1 + c)\ln(p_1 + c) + B, \end{aligned} \tag{8}$$

where A and B are two constants which are independent on p_1. By solving

$$\frac{d}{dp_1} Entropy(E_0) = 0 \tag{9}$$

and

$$\frac{d}{dp_1} Fuzziness(E_0) = 0, \tag{10}$$

we can obtain that both $Entropy(E_0)$ and $Fuzziness(E_0)$ with respect to the variable p_1 attain the maximum at $p_1 = \frac{1-c}{2}$ and monotonically increase when $p_1 < \frac{1-c}{2}$ and monotonically decrease when $p_1 > \frac{1-c}{2}$. It indicates that the entropy and fuzziness for a probability distribution have the same monotonic characteristics and extreme-values points. Furthermore, noting that $p_2 + \cdots + p_{n-1} = c$ and the symmetry of variables $p_1, p_2, \ldots, p_n$, we conclude that the entropy and fuzziness of a n-dimensional probability distribution attain their maximum at

$$p_1 = p_2 = \cdots = p_n = 1/n. \tag{11}$$

Roughly speaking, the fuzziness is an extension of the entropy.

When $n = 3$, the entropy and the fuzziness of $E_0 = (p_1, p_2, p_3)$ are depicted in Fig. 1, which is the 3D contour plot of entropy of $E_0 = (p_1, p_2, p_3)$. Correspondingly, Fig. 2 is the 3D contour plot of fuzziness. From Figs. 1 and 2, we can see that the entropy and fuzziness of $E_0 = (p_1, p_2, p_3)$ have many common features. Both of them are with the same shape. They attain their maximum when $p_1 = p_2 = 0.33$ and minimum when $p_1 = p_2 = 0$, or $p_1 = 0, p_2 = 1$, or $p_1 = 1, p_2 = 0$.

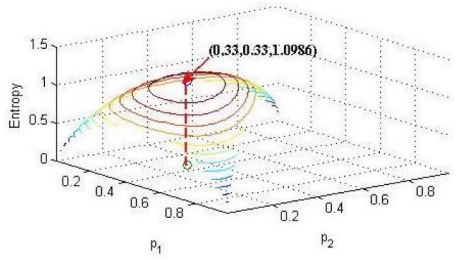

Fig. 1. Entropy of $E_0 = (p_1, p_2, p_3)$.

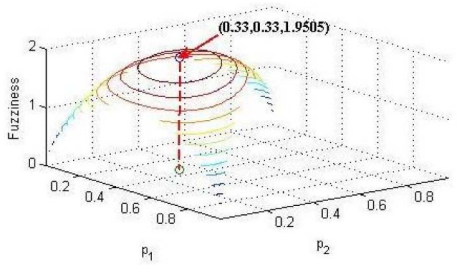

Fig. 2. Fuzziness of $E_0 = (p_1, p_2, p_3)$.

Fuzziness and ambiguity. Both of the fuzziness and ambiguity are defined on a fuzzy set. For a general possibility distribution $B = (\mu_1, \mu_2, \ldots, \mu_n)$, of which the fuzziness is defined as (2), we derive by solving

$$\begin{aligned} &\frac{\partial}{\partial \mu_1} Fuzziness(B) \\ &= \frac{\partial}{\partial \mu_2} Fuzziness(B) \\ &= \cdots \\ &= \frac{\partial}{\partial \mu_n} Fuzziness(B) \end{aligned} \tag{12}$$

that $Fuzziness(B)$ attains its maximum at

$$\mu_1 = \mu_2 = \cdots = \mu_n = 0.5. \tag{13}$$

It is easy to check that $Fuzziness(B)$ attains its minimum (zero) only at $\mu_i = 0$ or $\mu_i = 1$ for each $i (1 \le i \le n)$. The monotonic feature of the $Fuzziness(B)$ is simple, that is, $Fuzziness(B)$ monotonically increases in $(0, 0.5)$ and monotonically decreases in $(0.5, 1)$ for each $\mu_i (1 \le i \le n)$.

The ambiguity of $B = (\mu_1, \mu_2, \ldots, \mu_n)$ is defined as (3). According to [17], $Ambiguity(B)$ is a function with monotonicity, continuity, and symmetry. Its maximum is attainted at $\mu_1 = \mu_2 = \cdots = \mu_n$ and its minimum is attained at the case only one $\mu_j (1 \le j \le n)$ is not zero.

Combining the above analysis of $Fuzziness(B)$ and $Ambiguity(B)$, we can think of that the essential difference between the two uncertainties exists.

When $n = 2$, the fuzziness and ambiguity of $B = (\mu_1, \mu_2)$ can degenerate, respectively, to

$$\begin{aligned} Fuzziness(B) = &-\mu_1 \ln \mu_1 - (1-\mu_1)\ln(1-\mu_1) \\ &- \mu_2 \ln \mu_2 - (1-\mu_2)\ln(1-\mu_2) \end{aligned} \tag{14}$$

and

$$Ambiguity(B) = \begin{cases} (\mu_1/\mu_2) \cdot \ln 2, & \text{if } 0 \le \mu_1 < \mu_2; \\ \ln 2, & \text{if } \mu_1 = \mu_2; \\ (\mu_2/\mu_1) \cdot \ln 2, & \text{others.} \end{cases} \tag{15}$$

The pictures of $Fuzziness(B)$ in (14) and $Ambiguity(B)$ in (15) are depicted in Fig. 3.

From Fig. 3a, we can see that the plot of fuzziness of $B = (\mu_1, \mu_2)$ is plane symmetric and the symmetrical planes are $\mu_1 = 0.5$ and $\mu_2 = 0.5$, respectively. Fuzziness of $B = (\mu_1, \mu_2)$ attains its maximum at (0.5, 0.5) and attains its minimum at four endpoints (0,0), (0,1), (1,0), and (1,1).

In Fig. 3b, the surface above the plane $z = 0$ is the ambiguity of $B = (\mu_1, \mu_2)$. The lines in the plane $z = 0$ are the contours of the ambiguity of $B = (\mu_1, \mu_2)$. From the contours, we can see that the points with the same values of μ_1/μ_2 are with the same ambiguity. The more the contour approaches the line $\mu_1 = \mu_2$, the more of the ambiguity is.

2.5 Some Sample Selection Methods Associating with Uncertainties

From references, one can find a number of sample selection mechanisms which are based on simplified or revised uncertainties. Here, we give a brief list of some representative sample selection methods which adopt these uncertainties as their key techniques.

The first sample selection method based on uncertainty is proposed by Angluin [4] in 1988, which queries an instance in the area of uncertainty. The area of uncertainty in [4] is defined as

$$UA_S = \{s | \exists H_i, H_j \in VS \ \& \ H_i(s) \ne H_j(s), i \ne j, s \in S\}, \tag{16}$$

where H denotes a hypothesis, $VS = \{H_1, H_2, \ldots, H_m\}$ denotes the version space, which is a collection of hypotheses that are consistent with the training set; s is a sample in training set S. This method aims to learn a

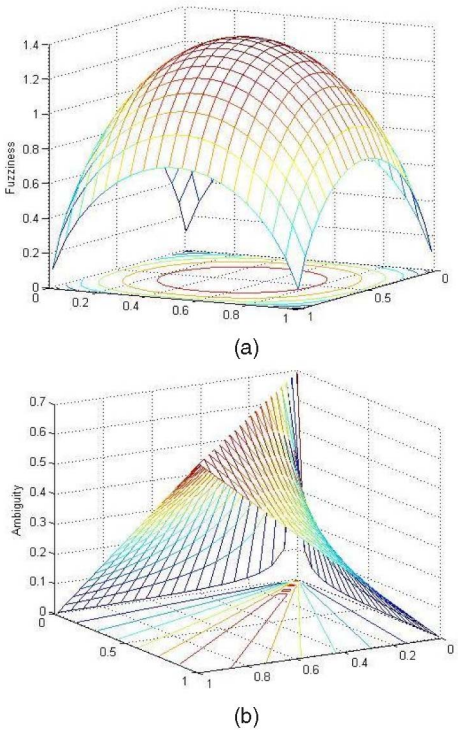

Fig. 3. (a) Fuzziness and (b) ambiguity of $B = (\mu_1, \mu_2)$.

concept by gradually reducing the size of uncertainty area using the queried samples. Compared with original concept learning algorithms, it avoids to querying those instances that are in the determined area. It just queries those instances in the uncertainty area, by which some hypotheses that are not consistent with the queries will be removed. Finally, a concept is learned. Therefore, the query mechanism accelerates the learning rate because it avoids to query the useless instances.

And then, Seung et al. [29] in 1992 developed the Query By Committee (QBC) algorithm, which is to query the instances according to the principle of maximal disagreement of the committee. QBC is based on such an observation that an instance with maximal disagreement is the most difficult one to classify. Disagreement is considered as a kind of uncertainty in QBC, which is defined as

$$VE_{n+1} = -\frac{|V_1|}{|V|}\log_2\frac{|V_1|}{|V|} - \frac{|V_2|}{|V|}\log_2\frac{|V_2|}{|V|}, \tag{17}$$

where VE_{n+1} denotes the voting entropy which is to measure the disagreement by using current committee for the (n+1)th instance denoted by e;

$|X|$ is the cardinality of a set X;

V the current version space defined on the training set S_n;

V_1 and V_2 are the version spaces defined on the next training set $S_{n+1} = S_n \cup \{e\}$ after the (n+1)th sample is added to the training set S_n under the assumption that the newly added instance e belongs to the first class and second, respectively.

Furthermore, Lewis et al. [20], [19] in 1994 proposed the uncertainty sampling. Different from QBC which is a voting mechanism based on many classifiers, uncertainty sampling mechanism just builds only one classifier which could predict the label of an instance and could provide a measurement of how certainty the prediction is. It selects the instance which is speculated to be probably misclassified because the label of the instance is unknown before asking experts. The uncertainty is associated with the posterior probability via Bayesian rule.

3 Our Proposed Maximum-Ambiguity-Based Sample Selection (MABSS) in Fuzzy Decision Tree Induction

3.1 Some Terminologies

Fuzzy decision tree is an extension of crisp decision tree to uncertainty environments. Similar to a crisp decision tree, a fuzzy decision tree is a directed acyclic graph, in which each edge connects two nodes from parent node to child node. The node which has no parent nodes is called the root, while the nodes which have no child nodes are called leaves. Different from crisp decision tree, each node in fuzzy decision tree represents a fuzzy subset. The root is the universal of discourse. All the child nodes generated from the same parent node constitute a fuzzy partition.

Consider a certain node R which is a fuzzy set defined on the sample space S. Let $C = \{C_1, C_2, \ldots, C_L\}$ be the class label set. It means that C_i is a fuzzy set defined on S for each $i(1 \le i \le L)$.

Definition 1. *The* relative frequency *of R to every class is defined as*

$$p_i = \frac{|C_i \bigcap R|}{|R|} = \frac{\sum_{j=1}^{n} min\{C_i(e_j), R(e_j)\}}{\sum_{k=1}^{n} R(e_k)}, \quad i = 1, 2, \ldots, L, \tag{18}$$

where $S = (e_1, e_2, \ldots, e_n)$. In references, p_i is considered as the degree of the implication $R \Rightarrow C_i$.

Definition 2. *If R is a leaf-node, the* classification ambiguity *of R is defined as*

$$Ambiguity(R) = \sum_{i=1}^{L}(p_i^* - p_{i+1}^*)\ln i, \tag{19}$$

where $(p_1, p_2, \ldots, p_L)$ is the relative frequency vector of R; $(p_1^*, p_2^*, \ldots, p_L^*)$ is the normalization of $(p_1, p_2, \ldots, p_L)$ with $1 = p_1^* \ge \cdots \ge p_i^* \ge p_{i+1}^* \ge \cdots \ge p_{L+1}^* = 0 (1 \le i \le L)$.

Definition 3. *If R is a nonleaf node having m child nodes $R_1, R_2, \ldots, R_m$, which are generated according to its corresponding values $V_1, V_2, \ldots, V_m$ of the expanding attribute F, i.e., $R_i = R \bigcap V_i, i = 1, 2, \ldots, m$. We define the weighted average ambiguity*

$$\begin{aligned} Ambiguity(R) &= \sum_{i=1}^{m} w_i \cdot Ambiguity(R_i) \\ &= \sum_{i=1}^{m} \frac{|R_i|}{|R|} \cdot Ambiguity(R_i) \end{aligned} \tag{20}$$

as the classification ambiguity of the nonleaf node R.

In this study, we define the ambiguity of a fuzzy decision as the averaged classification ambiguity of the root, which could be calculated recursively from the leaf nodes to the root according to (19)-(20).

In crisp decision tree, when an unseen new instance is matching to the decision tree, the matching output of the decision tree is an exact class because only one rule matches the instance. While a new instance is matching to a fuzzy decision tree, the matching output is not a certain class label but a vector, each element of which represents the membership degree of the instance belonging to the corresponding class, respectively.

Definition 4. *Let T be a fuzzy decision tree trained well, s be a new instance of which the class information is unknown. Matching the instance to the fuzzy decision tree T, we obtain a fuzzy set $\pi = (\pi_1, \pi_2, \ldots, \pi_L)$, in which each component represents the membership degree of s belonging to the corresponding class. Then, the estimated ambiguity of s is defined as*

$$EA(s) = Ambiguity(\pi), \tag{21}$$

where $Ambiguity(\pi)$ is given in (3).

3.2 Analysis on Samples with Maximal Ambiguity

Usually, sample selection aims to find those informative samples and add them to the training set to improve the performance of the current learner. Many existing sample

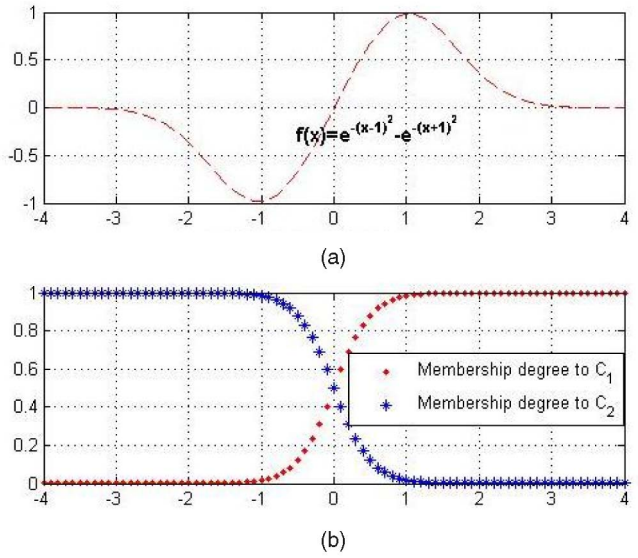

Fig. 4. A simple model of two binary classification problem. (a) Classification function. (b) Membership degrees.

selection algorithms are to select the misclassified samples, which are based on the idea that those misclassified samples are more helpful than those samples which are correctly classified regarding the improvement of the learning accuracy of the learner.

This idea can be extended to the uncertainty environment. The probably misclassified samples are usually in the vicinity of the decision boundary, which are difficult to classify by using the current learner. Thus, in our study, we think of that the samples which are with more classification ambiguity can provide more information to the learner.

To analyze intuitively the characteristics of the samples with maximal ambiguity, we take a simple demonstration of classification problem.

Consider a binary classification problem. Suppose that an instance is a point on the x-axis and its class label is determined by the function defined as

$$f(x) = e^{-(x-1)^2} - e^{-(x+1)^2}. \tag{22}$$

If $f(x) \geq 0$, the point will be classified to the first class with membership degree $C_1(x)$ which is evaluated by

$$C_1(x) = \frac{e^{-(x-1)^2}}{e^{-(x+1)^2} + e^{-(x-1)^2}}. \tag{23}$$

If $f(x) < 0$, the point will be classified to the second class with membership degree $C_2(x)$ which is evaluated by

$$C_2(x) = \frac{e^{-(x+1)^2}}{e^{-(x+1)^2} + e^{-(x-1)^2}}. \tag{24}$$

Fig. 4 gives the intuitive model of the binary classification problem.

The Fig. 4a gives the points distribution and corresponding classification function; Fig. 4b shows the membership degree of the points to every class. Clearly, we can get that $x = 0$ is the decision boundary of the classification problem. When $x > 0$, the membership degree of the point to the first class is bigger than that to the second class; thus, it will be classified to C_1; when $x < 0$, the membership degree of the point to the first class is smaller than that to the second class; thus, it will be classified to C_2.

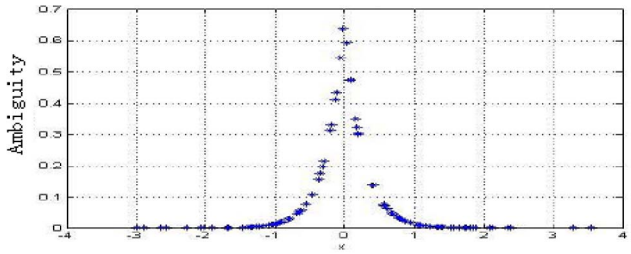

Fig. 5. Ambiguity of the points.

According to the membership degree function provided by (23) and (24), we get the ambiguity shown in Fig. 5.

It is clear to see that the points near the boundary $x = 0$ are with more ambiguity than those which are far away from $x = 0$. Usually, it is considered that the boundary points is easier to be misclassified by the learner than those far away from the boundary. And thus the boundary points are considered to be able to provide more information for current learner. Therefore, the sample with maximal classification ambiguity should be informative.

3.3 Our Proposed Algorithm

In this section, we will describe our proposed sample selection algorithm—maximum-ambiguity-based sample selection in fuzzy decision tree induction. The basic outline of this sample selection algorithm is described in Fig. 6.

First, we randomly select a certain number of instances from original data set and submit them to experts for labeling. The labeled set is considered as the initial training set. Then, we build a fuzzy decision tree using the training set and predict the unlabeled instances using the currently built decision tree. According to the prediction results (Estimated Ambiguity), we select one or some instances for annotation by domain experts. Finally, we add the selected instance(s) to the training set. The procedure will repeat several times until the number of the selected samples is up to the predefined threshold.

Next, we will give the detailed description of our proposed sample selection algorithm.

Step 1: Data partitions. Each data set is divided into three parts: training set, instance pool (short for pool), and testing set. Training set is used to build a classifier/learner which is used to select next instance; instance pool is a set of unlabeled instances which provide candidate instances for the learner to select; testing set is used to test the performance of the current classifier. In our experiments, we choose one from the fivefolds as testing set, one fold of

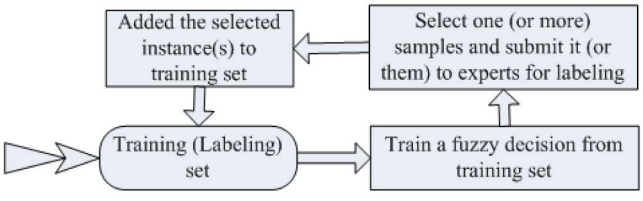

Fig. 6. Framework of our proposed sample selection algorithm.

TABLE 1
A/R Sampling (Random Selection Method)

Input : Pool $(e_1, e_2, \cdots, e_n)$, the degree of each sample to be selected: $w = (w_1, w_2, \cdots, w_n)$, where n is the number of instances in the pool.
Output the selected instance e^*
Process:
Step 1: Each element in w is divided by the sum of all the elements : $w' = \frac{1}{\sum_{i=1}^{n} w_i}(w_1, w_2, \cdots, w_n)$.
Step 2: Tag the border lines between adjacent sub regions. For example, the border between the first and second subregions is $\frac{w_1}{\sum_{i=1}^{n} w_i}$, and the border between the k-th and (k+1)-th subregions is $\sum_{j=1}^{k} w_j / \sum_{i=1}^{n} w_i$
Step 3: Produce a double data between 0 and 1 randomly. If the data is bigger than $\sum_{j=1}^{k} w_j / \sum_{i=1}^{n} w_i$ and smaller than or equal to $\sum_{j=1}^{k+1} w_j / \sum_{i=1}^{n} w_i$, then the k-th instance is selected. We use e^* to denote the selected instance.
Step 4: Label the class for e^* by experts and remove it from the pool.
Step 5: Output e^* with its label.

the remaining as the training set, and the others as the instance pool.

Step 2: Training a fuzzy decision tree by Min-A [40]. In our experiments, each attribute is discretized into two values by Kohonet's feature maps algorithm [40], and then each attribute is fuzzified into two linguistic terms by Triangular fuzzification method [40]. The final data set is the 0.45-strong set of the fuzzified data sets, which means that the cut level is 0.45. In the growing of the decision tree, the truth level threshold is set as 0.85, which means that a leaf-node is produced when the classification accuracy of a node is bigger than or equal to 0.85.

Step 3: Estimating the memberships of each instance in the pool to each class by using the newly built fuzzy decision tree and getting its classification ambiguity.

Step 4: Selecting the instance with the maximum classification ambiguity to label. Then, moving it to the training set from the pool.

Step 5: If the selected samples are less than the predefined size, then select next instance following the steps Step 2-4; otherwise, train a decision tree using the labeled samples and test the tree using testing set.

3.4 Some Notes

Many sample selection algorithms such as IBL, CNN, and their extensions could not handle the problem because their selection mechanisms are associated with the class labels of the samples to be selected. Their selection results are directly depended on the class labels of the samples. Thus, these algorithms could just condense the data set but could not reduce labeling cost.

Different from these algorithms, pool-based active learning method not have to know the class label of the instance before it is added to the training set, i.e., during the selection procedure, there is no need to label all the instances in the selection pool. Thus, the experts just need to label the selected samples when they are added to the training set, which is just a small part of the whole samples in the data set.

The main advantage of sample selection is that experts only need to label a part of samples (not all samples) and the labeling is usually cost. If labeling all samples is easy without cost, this advantage will vanish.

Compared with random selection mechanism, our proposed sample selection algorithm selects the samples with maximal classification ambiguity. It avoids labeling useless samples such as those which can be correctly classified without doubt. The samples with maximal classification ambiguity are usually in the neighborhood of the decision boundary and are considered that they could provide more information and make a more exact decision boundary. Thus, our proposed method could get a more representative and smaller training set than random selection method. The experimental results on databases give a convincing evidence.

Compared with the existing work [44], [45], there are the following main similarity and difference between them and our work: 1) Regarding class distribution information, Maytal et al. [44], [45] used the class probability estimation while our work used the possibility distribution. The difference between probability and possibility is given in Section 2.4, paragraph 3; 2) Regarding the sample selection mechanism, Maytal et al. [44], [45] are based on variance of probability estimation while our work is based on the ambiguity; 3) Both belong to a general methodology: uncertainty sampling [20].

3.5 Experimental Results

Generally, with respect to our proposed approach, there is no essential difference between the fuzzy and crisp label data sets, since the uncertainty comes from an estimated probability or possibility distribution, does not come from the original class labels. Thus, we conduct our following experiments on some UCI [46] benchmark databases with crisp labels, which is a degenerate case of our setting.

We experimentally compare three selection methodologies: Maximum Ambiguity-Based Sample Selection (i.e., our selection method), uncertainty sampling [20] and random selection method in terms of the following five aspects:

1. the number of leaf nodes;
2. the number of nodes;
3. average depth of the tree;
4. classification accuracy on pool (called pool accuracy); and
5. classification accuracy on the testing set (called testing accuracy).

We adopt the Acceptance/Rejection Sampling (short for A/R or AR) [43] to select samples randomly, which is the idea of roulette. The algorithm description is shown in Table 1.

The experiments are conducted on the some selected benchmark UCI databases with continuous attributes (http//archive.ics.uci.edu/ml/). The features of these databases are summarized in Table 2. The following is the analysis on experimental results.

TABLE 2
Features of Some UCI Databases

Databases	Instances	Attributes	Data Type	Classes	Class Distribution	Missing Values
Glass	214	9	Real	6	70/76/17/13/9/30	None
Iris	150	4	Real	3	50/50/50	None
Wine	178	13	Real	3	59/71/48	None
Ecoli	336	7	Real	8	143/77/52/35/20/5/2/2	None
Wdbc	569	31	Real	2	357/212	None
Breast	699(683)	9	Integer	2	458(444)/241(239)	16
Ionosphere	351	34	Real	2	225/126	None
Haberman	306	3	Integer	2	225/81	None
Transfusion	748	4	Real	2	178/560	None
Bupa	354	6	Integer	2	145/200	None
Sonar	208	60	Real	2	111/97	None
Yeast	1484	8	Real	10	463/429/244/163/51/44/37/30/20/5	None
Waveform	5000	21	Real	3	1657/1647/1696	None
Spambase	4601	57	Real	2	2788/1813	None
Segmentation	2310	19	Real	7	330/330/330/330/330/330/330	None
Wine Quality-White	4898	11	Real	7	5/175/880/20/163/1457/2198	None
Wine Quality-Red	1599	11	Real	6	18/199/638/10/53/681	None

Note: The datum 699 in "699(683)" about Breast denotes the number of records in original database, while 683 denotes the number of records after removing some records with missing values, and so are "458(444)"and "241(239)".

First, we explore the change tendencies of the performances of the trees as more and more selected samples are added to the training set, and compare the trees trained from the same number of samples selected by using different selection methods. When all samples in the pool are added to the training set, the trained tree is called a pool tree. The information of pool trees on selected databases is listed in Table 3.

As an example, we then analyze the experimental results on Glass, which is a database containing multiclass with unbalance distribution. In the experiments on Glass, we select 120 samples iteratively using our proposed method (MABSS) and the random selection, respectively. The average experimental results are depicted in Fig. 7.

In Fig. 7, the horizontal axis is the number of selected samples, and the vertical axis is the corresponding measurements. The red curves are the experimental results using our method, and the black curves are the results of random selection method. The green lines are the values of the pool trees.

Fig. 7a shows the testing accuracies of the trees trained from samples selected by MABSS and random selection methods. Clearly, we can see from Fig. 1a that, the testing accuracy of the random selection method is increasing all the time as more and more samples are added into the training set. This fact coincides with the idea that more training samples, and higher prediction ability. But the red curves ascend gradually initially, then descend and finally converge to the testing accuracy of pool tree. The reason why the testing accuracy of our method increases at early stage and decreases at late stage is that

1. the instances to be selected are usually insufficiently many;
2. more representative instances exist in the pool at early stage;

TABLE 3
Information of Pool Trees

Databases	Leaf Nodes	Nodes	Avg. Depth	Pool Accuracy	Testing Accuracy
Glass	27.6200	57.9600	5.7549	0.6676	0.5305
Iris	3.1000	5.2000	1.7218	0.7553	0.7580
Wine	6.3200	11.8200	2.8171	0.8892	0.8654
Ecoli	17.3600	34.6000	4.8200	0.8264	0.7816
Wdbc	2.0000	3.0000	1.0000	0.9057	0.9004
Breast	2.0000	3.0000	1.0000	0.8914	0.8869
Ionosphere	3.6100	6.3000	2.0041	0.8862	0.8839
Haberman	3.8000	6.6000	2.0136	0.7549	0.7354
Transfusion	4.7800	9.0600	2.5390	0.7662	0.7612
Bupa	21.8100	42.7600	4.8904	0.6435	0.5823
Sonar	30.4500	65.960	7.1755	0.9163	0.6988
Yeast	53.0800	109.8700	6.6715	0.5307	0.4917
Waveform	1417	2956.6	14.2	0.9000	0.7500
Spambase	271.2455	804.5909	28.6000	0.8439	0.8223
Segmentation	76.0500	190.0750	13.2835	0.8168	0.8105
Wine Quality-White	184.6800	406.7900	9.7200	0.5364	0.5013
Wine Quality-Red	189.6900	396.7800	9.0120	0.6544	0.5748

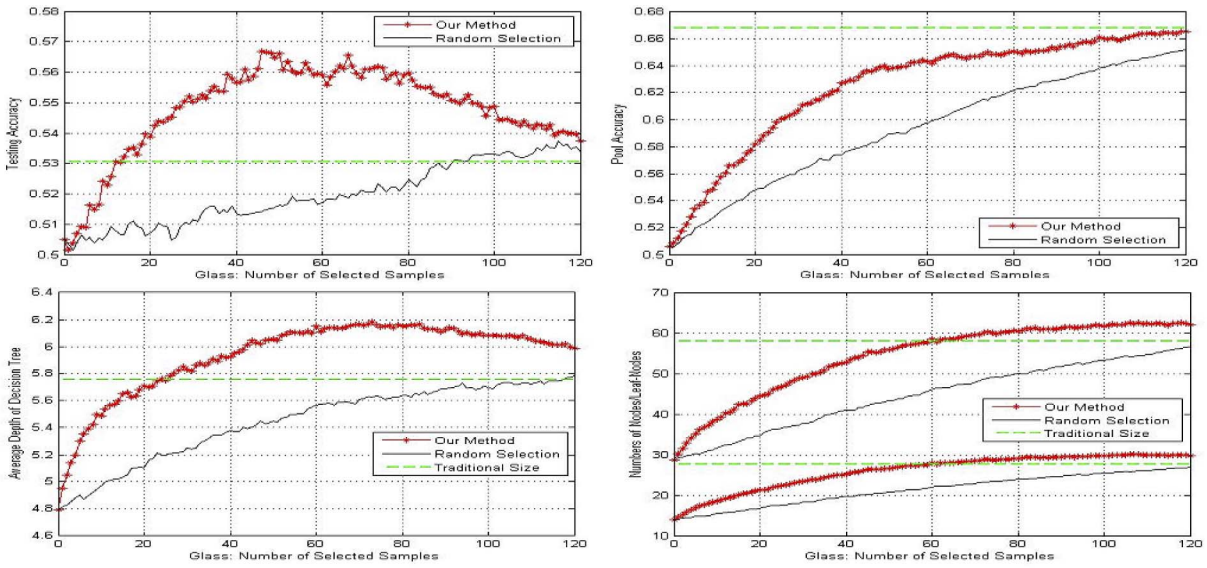

Fig. 7. Change tendencies of the trees as more samples are added to training set using MABSS and random selection method on Glass.

3. our method effectively selects the representative and removes the similar samples;
4. little sample exists at late stage and therefore samples selected by our method (MABSS) may be with little representativeness, which lead to a little decrease of testing accuracy.

Generally, if the selected samples to be selected are sufficiently many, at the early stage of selection, the averaged performance gradually increases with the sample added, but when the exact true model is approximately found, the new added samples will not significantly influence the performance. This phenomenon indicates that the selected samples using MABSS could indeed select representative samples.

From another point of view, to get a predefined testing accuracy, the selected samples by using our method are less than using random selection method. For example, to get to the level of the testing accuracy (samples in the pool as training set, about 171 samples), we just need to select around 10 samples (25 percent samples as training set, about 53 samples) by using our selection method, while about 90 samples will be selected (62 percent samples as training set, about 133 samples) when using random selection methodology.

Similar to Fig. 7a, in Fig. 7b, when more and more samples are selected to add to the training set, the pool accuracies of the trees increase gradually and finally arrive in the training accuracy of the pool tree.

Figs. 7c and 7d describe the average depth of the tree and the number of nodes/leaf nodes, respectively. From Figs. 7c and 7d, we can see that the sizes of the trees become larger when more and more samples are added into the training set and the size of the tree trained by the samples selected by using our method is a little larger than by using random selection method.

Then, we write down the testing accuracies when the selected numbers are 30, 40, 50, 60, 70, 80, 100, 120, respectively. They are shown in Table 4.

Based on Table 4, the significant level and confidence interval are applied in the analysis. It is found that the range abilities are less than 0.005. To verify the differences between the two methods, the statistical significance testing is used. We get the conclusion that the testing accuracies of our method are 3.5-4.3 percent higher than random selection when the number of selected samples is between 30 and 80.

We further analyze the experimental results on a larger database. Fig. 8 intuitively illustrates the experimental results on Spambase.

It is the same with Fig. 7, in Fig. 8, the horizontal axis is the number of selected samples, and the vertical axis is the corresponding measurements. The red curves are the experimental results using our method, and the black curves are the results of random selection method. The green lines are the corresponding values of the pool trees.

TABLE 4
Statistical Information of Testing Accuracies on Glass

No. of Samples	Selection Type	$\bar{X}$	S^2	n
30	MABSS	0.5501	0.0191^2	20
	Rand	0.5116	0.0175^2	20
	US	0.5313	0.0189^2	20
40	MABSS	0.5566	0.0185^2	20
	Rand	0.5135	0.0158^2	20
	US	0.5342	0.0135^2	20
50	MABSS	0.5660	0.0164^2	20
	Rand	0.5164	0.0199^2	20
	US	0.5389	0.0178^2	20
60	MABSS	0.5590	0.0145^2	20
	Rand	0.5171	0.0167^2	20
	US	0.5403	0.0168^2	20
70	MABSS	0.5609	0.0163^2	20
	Rand	0.5206	0.0152^2	20
	US	0.5432	0.0156^2	20
80	MABSS	0.5598	0.0170^2	20
	Rand	0.5247	0.0129^2	20
	US	0.5489	0.0189^2	20
100	MABSS	0.5488	0.0181^2	20
	Rand	0.5330	0.0187^2	20
	US	0.5464	0.0179^2	20
120	MABSS	0.5372	0.0130^2	20
	Rand	0.5339	0.0188^2	20
	US	0.5368	0.0158^2	20

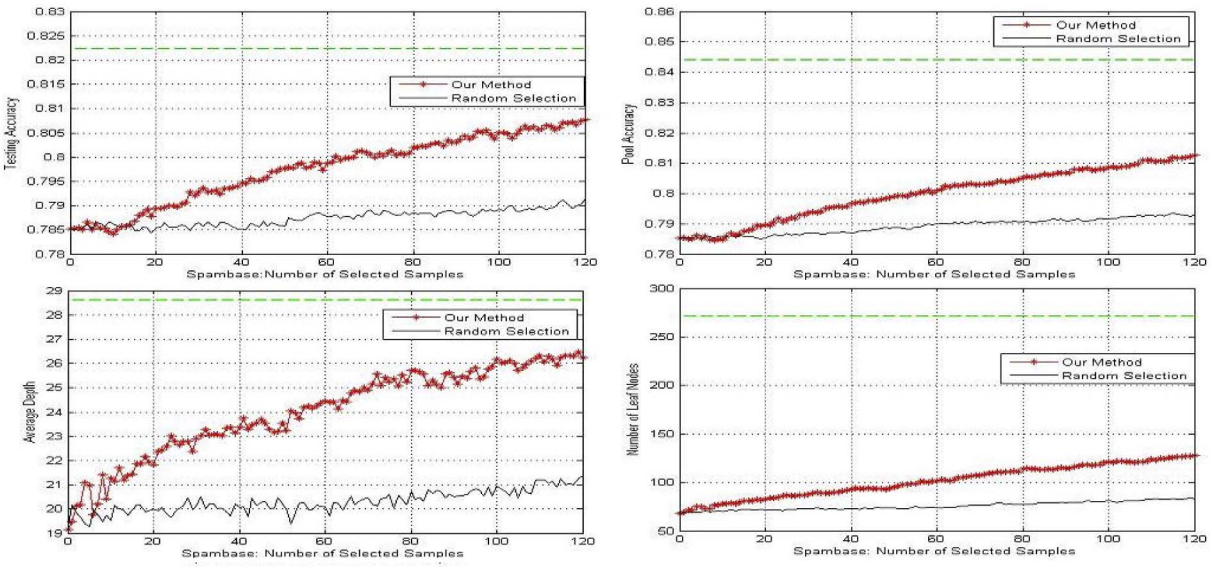

Fig. 8. Experimental results on Spambase.

From Fig. 8, we can see that the testing accuracies of the trees trained from samples selected by MABSS and random selection methods are gradually increase when more and more samples are added into the training set. Different from Fig. 7a, the red curves in Fig. 8a are always increase during the whole process and there is no descent stage. This fact is coincide with our assumption that our method could effectively select the representative samples if there exist sufficient instances and sufficient representative instances to be selected. One reason that both of the two curves of the two testing accuracies are below the testing accuracy of the pool tree is that the training set of the two trees are far less than the training set of the pool tree. (The proportion is about 580:3680.) Another reason is that the model for Spambase is the most complex among the data sets, so more training samples are required to estimate that more complex model.

Finally, we compare the testing accuracies and the sizes of the trees trained from samples selected by using the three methods when the number of selected samples is 60, and the statistical results are shown in Table 5 and Fig. 9.

Table 5 shows a comparative result about the testing accuracy. On all databases in Table 5, our method has a better testing accuracy than the other two methods, but is not always better than the pool tree. The reason is again explained as that our method can indeed select the representative samples if the instances in the pool are sufficient and the pool tree has a much bigger training set.

From Figs. 9a and 9b, we can see that, the size of the tree built by the samples selected by our method is a little larger than random selection in terms of the number of leaf nodes and the average depth of the tree, which means that the learned knowledge is more complete. The trees built on the samples selected by our method have less leaf nodes than pool trees especially when the pool trees are large such as on Yeast. Moreover, Fig. 7 shows that, after a peak, the testing accuracy on the small database decreases with the increasing

TABLE 5
Testing Accuracy of Three Sample Selection Methods

Databases	Rand (Mean/Var)	MABSS	Uncertainty Sampling	Pool Tree
Glass	$0.5171/0.0167^2$	$0.5590/0.0145^2$	$0.5403/0.0168^2$	0.5305
Iris	$0.7572/0.0065^2$	$0.7789/0.0175^2$	$0.7589/0.0118^2$	0.7580
Wine	$0.8667/0.0123^2$	$0.8843/0.0093^2$	$0.8697/0.103^2$	0.8654
Ecoli	$0.7636/0.0120^2$	$0.7722/0.0150^2$	$0.7589/0.0128^2$	0.7816
Wdbc	$0.8984/0.0037^2$	$0.9229/0.0024^2$	$0.8712/0.0049^2$	0.9004
Breast	$0.8818/0.0052^2$	$0.9330/0.0104^2$	$0.9102/0.0089^2$	0.8869
Ionosphere	$0.8573/0.0066^2$	$0.8674/0.0062^2$	$0.8913/0.099^2$	0.8839
Haberman	$0.7312/0.0067^2$	$0.7372/0.0051^2$	$0.7335/0.0068^2$	0.7354
Transfusion	$0.7628/0.0028^2$	$0.7627/0.0021^2$	$0.7815/0.0056^2$	0.7612
Bupa	$0.5580/0.0107^2$	$0.5409/0.0100^2$	$0.5448/0.0108^2$	0.5823
Sonar	$0.6929/0.0175^2$	$0.6948/0.0172^2$	$0.6932/0.0119^2$	0.6988
Yeast	$0.4583/0.0066^2$	$0.4664/0.0053^2$	$0.4486/0.0156^2$	0.4917
Waveform	$0.6892/0.0094^2$	$0.6930/0.0142^2$	$0.6901/0.0135^2$	0.7500
Spambase	$0.7887/0.0139^2$	$0.8016/0.0126^2$	$0.7983/0.0115^2$	0.8223
Segmentation	$0.7794/0.0265^2$	$0.7839/0.0167^2$	$0.7809/0.0216^2$	0.8105
Wine Quality-White	$0.4458/0.0173^2$	$0.4524/0.0141^2$	0.4518/0.01692	0.5013
Wine Quality-Red	$0.4942/0.0214^2$	$0.5056/0.0164^2$	$0.4987/0.0192^2$	0.5748

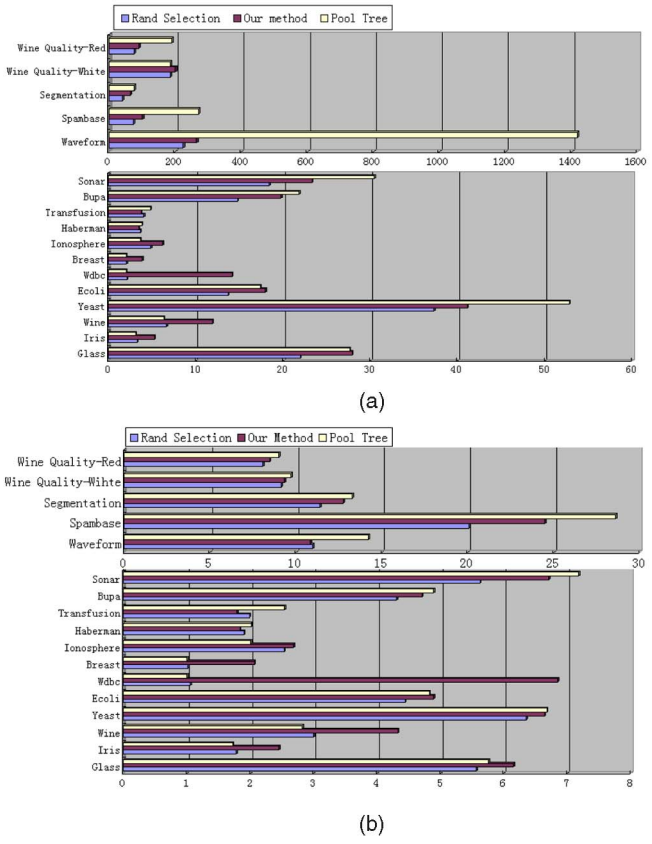

Fig. 9. The average tree size on Databases of UCI. (a) Average number of leaf nodes. (b) Average depth.

of number of samples and so does the depth of the tree. It implicitly indicates that the tree is not large enough.

4 Further Theoretical Analysis on Our Proposed Maximum-Ambiguity-Based Sample Selection in Fuzzy Decision Tree Induction

For simplicity, we suppose that the problem is two-class case with conditional attributes $A_1, A_2, \ldots, A_m$ (shown in Table 6). Each attribute $A_i (1 \le i \le m)$ takes two values: A_{i1} and A_{i2}. The decision attribute C also takes two values: C_1 and C_2. The sample space is denoted as $S_N = \{1, 2, \ldots, N\}$. Each vector $(a_{11}^{(i)}, a_{12}^{(i)}, a_{21}^{(i)}, a_{22}^{(i)}, \ldots, a_{m1}^{(i)}, a_{m2}^{(i)}, c_1^{(i)}, c_2^{(i)})$ is a sample for each $i (1 \le i \le N)$. Each $a_{ij}^{(k)}$ or $c_j^{(k)}$ is a real number in the interval [0, 1] for all $i (1 \le i \le m), j (1 \le j \le 2), k (1 \le k \le N)$. $A_{11}, A_{12}, A_{21}, A_{22}, \ldots, A_{m1}, A_{m2}, C_1, C_2$ are fuzzy subsets defined on S_N. For example, $A_{11} = (a_{11}^{(1)}, a_{11}^{(2)}, \ldots, a_{11}^{(N)})$ and $C_1 = (c_1^{(1)}, c_1^{(2)}, \ldots, c_1^{(N)})$.

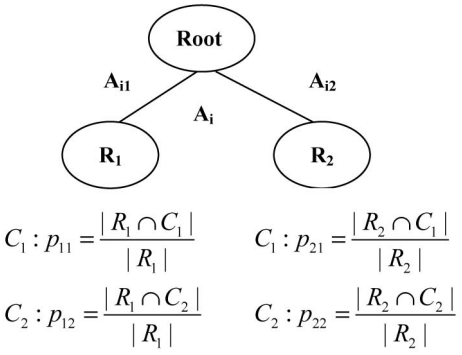

Fig. 10. A simple model of fuzzy decision tree.

4.1 Decision Boundary of Leaf Nodes

For simplicity, we consider the case shown in Fig. 10. R_1 and R_2 are two leaf nodes expanding from root according to expanded attribute $A_i (1 \le i \le m)$. It means that R_1 and R_2 are two fuzzy subsets defined on S_N, i.e., $R_1 = \{a_{i1}^{(1)}, a_{i1}^{(2)}, \ldots, a_{i1}^{(N)}\}$, $R_2 = \{a_{i2}^{(1)}, a_{i2}^{(2)}, \ldots, a_{i2}^{(N)}\}$.

Let $p_{ij} = |R_i \bigcap C_j| / |R_i|$ be the confidence degree of $R_i \Rightarrow C_j$ where the symbol $\Rightarrow$ means the operation of implication.

It is noted that $p_{11} > p_{12}$ and $p_{21} < p_{22}$, respectively, imply that R_1 can be labeled as C_1 with confidence p_{11} and R_2 can be labeled as C_2 with confidence p_{22}.

If a new instance x (numbered N+1) is given to match the decision tree, the membership degrees of the instance belonging to R_1 and R_2 will be $A_{i1}(x)$ and $A_{i2}(x)$, respectively.

According to the matching rules,

if $A_{i1}(x) \cdot p_{11} > A_{i2}(x) \cdot p_{22}$, then x is classified to C_1 with confidence p_{11};

TABLE 6
Sample Space

	A_1		A_2		...	A_m		C	
ID	A_{11}	A_{12}	A_{21}	A_{22}	...	A_{m1}	A_{m2}	C_1	C_2
1	$a_{11}^{(1)}$	$a_{12}^{(1)}$	$a_{21}^{(1)}$	$A_{22}^{(1)}$	...	$a_{m1}^{(1)}$	$a_{m2}^{(1)}$	$c_1^{(1)}$	$c_2^{(1)}$
2	$a_{11}^{(2)}$	$a_{12}^{(2)}$	$a_{21}^{(2)}$	$a_{22}^{(2)}$	...	$a_{m1}^{(2)}$	$a_{m2}^{(2)}$	$c_1^{(2)}$	$c_2^{(2)}$
	...		...		...		...		
N	$a_{11}^{(N)}$	$a_{12}^{(N)}$	$a_{21}^{(N)}$	$a_{22}^{(N)}$	...	$a_{m1}^{(N)}$	$a_{m2}^{(N)}$	$c_1^{(N)}$	$c_2^{(N)}$
x	$A_{11}(x)$	$A_{12}(x)$	$A_{21}(x)$	$A_{22}(x)$	...	$A_{m1}(x)$	$A_{m2}(x)$	$c_1^{(N+1)} =?$	$c_2^{(N+1)} =?$
								$C_1(x)$	$C_2(x)$

Note: $c_1^{(N+1)}$ and $c_2^{(N+1)}$ will be given by domain experts, which denote the real membership degrees with that the instance x belongs to C_1 and C_2, respectively; $C_1(x)$ and $C_2(x)$ are the prediction results for the instance x by using the fuzzy decision tree trained from the training set S_N.

if $A_{i1}(x) \cdot p_{11} < A_{i2}(x) \cdot p_{22}$, then x is classified to C_2 with confidence p_{22};

if $A_{i1}(x) \cdot p_{11} = A_{i2}(x) \cdot p_{22}$, we separately handle it.

Noting that $A_{i1}(x) + A_{i2}(x) = 1$ and then the inequation $A_{i1}(x) \cdot p_{11} > A_{i2}(x) \cdot p_{22}$ is equivalent to $A_{i1}(x) > p_{22}/(p_{11} + p_{22})$, which implies that the instance x will be classified to C_1 when $A_{i1}(x) > p_{22}/(p_{11} + p_{22})$.

Similarly, the inequation $A_{i1}(x) \cdot p_{11} < A_{i2}(x) \cdot p_{22}$ is equivalent to $A_{i1}(x) < p_{22}/(p_{11} + p_{22})$, which implies that the instance x will be classified to C_2 when $A_{i1}(x) < p_{22}/(p_{11} + p_{22})$.

Noting that the value $p_{22}/(p_{11} + p_{22})$ plays a role of threshold, we define

$$BN = \frac{p_{22}}{p_{11} + p_{22}} \tag{25}$$

as the decision boundary of two leaf nodes R_1 and R_2 in the fuzzy decision tree.

4.2 The Changes of the Fuzzy Decision Tree when Adding a Sample to the Training Set

When a new sample is added to the training set and the tree is retrained, the fuzzy sets denoting leaf nodes will change, and so do the relative frequencies of the leaf nodes. Therefore, the classification ambiguities of the leaf nodes will change, and so does the ambiguity of the fuzzy decision tree.

We now focus on the analysis of the changes of the fuzzy decision tree when adding a new sample to the training set.

When a new sample $x = (A_{11}(x), A_{12}(x), A_{21}(x), \ldots, A_{m1}(x), A_{m2}(x), c_1^{(N+1)}, c_2^{(N+1)})$ is added to the current training set S_N, the new decision tree will be again built from the new training set S_{N+1}.

Suppose that R_1' and R_2' are two fuzzy subsets representing the left and right leaf node of the new fuzzy decision tree, respectively. Then, $R_1' = (R_1, A_{i1}(x))$, $R_2' = (R_2, A_{i2}(x))$, $C_1' = (C_1, c_1^{(N+1)})$, $C_2' = (C_2, c_2^{(N+1)})$.

Therefore,

$$\begin{aligned} p_{11}' &= \frac{\left|\left(R_1, A_{i1}(x)\right) \bigcap \left(C_1, c_1^{(N+1)}\right)\right|}{\left|\left(R_1, A_{i1}(x)\right)\right|} \\ &= \frac{\left|R_1 \bigcap C_1\right| + \left(A_{i1}(x) \bigwedge c_1^{(N+1)}\right)}{\left|(R_1, A_{i1}(x))\right|} \\ &= \frac{|R_1 \bigcap C_1|}{|R_1|} \cdot \frac{|R_1|}{|R_1| + A_{i1}(x)} + \frac{min\{A_{i1}(x), c_1^{(N+1)}\}}{|R_1| + A_{i1}(x)} \\ &= p_{11} \cdot \frac{|R_1|}{|R_1| + A_{i1}(x)} + \frac{min\left\{A_{i1}(x), c_1^{(N+1)}\right\}}{|R_1| + A_{i1}(x)}. \end{aligned} \tag{26}$$

Similarly,

$$p_{12}' = p_{12} \cdot \frac{|R_1|}{|R_1| + A_{i1}(x)} + \frac{min\left\{A_{i1}(x), c_2^{(N+1)}\right\}}{|R_1| + A_{i1}(x)} \tag{27}$$

$$p_{21}' = p_{21} \cdot \frac{|R_2|}{|R_2| + A_{i2}(x)} + \frac{min\left\{A_{i2}(x), c_1^{(N+1)}\right\}}{|R_2| + A_{i2}(x)} \tag{28}$$

$$p_{22}' = p_{22} \cdot \frac{|R_2|}{|R_2| + A_{i2}(x)} + \frac{min\left\{A_{i2}(x), c_2^{(N+1)}\right\}}{|R_2| + A_{i2}(x)}. \tag{29}$$

If the sample x belongs to C_1, then $c_1^{(N+1)} = 1, c_2^{(N+1)} = 0$, and (14)-(17) can be simplified as

$$p_{11}' = p_{11} \cdot \frac{|R_1|}{|R_1| + A_{i1}(x)} + \frac{A_{i1}(x)}{|R_1| + A_{i1}(x)} \tag{30}$$

$$p_{12}' = p_{12} \cdot \frac{|R_1|}{|R_1| + A_{i1}(x)} \tag{31}$$

$$p_{21}' = p_{21} \cdot \frac{|R_2|}{|R_2| + A_{i2}(x)} + \frac{A_{i2}(x)}{|R_2| + A_{i2}(x)} \tag{32}$$

$$p_{22}' = p_{22} \cdot \frac{|R_2|}{|R_2| + A_{i2}(x)}. \tag{33}$$

Equations (30)-(33) give the relationship between classification frequencies in leaf nodes before and after adding a new sample to the training set under the assumption that the sample x belongs to C_1. Similarly, we can consider the case of x belonging to C_2. Here, we do not discuss it further.

4.3 Changes of Classification Ambiguity of Leaf Nodes

To analyze the changes of classification ambiguities of the leaf nodes, we first consider the relationships between relative frequencies and classification ambiguities of the leaf nodes.

Consider a leaf node R including L classes. Let p_1, $p_2, \ldots, p_L$ be the relative frequencies of the leaf node and $p_1 > p_2 > \cdots > p_L$. Suppose that the left leaf node changes from R_1 to R_1' and the right leaf node changes from R_2 to R_2'. Then, we have the following two theorems.

Theorem 1. *$Ambiguity(R)$ will decrease when p_1 increases and all the others remain unchanged; $Ambiguity(R)$ will increase when any $p_i (1 < i \le L)$ increases and all the others remain unchanged.*

Proof.

$$\begin{aligned} Ambiguity(R) &= \sum_{i=1}^{L} (p_i - p_{i+1}) \ln i \\ &= \left(\frac{p_1}{p_1} - \frac{p_2}{p_1}\right)\ln 1 + \left(\frac{p_2}{p_1} - \frac{p_3}{p_1}\right)\ln 2 \\ &\quad + \cdots + \left(\frac{p_L}{p_1} - \frac{p_{L+1}}{p_1}\right)\ln L \\ &= \ln 1 + p_2/p_1(\ln 2 - \ln 1) + p_3/p_1(\ln 3 - \ln 2) \\ &\quad + \cdots + p_L/p_1(\ln L - \ln(L-1)) \\ \frac{\partial Ambiguity(R)}{\partial p_1} &= -\frac{p_2}{p_1^2}(\ln 2 - \ln 1) - \frac{p_3}{p_1^2}(\ln 3 - \ln 2) \\ &\quad - \cdots - \frac{p_L}{p_1^2}(\ln L - \ln(L-1)) < 0 \\ \frac{\partial Ambiguity(R)}{\partial p_i} &= \frac{1}{p_1}(\ln i - \ln(i-1)) > 0 \quad (2 \le i \le L). \end{aligned}$$

□

Theorem 2. *Ambiguity(R) will not change when $p_1, p_2, \ldots, p_L$ change in the same proportions.*

From (30)-(31), we can see that the computation of changes of p_{11} and p_{12} can be divided into two steps: first, p_{11} and p_{12} multiply the same factor $\frac{|R_1|}{|R_1|+A_{i1}(x)}$, and then add $\frac{A_{i1}(x)}{|R_1|+A_{i1}(x)}$ to p_{11}. In this case, the classification ambiguity of the left leaf node of the fuzzy decision tree will decrease.

Similarly, from (32)-(33), the computation of changes of p_{21} and p_{22} can also be divided into two steps: first, p_{21} and p_{22} multiply the same proportion $\frac{|R_2|}{|R_2|+A_{i2}(x)}$, and then add $\frac{A_{i2}(x)}{|R_2|+A_{i2}(x)}$ to p_{21}. Therefore, the classification ambiguity of the right leaf node of the fuzzy decision tree will increase under the assumption $p'_{21} < p'_{22}$.

Clearly, when adding a new sample that belongs to C_1 to the training set S_N, compared with the original fuzzy decision tree trained from S_N, the classification ambiguity of the left node of the new fuzzy decision tree trained from S_{N+1} will decrease, while the classification ambiguity of the right node of the new fuzzy decision tree will increase under the assumption that $p'_{21} < p'_{22}$.

4.4 Change Rate of the Classification Ambiguities of Leaf Nodes

According to Theorems 1 and 2, when the relative frequencies of a leaf-node multiply the same proportions, the classification ambiguity of the leaf node will not change. When one of the relative frequencies increases and the others remain unchanged, the classification ambiguity will decrease or increase.

Therefore, in order to analyze the change rate of the classification ambiguity for a leaf node, we just need to analyze the ratio of the relative frequencies of the leaf node.

We first pay attention to p_{11}/p_{12}.

When adding a sample belonging C_1 to the training set, p_{11} and p_{12} will multiply the same factor, and then add $\frac{A_{i1}(x)}{|R_1|+A_{i1}(x)}$ to p_{11}.

For simplicity, we denote that $y = A_{i1}(x), a = |R_1|$ and $f(y) = \frac{y}{a+y}$. Clearly, $a \geq y, y \in [0,1]$. Therefore,

$$\frac{df}{dy} = \frac{y}{(a+y)^2} \geq 0,\ (y \in [0,1], a \geq y) \tag{34}$$

$$\left(\frac{df}{dy}\right)' = \frac{a-y}{(a+y)^3} \geq 0,\ (y \in [0,1], a \geq y), \tag{35}$$

it is easy to check that f is an increasing concave function. When $A_{i1}(x)$ is changing from 0 to 1, $f(A_{i1}(x))$ always increases and the increase-rate increases. These imply that when adding a sample that belongs to C_1 in the training set, p_{11}/p_{12} always increases and the increase-rate also increases as the attribute value $A_{i1}(x)$ is changing from 0 to 1.

Similarly, we consider p_{21}/p_{22}.

As we known, when adding a sample belonging to C_1 to the training set, p_{21} and p_{22} will multiply the same proportion, and then add $\frac{A_{i2}(x)}{|R_2|+A_{i2}(x)}$ to p_{21}.

Noting that $A_{i1}(x) + A_{i2}(x) = 1$, we denote that $y = A_{i1}(x), b = |R_2|$ and $g(y) = \frac{1-y}{b+1-y}$. Clearly, $y \in [0,1]$. We have

$$\frac{dg}{dy} = \frac{-b}{(b+1-y)^2} \leq 0,\ (y \in [0,1]) \tag{36}$$

$$\left(\frac{dg}{dy}\right)' = \frac{-2b}{(b+1-y)^3} \leq 0,\ (y \in [0,1]). \tag{37}$$

Clearly, g is a decreasing concave function. When $A_{i1}(x)$ is changing from 0 to 1, $g(A_{i1}(x))$ always decreases and the decrease-rate decreases. These imply that when adding a sample that belongs to C_1 in the training set, p_{21}/p_{22} always decreases on the assumption that p_{22} is bigger than p_{21}. And the decrease-rate decreases as the attribute value $A_{i1}(x)$ is changing from 0 to 1.

According to the analyses above, when adding a sample belonging to C_1 in the training set based on the simplified model shown in Fig. 8, p_{11}/p_{12} will always increase and the increase-rate also increases with the change of $A_{i1}(x)$ from 0 to 1; p_{21}/p_{22} will decrease and the decrease-rate also decreases with the change of $A_{i1}(x)$ from 0 to 1. Therefore, the classification ambiguity of the left leaf node will decrease and the decrease-rate will decrease from maximum to 0, while the classification ambiguity of the right leaf node will increase and the increase-rate will increase from 0 to maximum. Similarly, when adding a sample belonging to C_2 to the training set, the classification ambiguity of the left leaf node will increase and the classification ambiguity of the right leaf node will decrease. But both of their change rates will accelerate.

4.5 The Ambiguity's Change of the Fuzzy Decision Tree

According to Definition 3, the classification ambiguity of the root is associated with the classification ambiguities of all the leaf nodes. The changes of the classification ambiguities of all the leaf nodes will result in the change of the classification ambiguity of the root, i.e., the ambiguity of the whole decision tree.

Following the analyses above, when adding a sample to the training set, the classification ambiguity of one leaf node will increase and its increase rate also increases, while the classification ambiguity of its sibling leaf node decreases and the decrease rate decreases. Therefore, there exists a point denoted by $\lambda(\lambda \in [0,1])$, at which the increase rate of the left leaf node equals to the decrease rate of the right leaf node when $A_{i1}(x)$ is changing from 0 to 1.

Therefore, we can derive the following conclusions:

1. When $A_{i1}(x)$ is in the interval $[0, \lambda)$, because the increase rate of the classification ambiguity of the right leaf node is bigger than the decrease rate of the classification ambiguity of the left leaf node, the ambiguity of the decision tree increases and the increase rate becomes slower as $A_{i1}(x)$ approaches to λ;
2. When $A_{i1}(x)$ is in the interval $(\lambda, 1]$, because the decrease rate of one leaf node is bigger than the increase rate of the other node arising from the same parent, the ambiguity of the decision tree decreases and the decrease rate becomes faster as $A_{i1}(x)$ is away from λ;

3. When $A_{i1}(x)$ equals to λ, no matter the ambiguity of the fuzzy decision tree increases or decreases, the change rate of the decision tree's ambiguity is smallest.

4.6 Summary

Sample selection method based on maximal classification ambiguity in fuzzy decision tree is to select the instance with maximal evaluated ambiguity, i.e., the instance is the one which satisfies $C_1(x) \approx C_2(x)$. According to the matching to the fuzzy decision tree for an instance, the instance with maximal evaluated ambiguity is closest to the decision boundary of the decision tree, i.e., the instance whose $A_{i1}(x)$ approaches BN as possible where BN is decision boundary defined in Section 4.1. We now analyze the relations between the selected instance(s) and the ambiguity of the fuzzy decision tree on the assumption that $\lambda = BN$.

According to the results in Section 4.1 that when $A_{i1}(x)$ is greater than BN, it will be classified to C_1; when $A_{i1}(x)$ is smaller than BN, it will be classified to C_2. Combining the results in Section 4.2, we can see that when adding an instance that is classified correctly using the current fuzzy decision tree to the training set, compared with the old decision tree, the ambiguity of the newly built the decision tree will decrease. On the contrary, when adding an instance that is misclassified, the ambiguity will increase. When adding an instance with maximal ambiguity, no matter it is classified correctly or incorrectly and no matter the ambiguity of the decision tree increases or decreases, the change of the decision tree is smallest.

Selecting samples to incrementally generate a decision tree is similar to an action in our real life. The channel of a TV is adjusted by a button. If we want to improve the TV picture by turning the button, but it is uncertain that which direction is correct. In this case, a very slight turning is the safest choice. Induction of incremental decision tree with sample selection can be regarded as adjusting the decision tree through adding instances. Selecting an instance with the maximal evaluated ambiguity can minimize the adjustment of the decision tree.

5 Conclusions

This paper proposes a sample selection method based on the maximal classification ambiguity in fuzzy decision tree induction and gives an analysis on the significance of the sample selection methodology. It selects the instance with maximal evaluation ambiguity when the instance is matching to the fuzzy decision tree. The selected instance can minimize the adjustment of the generated fuzzy decision tree and finally build a fuzzy decision tree with high performance gradually. Our numerical experiments give a sufficient evidence and support to the corresponding theoretical inference.

The following conclusions can be drawn in our study:

1. The sample selection method we proposed in this study is based on the principle of the maximal classification ambiguity to select the samples with maximal evaluated ambiguity in fuzzy decision tree induction;
2. Usually the instance selected by using our proposed method is in the vicinity of the decision boundary. The basic idea is similar to CNN, IBL, etc., which only store those misclassified instances. The instances in the vicinity of the decision boundary are regarded as the instances that are likely to be misclassified using the current fuzzy decision tree;
3. The ambiguity of the decision tree will change when an instance is added to the training set. When adding an instance that is correctly classified using the current decision tree, the ambiguity of the fuzzy decision tree will decrease. While adding an instance that is incorrectly classified, the ambiguity of the fuzzy decision tree will increase. However, when adding an instance with the maximal evaluated ambiguity, which may be classified correctly or incorrectly by using the current decision tree, no matter the ambiguity of the fuzzy decision tree increases or decreases, the adjustment of the decision tree is minimal.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (61170040), by the Natural Science Foundation of Hebei Province (F2008000635, F2012201023), by the key project of applied fundamental research of Hebei Province (08963522D), and by the plan of first 100 excellent innovative scientists of Education Department in Hebei Province.

References

[1] N. Abe and H. Mamitsuka, "Query Learning Strategies Using Boosting and Bagging," *Proc. 15th Int'l Conf. Machine Learning*, pp. 1-10, 1998.
[2] D.W. Aha, D. Kibler, and M.K. Albert, "Instance-Based Learning Algorithms," *Machine Learning*, vol. 6, pp. 37-66, 1991.
[3] F. Angiulli, "Fast Condensed Nearest Neighbor Rule," *Proc. 22nd Int'l Conf. Machine Learning: ACM Int'l Conf. Proceeding Series*, vol. 119, pp. 25-32, 2005.
[4] D. Angluin, "Queries and Concept Learning," *Machine Learning*, vol. 2, no. 4, pp. 319-342, 1988.
[5] Y. Baram, R. El-Yaniv, and K. Luz, "Online Choice of Active Learning Algorithms," *J. Machine Learning Research*, vol. 5, pp. 255-291, 2003.
[6] H. Brighton and C. Mellish, "Advances in Instance Selection for Instance-Based Learning Algorithms," *Data Mining and Knowledge Discovery*, vol. 6, pp. 153-172, 2002.
[7] C. Campbell, N. Cristianini, and A. Smola, "A Query Learning with Large Margin Classifiers," *Proc. 17th Int'l Conf. Machine Learning*, pp. 111-118, 2000.
[8] C.L. Chang, "Finding Prototypes for Nearest Neighbor Classifiers," *IEEE Trans. Computers*, vol. C-23, no. 11, pp. 1179-1184, Nov. 1974.
[9] D. Cohn, Z. Ghahramani, and M.I. Jordan, "Active Learning with Statistical Models," *Advances in Neural Information Processing Systems*, vol. 7, pp. 705-712, 1995.
[10] D. Cohn and A.R. Ladner, "Improving Generalization with Active Learning," *Machine Learning*, vol. 5, no. 2, pp. 201-221, 1994.
[11] I. Dagon and S. Engelson, "Committee-Based Sampling for Training Probabilistic Classifiers," *Proc. 12th Int'l Conf. Machine Learning*, pp. 150-157, 1995.
[12] A. Deluca and S. Termini, "A Definition of Nonprobabilistic Entropy in the Setting of Fuzzy Sets Theory," *Information and Control*, vol. 20, pp. 301-312, 1972.
[13] D. Dubois and H. Prade, "Rough Fuzzy Sets and Fuzzy Rough Sets," *Int'l J. General Systems*, vol. 17, pp. 191-209, 1990.
[14] G.W. Gates, "The Reduced Nearest Neighbor Rule," *IEEE Trans. Information Theory*, vol. IT-18, no. 3, pp. 431-433, May 1972.
[15] P.E. Hart, "The Condensed Nearest Neighbor Rule," *IEEE Trans. Information Theory*, vol. IT-14, no. 3, pp. 515-516, May 1968.

[16] R.V.L. Hartley, "Transmission of Information," *The Bell System Technical J.*, vol. 7, pp. 535-563, 1949.
[17] M. Higashi and G.J. Klir, "Measures of Uncertainty and Information Based on Possibility Distributions," *Int'l J. General Systems*, vol. 9, no. 1, pp. 43-58, 1983.
[18] V.S. Iyengar, C. Apte, and T. Zhang, "Active Learning Using Adaptive Resampling," *KDD '00: Proc. Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining*, pp. 91-98, 2000.
[19] D. Lewis and J. Catlett, "Heterogenous Uncertainty Sampling for Supervised Learning," *Proc. 17th Ann. ACM-SIGR Conf. Research and Development in Information Retrival*, pp. 148-156, 1994.
[20] D. Lewis and W.A. Gail, "A Sequential Algorithm for Training Text Classifiers," *Proc. 17th ACM Int'l Conf. Research and Development in Information Retrieval*, pp. 3-12, 1994.
[21] H. Li, "Probability Representation of Fuzzy System," *Science in China Series E: Technological Sciences*, vol. 36, no. 4, pp. 373-397, 2006.
[22] B. Liu, *Uncertainty Theory*, pp. 57-138, third ed., Springer, http://orsc.edu.cn/liu/ut.pdf, 2008.
[23] P. Melville and R.J. Mooney, "Diverse Ensembles for Active Learning," *Proc. 21th Int'l Conf. Machine Learning: ACM Int'l Conf. Proceeding Series*, vol. 69, pp. 74-74, 2004.
[24] H.T. Nguyen and A. Smeulders, "Active Learning Using Pre-Clustering," *ICML '04: Proc. 21st Int'l Conf. Machine Learning*, pp. 79-86, 2004.
[25] Z. Pawlak, "Rough Sets," *Int'l J. Information and Computer Sciences*, vol. 11, pp. 341-356, 1982.
[26] G.L. Ritter, H.B. Woodruff, S.R. Lowry, and T.L. Isenhour, "An Algorithm for a Selective Nearest Neighbor Rule," *IEEE Trans. Information Theory*, vol. IT-21, no. 6, pp. 665-669, Nov. 1975.
[27] N. Roy and A. Mccallum, "Toward Optimal Active Learning through Sampling Estimation of Error Reduction," *Proc. 18th Int'l Conf. Machine Learning*, pp. 441-448, 2001.
[28] G. Schohn and D. Cohn, "Less Is More: Active Learning with Support Vector Machines," *Proc. 17th Int'l Conf. Machine Learning*, pp. 839-846, 2000.
[29] H.S. Seung, M. Opper, and H. Sompolinsky, "Query by Committee," *Proc. Ann. Workshop Computational Learning Theory*, pp. 287-294, 1992.
[30] C.E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical J.*, vol. 27, pp. 379-423, 623-656, 1948.
[31] C. Tohompson, M.E. Califf, and R. Mooney, "Active Learning for Natural Language Parsing and Information Extraction," *Proc. 16th Int'l Conf. Machine Learning*, pp. 406-414, 1999.
[32] S. Tong and D. Koller, "Support Vector Machine Active Learning with Applications to Text Classification," *Proc. 17th Int'l Conf. Machine Learning (ICML '00)*, pp. 999-1006, 2000.
[33] S. Tong and D. Koller, "Active Learning for Parameter Estimation in Bayesian Networks," *Advances in Neural Information Processing Systems*, pp. 647-653, 2000.
[34] E.C.C. Tsang and X.Z. Wang, "An Approach to Case-Based Maintenance: Selecting Representative Cases," *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 19, pp. 79-89, 2005.
[35] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1995.
[36] X.Z. Wang, J.H. Yan, R. Wang, and C.R. Dong, "A Sample Selection Algorithm in Fuzzy Decision Tree Induction and Its Theoretical Analyses," *ICSMC '07: Proc. IEEE Int'l Conf. Systems, Man, and Cybernetics*, pp. 3621-3626, 2007.
[37] D.L. Wilson, "Asymptotic Properties of Nearest Neighbor Rules Using Edited Data," *IEEE Trans. Systems, Man and Cybernetics*, vol. 2, no. 3, pp. 408-421, July 1972.
[38] D.R. Wilson and T.R. Martinez, "Instance Pruning Techniques," *Proc. 14th Int'l Conf. Machine Learning*, pp. 403-411, 1997.
[39] D.R. Wilson and T.R. Martinez, "Reduction Techniques for Instance-Based Learning Algorithms," *Machine Learning*, vol. 38, no. 3, pp. 257-286, 2000.
[40] Y.F. Yuan and M.J. Shaw, "Induction of Fuzzy Decision Trees," *Fuzzy Sets and Systems*, vol. 69, pp. 125-139, 1995.
[41] L.A. Zadeh, "Fuzzy Sets," *Information and Control*, vol. 8, pp. 338-53, 1965.
[42] R.V.L. Hartley, "Transmission of Information," *The Bell System Technical J.*, vol. 7, no. 3, pp. 535-563, 1928.
[43] O. Frank and R. Doron, "Random Sampling from Databases: A Survey," *Statistics and Computing*, vol. 5, no. 1, pp. 25-42, Mar. 1995.
[44] S.-T. Maytal and P. Foster, "Active Sampling for Class Probability Estimation and Ranking," *Machine Learning*, vol. 54, no. 2, pp. 153-178, 2004.
[45] M. Prem, M.Y. Stewart, S.-T. Maytal, and J.M. Raymond, "Active Learning for Probability Estimation Using Jensen-Shannon Divergence," *Proc. 16th European Conf. Machine Learning (ECML)*, pp. 268-279, 2005.
[46] UCI Machine Learning Repository, http//archive.ics.uci.edu/ml/, 2011.

Xi-Zhao Wang received the PhD degree in computer science from Harbin Institute of Technology, China, in 1998. He is presently the dean and professor of the College of Mathematics and Computer Science, Hebei University, China. From September 1998 to September 2001, he served as a research fellow in the Department of Computing, Hong Kong Polytechnic University, Hong Kong. He became full professor and dean of the College of Mathematics and Computer Science in Hebei University in October 2001. His main research interests include learning from examples with fuzzy representation, fuzzy measures and integrals, neuro-fuzzy systems and genetic algorithms, feature extraction, multiclassifier fusion, and applications of machine learning. He has 160+ publications including four books, seven book chapters, and 90+ journal papers in *IEEE Transactions on PAMI/SMC/FS*, *Fuzzy Sets and Systems*, *Pattern Recognition*, etc. He has been the PI/Co-PI for 16 research projects supported partially by the National Natural Science Foundation of China and the Research Grant Committee of Hong Kong Government. He is a senior member of the IEEE (Board of Governor member in 2005, 2007-2009); the Chair of IEEE SMC Technical Committee on Computational Intelligence, an associate editor of *IEEE Transactions on SMC*, *Part B*; an associate editor of *Pattern Recognition and Artificial Intelligence*; a member of editorial board of *Information Sciences*; and an executive member of the Chinese Association of Artificial Intelligence. He was the recipient of the IEEE-SMCS Outstanding Contribution Award in 2004 and the recipient of IEEE-SMCS Best Associate Editor Award in 2006. He is the general cochair of the 2002-2009 International Conferences on Machine Learning and Cybernetics, cosponsored by IEEE SMCS. He is a distinguished lecturer of IEEE SMC Society.

Ling-Cai Dong received the bachelor's degree in software engineering from Hebei University in June 2007. She is currently working toward the postgraduate degree in the Department of Mathematics and Computer Science, Hebei University, China. Her main research interests include machine learning. She is a student member of the IEEE.

Jian-Hui Yan received the master's degree in software engineering from Hebei University in June 2007. He is currently working toward the postgraduate degree in the Department of Mathematics and Computer Science, Hebei University, China. His main research interests include machine learning.