8-2012

Using Attribute Behavior Diversity to Build Accurate Decision Tree Committees for Microarray Data

Qian Han

Wright State University - Main Campus

Guozhu Dong

Wright State University - Main Campus, guozhu.dong@wright.edu

Follow this and additional works at: <https://corescholar.libraries.wright.edu/knoesis>

Part of the [Bioinformatics Commons](#), [Communication Technology and New Media Commons](#), [Databases and Information Systems Commons](#), [OS and Networks Commons](#), and the [Science and Technology Studies Commons](#)

Repository Citation

Han, Q., & Dong, G. (2012). Using Attribute Behavior Diversity to Build Accurate Decision Tree Committees for Microarray Data. *Journal of Bioinformatics and Computational Biology*, 10 (4), 1250005-1-1250005-14. <https://corescholar.libraries.wright.edu/knoesis/382>

This Article is brought to you for free and open access by the The Ohio Center of Excellence in Knowledge-Enabled Computing (Kno.e.sis) at CORE Scholar. It has been accepted for inclusion in Kno.e.sis Publications by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.

Using Attribute Behavior Diversity to Build Accurate Decision Tree Committees for Microarray Data*

Qian Han and Guozhu Dong
Department of Computer Science and Engineering
Wright State University, Dayton, OH 45435, USA
han.6@wright.edu, guozhu.dong@wright.edu

Abstract

DNA microarrays (gene chips), frequently used in biological and medical studies, measure expressions of thousands of genes per sample. Using microarray data to build accurate classifiers for diseases is an important task. This paper introduces an algorithm, called **C**ommittee of Decision Trees by **A**tttribute **B**ehavior **D**iversity (CABD), to build highly accurate ensembles of decision trees for such data. Since a committee's accuracy is greatly influenced by the diversity among its member classifiers, CABD uses two new ideas to "optimize" that diversity, namely (1) the concept of *attribute behavior based similarity* between attributes, and (2) the concept of *attribute usage diversity* among trees. The ideas are effective for microarray data, since such data have many features and behavior similarity between genes can be high. Experiments on microarray data for six cancers show that CABD outperforms previous ensemble methods significantly and outperforms SVM, and show that the diversified features used by CABD's decision tree committee can be used to improve performance of other classifiers

*The work was supported in part by NSF IIS-1044634 and by DAGSI. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

such as SVM. CABD has potential for other high dimensional data, and its ideas may apply to ensembles of other classifier types.

Key Words: Ensemble methods, decision trees, microarray data, attribute behavior similarity, attribute usage diversity.

1 Introduction

DNA microarrays (gene chips), frequently used in biological/medical studies, measure expressions of thousands of genes per sample. Using microarray data to build accurate classifiers for diseases/cancers is an important task, having applications for disease understanding, disease diagnosis, and so on.

The ensemble^{1 2} based approach is very popular because it can produce very accurate classifiers. Considerable work has been done on building ensembles of classifiers in general, and on building ensembles of decision tree [16] classifiers for microarray data in particular. It is well known that a committee’s accuracy is greatly influenced by the diversity among the member classifiers³. Hence, previous ensemble studies proposed various methods to build diversified committees of decision trees, including feature set randomization of Random Forest [5], data instance weighting of Boosting [9], training data randomization of Bagging [4], unique root features of CS4 [14], and feature set disjointness of MDMT [12]. (See Section 2.) Importantly, in all feature-set manipulation based methods discussed above, only name-equivalence based similarity between attributes is used – two attributes are considered similar only if they have the same name.

However, previous ensemble approaches did not consider using behavior similarity among attributes, or using a systematic attribute-usage based method to determine which tree *should* use which attributes. To fill that gap, this paper proposes a new algorithm, called **Committee of Decision**

¹“Ensemble” and “committee” are synonyms, and so are “feature” and “attribute.”

²A classifier ensemble classifies a test case using the vote of its members.

³This is a well known observation, discussed in e.g. [13].

Trees by **A**tttribute **B**ehavior **D**iversity (CABD), to build highly accurate ensembles of decision trees for microarray data. CABD uses two new ideas. (1) CABD uses *attribute behavior based similarity* between two attributes A and B to measure how similar their class distributions are. This similarity is very useful for microarray data, since there are many genes in such data, since this similarity between distinct genes can be very high⁴, and since this similarity among attributes used by decision trees can influence classification behavior diversity among those decision trees. (2) CABD explicitly uses the similarity between attribute sets used by different trees in split attribute selection. Experiments on microarray data for six cancers show that CABD outperforms previous ensemble methods significantly.

We recommend that CABD be used for very high dimensional datasets where class distributions of many attribute pairs are very high, especially where other classifiers’s accuracy are far away from 100%. Moreover, we note that CABD can serve as a feature selection method – the diversified features used by CABD’s decision tree committee can be used to improve performance of other classifiers such as SVM.

Besides microarray data, CABD has potential to be useful for other high dimensional data. Moreover, CABD’s ideas may be adaptable for ensembles of other classifier types in addition to decision trees.

The rest of the paper is organized as follows. Section 2 reviews related work on various ensemble methods. Sections 3 and 4 introduce the concepts of attribute behavior based similarity and attribute usage diversity, respectively. Section 5 presents our CABD algorithm. An experimental evaluation is reported in Section 6. Section 7 concludes the paper.

⁴For example, 5% of all attribute pairs of Colon Cancer have behavior similarity $\geq 99\%$.

2 Related Work

We discuss five ensemble methods⁵, regarding how they build decision tree ensembles and the ideas they use to achieve ensemble diversity. All except Boosting use equal weight committee voting. Bagging and Boosting are generic ensemble methods, while the others are limited to decision trees. CS4 and MDMT were specially designed for microarray data.

Bagging [4] creates an ensemble by sampling with replacement from the original set of training data, to create new training sets for the classifiers. It achieves ensemble diversity through training data randomization.

Boosting [9] builds an ensemble iteratively, in a manner to have new classifiers emphasize hard-to-classify data examples. Each classifier is created using a set of training data where each training example has a weight. Examples incorrectly classified by current classifiers have larger weight for the next iteration. Boosting achieve ensemble diversity by using data weighting, which is based on classification behavior of classifiers.

The random forest method [5] uses randomization of both feature-set and data-set to achieve ensemble diversity. For each node of a decision tree, a subset of the available features is randomly selected and the best split among those features is selected. Moreover, sampling with replacement is used to create the training set of data for each individual tree.

CS4 [14] uses distinct tree roots to obtain ensemble diversity. It constructs a k decision tree ensemble by using each feature whose information gain is among the k highest, as the root node of exactly one of the trees.

MDMT [12] uses feature-set disjointness to obtain ensemble diversity. It constructs an ensemble iteratively, building each new tree using only attributes not yet used by previously built trees.

CABD is novel in its use of attribute behavior based diversity and its use of attribute set usage diversity, in order to increase ensemble diversity.

⁵Reference [2] provides an experimental comparison of several ensemble methods for decision trees, including the first three discussed here.

As will be seen in the experimental evaluation, CABD achieves significantly higher classification accuracy than the other methods for microarray data.

3 Class Distribution Based Behavior Similarity

This section discusses how to define class distribution similarity (sim_{cd}) and scaled class distribution similarity (sim_{cd}^*), in order to adequately treat different levels of class distribution similarity for use in decision tree diversity.

3.1 Defining Basic Class Distribution Similarity (sim_{cd})

We first discuss the goal and rationale for our basic similarity measure, and then give our definition for the measure.

Ultimately, we want to define a basic class distribution similarity so that, two attributes A and B of a dataset D are highly similar if (*) they play similar roles in decision trees. By (*), we mean: There is some order-preserving mapping μ between the values of A and those of B as far as decision trees are concerned. That is, for each decision tree T on D where a value a of A is used to split some node V of T , the classification behavior of T will be preserved if we use the value $b = \mu(a)$ of B to split V , and this relationship holds for every possible value of a of A .

We have chosen to achieve that goal, by first binning each attribute's range into some fixed number of intervals using equi-density binning⁶, then using the class counts of the intervals to describe the class distribution of each attribute, and finally using cosine between the class distributions of two attributes to define our basic class distribution similarity.

Using this equi-density binning based approach allows us to efficiently use the similarity value to reflect (*). This can be explained as follows: First, each split value a of A can be approximated by the bin boundary a' nearest

⁶Given m , the *equi-density* method divides A 's range into m intervals all having an equal (or nearly equal if it is impossible to make them equal) number of matching tuples.

to a . (We use a fairly large number of bins to make the approximation accurate. Of course we should not have too many intervals, to ensure that each bin has sufficiently many matching tuples to make the counts meaningful.) The behavior of a' for a node of a decision tree is indicated by the class distribution of the $A \leq a'$ side and that of the $A > a'$ side. When A and B have highly similar class distributions, a' has a corresponding bin boundary b' among B 's bins such that the class distributions of the two sides of b' are very similar to those of the two sides of a' .

Let D be a given dataset and m be the desired number of intervals. The class distribution of an attribute A in D is defined by

$$\text{CD}(A) = (N_{A11}, N_{A21}, \dots, N_{A1m}, N_{A2m}), \quad (1)$$

where $N_{Aij} = |\{t \mid a_{j-1} < t(A) \leq a_j, t \in D, t\text{'s class is } C_i\}|$, and $a_0, \dots, a_m$ are A 's bin boundaries produced by equi-density binning.

Table 1 shows the class distributions of two attributes A and B . Observe that the total counts for the bins are all roughly 12, and the sum of the counts for the bins is 62 (the size of the underlying dataset), for both A and B . The bin boundaries of different attributes can be different.

Table 1: Class Distributions of Two Attributes

	Bin1	Bin2	Bin3	Bin4	Bin5
A	(1,11)	(1,11)	(1,11)	(8,4)	(11,3)
B	(9,3)	(7,5)	(5,7)	(1,11)	(0,14)

The *class distribution similarity* between two attributes A and B is defined as the cosine (or normalized dot product) of their class distributions:

$$\text{sim}_{\text{cd}}(A, B) = \frac{\text{CD}(A) \cdot \text{CD}(B)}{\|\text{CD}(A)\| * \|\text{CD}(B)\|}. \quad (2)$$

Observe that $\text{sim}_{\text{cd}}(A, A) = 1$ for all attributes A . Hence sim_{cd} recognizes “name equivalence” based similarity (see Section 6.4.2) between attributes.

3.2 Defining Scaled Class Distribution Similarity (sim_{cd}^*)

We introduce a “scaled” approach to differentiate three types/levels of sim_{cd} to help us avoid drawbacks associated with using sim_{cd} directly to measure attribute usage diversity between trees. We first define the scaled class distribution similarity (sim_{cd}^*), and then explain the rationales.

We define sim_{cd}^* between attributes A and B for a dataset D as follows:

$$\text{sim}_{\text{cd}}^*(A, B) = \begin{cases} 1 & \text{if } A \text{ and } B \text{ are the same attribute} \\ \frac{\text{sim}_{\text{cd}}(A, B)}{\kappa} & \text{if } A \neq B \text{ and } \text{sim}_{\text{cd}}(A, B) > 0.6 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Here, κ is a (normalization) constant, which is set to 4 in our experiments.

We now discuss our rationales for the way we defined sim_{cd}^* . Basically, the definition allows us to differentiate different levels/kinds of sim_{cd} values.

(1) We let $\text{sim}_{\text{cd}}^*(A, B) = 1 (= \text{sim}_{\text{cd}}(A, B))$ when A and B are the same attribute, to ensure that each attribute is 100% similar to itself by sim_{cd}^* .

(2) When A and B are not the same attribute, our definition makes $\text{sim}_{\text{cd}}^*(A, B)$ less than $\text{sim}_{\text{cd}}(A, B)$. This allows us to differentiate the case where $\text{sim}_{\text{cd}}(A, B) \approx 1$ and $A \neq B$, from the case where $A = B$. Since we want such $\text{sim}_{\text{cd}}^*(A, B)$ to have an impact on building diversified ensembles, we do not want to make $\text{sim}_{\text{cd}}^*(A, B) = 0$. So we choose to make $\text{sim}_{\text{cd}}^*(A, B)$ smaller than $\text{sim}_{\text{cd}}(A, B)$, by dividing it using a fixed constant κ .

(3) We make $\text{sim}_{\text{cd}}^*(A, B) = 0$ when $\text{sim}_{\text{cd}}(A, B)$ is small. In this way, we ensure that insignificant sim_{cd} between attributes is ignored. This is based on the following analysis on sim_{cd} . It is observed that, when $\text{sim}_{\text{cd}}(A, B)$ is very small, the class distributions of A and of B do not indicate much similarity. In fact, the corresponding intervals of the two attributes often have oppose majority classes. For the example given in Table 1, $\text{sim}_{\text{cd}}(A, B) = 0.49$ and the two attributes have oppose majority classes in 4 out of the 5 intervals.

4 Attribute Usage Diversity

As discussed earlier, our CABD algorithm builds ensembles of decision trees aimed at maximizing attribute usage diversity among the trees. Attribute usage diversity among trees depends on the attribute usage summary of individual trees, and attribute usage based tree difference. We define the two necessary concepts first, before defining attribute usage diversity.

4.1 Attribute Usage Summary (AUS)

Our attribute usage summary for a tree consists of weighted counts of occurrences of attributes in the tree. The weights reflect the importance of the attributes in the tree; we view attributes used near the root of the tree as more important than those used far away from the root.

Specifically, suppose T is a decision tree built from a dataset D . For each attribute A , let $Cnt_T(A)$ denote the occurrence counts of A in T , and let $avgLvl_T(A)$ denote the average level numbers of occurrences of A in T . The root is at level 0. Let $A_1, \dots, A_n$ be a fixed enumeration of the attributes of D . Then, the *attribute usage summary* (AUS) of T is defined as:

$$AUS(T) = (\frac{Cnt_T(A_1)}{avgLvl_T(A_1) + 1}, \dots, \frac{Cnt_T(A_n)}{avgLvl_T(A_n) + 1}). \quad (4)$$

This formulation ensures that occurrences of A_i contribute larger numbers when $avgLvl_T(A_i)$ is small (or if A tends to occur near the root).

4.2 Tree Pair Difference (TPD)

Since every two attributes can have positive sim_{cd}^* similarity value between them, we need to include the contribution of each pair of attributes used in two given trees in defining their tree pair difference. We introduce a $\star$ operation to address that requirement. Suppose that T_1 and T_2 are two trees, and $AUS(T_i) = (f_{i1}, \dots, f_{in})$ for each i . Then

$$AUS(T_1) \star AUS(T_2) = \sum_{1 \leq i, j \leq n} (f_{1i} * f_{2j} * \text{sim}_{\text{cd}}^*(A_i, A_j)). \quad (5)$$

The tree pair difference between two trees T_1 and T_2 is defined as follows:

$$\text{TPD}_\star(T_1, T_2) = 1 - \frac{\text{AUS}(T_1) \star \text{AUS}(T_2)}{\|\text{AUS}(T_1)\|_\star \cdot \|\text{AUS}(T_2)\|_\star}. \quad (6)$$

We define $\|\text{AUS}(T)\|_\star$ to be $\text{AUS}(T) \star \text{AUS}(T)$.

Later we will also use variants of the above definition of $\text{TPD}_\star$, which use other similarity measures between attributes. For example, we may consider the “name equivalence” based similarity where $\text{sim}_{\text{NE}}(A, B) = 1$ if A and B are the same attribute and $\text{sim}_{\text{NE}}(A, B) = 0$ otherwise. Then the corresponding tree pair difference between two trees T_1 and T_2 will become

$$\text{TPD}(T_1, T_2) = 1 - \frac{\text{AUS}(T_1) \cdot \text{AUS}(T_2)}{\|\text{AUS}(T_1)\| \cdot \|\text{AUS}(T_2)\|}. \quad (7)$$

4.3 Attribute Usage Diversity between a Tree and a Tree Set

Let $\text{TreeSet} = \{T_1, T_2, \dots, T_m\}$ be a set of trees, and T another tree.⁷ The *attribute usage diversity* between T and TreeSet is defined to be

$$\text{AUD}(T, \text{TreeSet}) = \min_{i=1}^m \text{TPD}_\star(T, T_i). \quad (8)$$

The *min* operator is used, so that the difference between the most similar pair of trees is used as the diversity.

5 The CABD Algorithm

Our CABD algorithm uses an iterative process to build a desired number (denoted by k) of diversified decision trees. It builds the first tree in the same way as C4.5 does. For each subsequent tree T and each node V of T , it considers an objective function (denoted by IT) that combines information gain of an attribute A and the attribute usage diversity (AUD) between the new tree that results after using A to split V of T and the previous trees,

⁷In our algorithm, TreeSet will represent the set of previously built trees, and T the current tree (possibly after using a candidate attribute to split a node V).

and it selects the split attribute to maximize that objective function. The majority class of a leaf node is assigned as the class label of the node. The pseudo-code for CABD is given in Algorithm 1.

Algorithm 1. Decision Tree Committees Using
Attribute Behavior Diversity (CABD)

Input: training dataset D and number k

Output: a diversified committee of k decision trees

Method:

1. Let $T_1, \dots, T_k$ denote the k decision trees to be built;
 2. For $i = 1$ to k do
 3. Recursively split the next tree node V of T_i , when T_i 's
 4. nodes are visited in the depth-first order, as follows:
 5. Select attribute B and value b_V such that
 6. $IT(T_i(B, b_V)) = \max\{IT(T_i(A, a_V)) \mid a_V \text{ is a candidate}$
 7. $\text{split value for attribute } A \text{ at } V\}$;
 8. If B and b_V exist
 9. Then use them to split V and generate V 's two children;
 10. Else V is made a leaf node, labeled by its majority class;
 11. Output $\{T_1, \dots, T_k\}$ as the diversified decision tree committee.
-
-

Remarks: (a) In the process of building one decision tree, CABD differs from C4.5 in one significant aspect when selecting split attributes for nodes: It considers the contribution to diversity of a candidate attribute A , in addition to A 's information gain. (b) In the iterative process of building k trees, CABD differs from previous ensemble methods in two key aspects: It considers attribute behavior based similarity and it explicitly evaluates attribute usage diversity among the trees.

We now describe the objective function IT used by CABD to select split attributes. Let $TreeSet = \{T_1, T_2, \dots, T_m\}$ be the set of previously built

trees, and T the current (partial) tree under construction. Let V be the current node of T to split, and A and a_V resp. a candidate splitting attribute/value. Let $T(A, a_V)$ be the tree obtained by splitting V in T using attribute A and split value a_V . Then IT is defined by:

$$IT(T(A, a_V)) = IG(A, a_V) + \text{AUD}(T(A, a_V), \{T_1, \dots, T_m\}), \quad (9)$$

where $IG(A, a_V)$ is the information gain⁸ when V is split by A and a_V .

6 Experimental Evaluations

This section reports an experimental evaluation of CABD using six microarray datasets for cancers (see Table 2). The results show that CABD outperforms competing decision tree based ensemble methods, CABD outperforms SVM [6], and using CABD’s features improves SVM’s performance.

This paper used the same experiment settings as [12]: Accuracy results were obtained using 10-fold cross-validation, and for each dataset, the union of the training and test data sets given in the original papers for the dataset was used (to enlarge the dataset for the experiments).⁹ The number of decision trees per ensemble was set to 25.

Accuracy results for the competing ensemble methods and C4.5 are extracted from [12]. Accuracies of SVM were obtained using the SVM implementation of WEKA¹⁰.

For datasets having many thousands of genes, attributes with very low information gains have very little chance to be selected in decision trees. So CABD only considers those attributes ranked in the top 30% by information

⁸Suppose D' is the dataset for V and C_1, C_2 are the classes. Define $\text{Info}(D') = -\sum_{i=1}^2 p_i \log_2(p_i)$, where p_i is the probability of a tuple belonging to class C_i . Let $D'_\ell = \{t \in D' \mid t[A] \leq a_V\}$ and $D'_r = \{t \in D' \mid t[A] > a_V\}$. Define $\text{Info}(A, a_V) = \sum_{i=\ell, r} \frac{|D'_i|}{|D'|} \times \text{Info}(D'_i)$, and $IG(A, a_V) = \text{Info}(D') - \text{Info}(A, a_V)$.

⁹In cross validation, classifiers do not access the testing data in the training phase.

¹⁰www.cs.waikato.ac.nz/ml/weka/

gain to save computation time. (If there are ≤ 2000 attributes, all are used.) CABD could have done better if we considered all attributes.

Table 2: Description of the Datasets

Dataset	No. of Tuples	No. of Attributes	No. of Classes	Reference
Breast	97	24481	2	[18]
Colon	62	2000	2	[1]
Leukemia	72	7129	2	[10]
Lung	181	12533	2	[11]
Ovarian	253	15154	2	[15]
Prostate	21	12600	2	[17]

6.1 CABD Outperforms Other Ensemble Classifiers

In this section we compare CABD against other ensemble methods and C4.5 with respect to accuracy and receiver operating characteristics (ROC).

We first consider the accuracy based comparison. Table 3 shows that CABD outperforms all five competing ensemble algorithms and C4.5 on the six datasets. We compare them using 4 common comparison methods.

- CABD is more accurate on average: The average accuracy of CABD is 1.6% (absolute improvement) better than MDMT (the best competing method) and about 7%–11% (absolute improvement) better than C4.5, Boosting, Random Forest and Bagging.
- CABD achieves the best accuracy on five of the six datasets, when compared against the 5 competing ensemble methods all together.
- CABD is much better than other approaches when considered in pairwise Win-Loss-Tie comparison: CABD has (6 wins, 0 loss, 0 tie) against

each of C4.5, Boosting, Random Forest, Bagging, has (5 wins, 1 loss, 0 tie) against MDMT, and has (4 wins, 1 loss, 1 tie) against CS4.

- Even when compared against the best competitors, CABD has more significant wins. (We say a method achieves a significant win over another if the relative accuracy improvement by the former over the latter is $\geq 5\%$.) Indeed, CABD has two significant wins over CS4 while CS4 has just one over CABD, and CABD has two significant wins over MDMT while MDMT has none over CABD.

The exceptional performance of CABD is due to its use of attribute behavior based similarity and its explicit use of attribute usage diversity in the tree building process.

Table 3: Accuracy Comparison (RF: Random Forest; bold: best accuracy)

Dataset	C4.5	RF	Boosting	Bagging	CS4	MDMT	CABD
Breast	62.9%	61.9%	61.9%	66.0%	68.0%	64.3%	68.0%
Colon	82.3%	75.8%	77.4%	82.3%	82.3%	85.8%	86.9%
Leukemia	79.2%	86.1%	87.5%	86.1%	98.6%	97.5%	93.2%
Lung	95.0%	98.3%	96.1%	97.2%	98.9%	98.9%	99.5%
Ovarian	95.7%	94.1%	95.7%	97.6%	99.2%	96.4%	99.6%
Prostate	33.3%	52.4%	33.3%	42.9%	47.6%	60%	65.0%
Average	74.6%	78.1%	75.3%	78.7%	82.4%	83.8%	85.4%

Next we compare CABD against other methods using the ROC curve [3], which is frequently used to compare different classification methods. A classifier with a larger area under the curve is usually considered better than another with a smaller area under the curve (AUC).

Figure 1 shows the ROC curves for CABD and other methods, obtained in WEKA by varying the threshold value on the class probability estimates.

Table 4: Average sim_{cd} and Percentages for High sim_{cd}

	Breast	Colon	Leukemia	Lung	Ovarian	Prostate
avg sim_{cd}	0.88	0.94	0.83	0.94	0.85	0.90
$\text{sim}_{\text{cd}} \geq 0.99$	0.1%	5%	0.02%	6%	3%	6%
$\text{sim}_{\text{cd}} \geq 0.98$	1%	14%	0.5%	24%	15%	8%
$\text{sim}_{\text{cd}} \geq 0.95$	11%	49%	9%	57%	44%	35%

The curve for CABD is more close to the upper left corner and has larger area under curve (AUC) than the other methods, such as Random Forest, Bagging, and CS4. The figure implies that CABD outperforms other ensemble methods with larger AUC.

Figure 1: ROC Curve Comparison

Remark: We observe that CABD is likely to lead to significant improvement over any other ensemble method in situations where (a) the attributes in many attribute pairs are highly similar to each other, and (b) the other

method is not already achieving near perfect accuracy of 98% or higher. For example, CABD significantly outperforms MDMT on Breast, Ovarian, and Prostate. For the three datasets, the behavior similarity among attributes is high for Ovarian and Prostate (see Table 4, which shows¹¹ the average sim_{cd} between all considered attribute pairs, and the percentage of attribute pairs (among all attribute pairs) with very high sim_{cd} ($\geq p$)). The above observation should be used as a rule-of-thumb; if time permits, one may want to consider using CABD for all datasets where other ensemble methods do not achieve near perfect accuracy.

6.2 CABD Outperforms Support Vector Machine

Now we compare CABD against the well known support vector machine (SVM) [6] (with polynomial kernels), which is often more accurate than other classifiers. It should be noted that CABD has an advantage – it is more understandable than SVM since CABD’s decision trees are more understandable whereas SVM is like a “block box”.

Table 5 shows that CABD outperforms SVM by more than 2% (absolute improvement) on average. Clearly CABD and SVM are tied on Breast, and they are essentially tied on Colon, Lung, and Ovarian (the difference in accuracy is $\leq 0.4\%$). For the remaining two datasets, CABD beats SVM by a very large relative improvement of 36% on Prostate, compared with a much smaller relative loss of 4% on Leukemia.

6.3 Use CABD’s Features to Improve SVM

The features used by the decision trees in CABD’s committee can be used as a diversified feature set to improve the performance of other classifiers. For example, Table 6 shows that the accuracy of SVM that uses only the attributes used by CABD’s committee beats the SVM that uses all attributes

¹¹The table only considered the attributes used as candidate attributes in the experiments (see the last paragraph of Section 6).

Table 5: Accuracy Comparison between CABD and SVM

	Breast	Colon	Leukemia	Lung	Ovarian	Prostate	Average
CABD	68.0%	86.9%	93.2%	99.5%	99.6%	65.0%	85.4%
SVM	68.0%	87.1%	97.2%	98.9%	100%	47.6%	83.1%

by a large margin of 4.8% (absolute improvement).

Table 6: Accuracy Comparison between SVM_{SA} and SVM

	Breast	Colon	Leukemia	Lung	Ovarian	Prostate	Average
SVM_{SA}	84.5%	85.5%	100%	100%	100%	57.1%	87.9%
SVM	68.0%	87.1%	97.2%	98.9%	100%	47.6%	83.1%

6.4 Other κ , Similarity and Discretization Measures

6.4.1 Impact of κ on CABD

In all our experiments, the normalization constant κ of Equation (3) was set to 4. Table 7 gives the performance of CABD for $\kappa = 2, 4, 6$. We can see that $\kappa = 4$ achieves the highest average accuracy over all datasets. In general, we recommend that $\kappa = 4$ be used. However, depending on the characteristics of a given dataset, other κ can be evaluated.

6.4.2 Other Similarity Measures

Table 8 shows that CABD is much better than the cases when sim_{cd}^* used by CABD is replaced by two other similarity measures, implying that attribute behavior similarity is a very useful concept for increasing ensemble

Table 7: Impact of the Normalization Constant κ

Dataset	$sim *_{cd} (\kappa = 2)$	$sim *_{cd} (\kappa = 4)$	$sim *_{cd} (\kappa = 6)$
Breast	65.0%	68.0%	64.9%
Colon	80.5%	86.9%	83.8%
Leukemia	94.6%	93.2%	97.3%
Lung	99.5%	99.5%	98.9%
Ovarian	98.4%	99.6%	99.6%
Prostate	48.3%	65.0%	51.7%
Average	81.1%	85.4%	82.7%

diversity and accuracy. (a) The *name equivalence based similarity*, sim_{NE} , is defined by $sim_{NE}(A, B) = 1$ if A and B are the same attribute and $sim_{NE}(A, B) = 0$ otherwise. (b) The *high sim_{cd}^* only similarity*, sim_{high} , is defined by $sim_{high}(A, B) = sim_{cd}^*(A, B)$ if $sim_{cd}^*(A, B)$ is in the top 15% of sim_{cd} values, and $sim_{high}(A, B) = 0$ otherwise.

Table 8: sim_{cd}^* vs Other Similarity Measures

Dataset	sim_{cd}^*	sim_{NE}	sim_{high}
Breast	68.0%	65.0%	66.9%
Colon	86.9%	83.8%	85.5%
Leukemia	93.2%	95.9%	94.9%
Lung	99.5%	99.4%	98.9%
Ovarian	99.6%	99.0%	99.6%
Prostate	65.0%	56.7%	46.7%
Average	85.4%	83.3%	82.1%

6.4.3 Other Discretization Measures

While the experiments above used the equi-density binning method to measure the similarity between attributes, other binning methods can also be used, including the popular entropy based method [7, 8]. To discretize an attribute, the entropy based method chooses the value having the minimum entropy as the cut value, and may recursively partition the resulting intervals until enough intervals are made.

Table 9 compares the accuracy of equi-density binning with entropy based binning (which discretizes each attribute into four intervals). The equi-density method outperforms the entropy based method in all datasets, with an absolute improvement of 7.4% on average. Hence equi-density binning is the better way for measuring attribute similarity.

Table 9: Accuracy Comparison: Equi-Density vs Entropy Based Binning

	Breast	Colon	Leukemia	Lung	Ovarian	Prostate
Equi-density	68.0%	86.9%	93.2%	99.5%	99.6%	65.0%
Entropy based	67.9%	71.0%	87.5%	92.8%	96.8%	51.7%

7 Concluding Remarks

In this paper, we proposed the CABD algorithm to build diversified ensembles of decision trees for microarray gene expression data. The algorithm is based on two key ideas to “optimize” ensemble diversity: (1) It introduces the concept of *attribute behavior based similarity* between attributes. (2) It uses the concept of *attribute usage diversity* among trees in split feature selection. Experiments show that CABD outperforms previous ensemble methods and outperforms SVM, and the features used by CABD’s committees can be used by other classifiers to improve their performance. We recommend that

CABD be used for very high¹² dimensional datasets where class distributions of many attribute pairs are very high and other classifiers do not have very high accuracy.

References

- [1] U. Alon, et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96:6745–6750, 1999.
- [2] Robert E. Banfield, Lawrence O. Hall, Kevin W. Bowyer, and W.P. Kegelmeyer. A comparison of decision tree ensemble creation techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:173–180, 2007.
- [3] Andrew P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.
- [4] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
- [5] Leo Breiman. Random forests. *Machine Learning*, 45, 2001.
- [6] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.
- [7] Usama M. Fayyad and Keki B. Irani. The attribute selection problem in decision tree generation. In *Proceedings of the tenth national conference on Artificial intelligence*, AAAI’92, pages 104–110. AAAI Press, 1992.
- [8] Usama M. Fayyad and Keki B. Irani. On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, 8:87–102, 1992.
- [9] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156, 1996.
- [10] T.R. Golub, et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
- [11] Gavin J. Gordon, et al. Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. *Cancer Research*, 62:4963–4967, 2002.

¹²CABD may not have advantage over other ensemble methods for data with very low dimensions. This was confirmed by experiments.

- [12] Hong Hu, Jiuyong Li, Hua Wang, Grant Daggard, and Mingren Shi. A maximally diversified multiple decision tree algorithm for microarray data classification. In *Workshop on Intelligent Systems for Bioinformatics, Hobart, Australia*, 2006.
- [13] Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003.
- [14] Jinyan Li and Huiqing Liu. Ensembles of cascading trees. In *IEEE International Conference on Data Mining*, pages 585–588, 2003.
- [15] Emanuel F Petricoin III, et al. Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet*, 359:572–577, Feb. 2002.
- [16] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.
- [17] Dinesh Singh, et al. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1:203–209, Mar. 2002.
- [18] Laura J. Van’t Veer, et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–536, 2002.