International Journal of Database Theory and Application
Vol.9, No.12 (2016), pp.203-216
http://dx.doi.org/10.14257/ijdta.2016.9.12.19

Research on an Improved Decision Tree Classification Algorithm

Wenyi Xu

Heyuan Polytechnic, Heyuan 517000, china
20978646@qq.com

Abstract

In the paper, with the introduction of data mining algorithm of the classification in detail, and then combining the classification algorithm and incremental learning technology, an incremental decision tree algorithm is proposed to solve the problem of incremental learning and analysis the experimental data for this algorithm. The paper used ID3 and C4.5 algorithm for detailed research. According to two algorithms, combining Bayesian classification algorithm's incremental learning characteristic, the paper proposed an incremental decision tree algorithm , and by the analysis of experimental data. This algorithm can solve the incremental learning problem of the decision tree algorithm very well.

***Keywords**: Data mining, Classification algorithm, Decision tree, Incremental learning*

1. Introduction

In the practical application of data mining, there are some problems [1-10]: how to efficiently classify those data and then fetch important information from them with data mining technique? [1-18]That is a hot research topic since the emergence of data mining science [19-28]. A good and efficient data mining classifying algorithm can reduce greatly the difficulty of knowledge mining in later period to get two-fold results with half the effort, making data mining result more distinct and accurate [29-37].

The appearance of database helps us save a great deal of information data resource, where lots of valuable hidden information is included, which can be used to offer foundations for making wise decision. Data mining is mainly used for data classification and prediction to withdraw useful rules and pattern [38-46]. Decision tree classifying algorithm is one method of data mining algorithm which can support well data type of large database, with excellent scalability [47-55]. Also the classification pattern ultimately generated by the algorithm is easily transformed into rules [56-60].

Mainstream methods of data mining classifying algorithms are decision tree algorithm and two common ones like id3 algorithm and C4.5 algorithm. But in real practice, the two methods do not have good adaptability to the new added data volume, i.e. how to make correct classification of rapidly growing data is the key to the whole classifying process.

Based on the above research background, the paper aims to make in-depth studies on data mining core algorithm and choose its classification as key research direction to discuss decision tree algorithm, and do case study and improvements of decision tree algorithm. By combining with Bayesian algorithm, the paper proposes an incremental decision tree algorithm which can solve incremental learning problem and validates its effectiveness through experimental data analysis.

2. Overview of Decision Tree Classification Algorithms

First, you need to understand the definition of the decision tree, the decision tree structure and the flow chart is very similar. Each non leaf node in the tree is a test of a property, and the output of the test is the branch of the tree.

ISSN: 2005-4270 IJDTA

The root node of the decision tree is the initial node, or the class distribution or class is each leaf node. The decision tree classification algorithm consists of two steps: decision tree and decision tree pruning.

2.1Decision Tree Construction Algorithm

For the decision tree construction algorithm we need a sample data set containing the class label as the input data, the shape of the decision tree is the two fork tree or multi fork tree. The conclusion of the final logical judgment is that the decision tree is the edge of the decision tree.

We mainly use the method of top-down recursive construction of decision tree. The method is to use a node which can represent the training sample as the initial root node of the decision tree; Suppose that the sample is in the same class, using the class label to label this node, and it is defined as the leaf node. If not in the same class, the heuristic information of the decision tree classification algorithm is used to measure the value of the gain of the information entropy. And the sample can be quickly classified attributes to identify the test attribute or attribute as the node.

It should be noted that all of the properties of the decision tree construction algorithm are taken as discrete values. The algorithm has a more obvious shortcomings, in the data mining before. If the data is continuous, the data must be discretized, because it can only deal with the discrete attributes.

The sample data is partitioned according to the branch that is created with the known test attribute. Then repeat such steps, in the classification of each division, uses recursive way to form the classification of the decision tree samples. When a property has appeared in the decision tree node, then there is no need to investigate the nodes of the subsequent nodes.

Recursively divides the decision tree, when there are several conditions to terminate the recursion:

1All data types of a node are the same;

2Not used to further divide the remaining attributes of the sample, we use the majority voting method to decide. Property of the class to convert the remaining attributes into leaf nodes, and use the sample data. The largest number of attributes in the class label to label;

3 The majority voting method is adopted to test the non-sample data of some branch test attributes to determine, directly to the data sample set to occupy most of the attributes of the class label to create a tree leaf node.

Algorithm decision tree construction algorithm

Algorithm input: discrete attributes of the training data set data_list candidate attributes set list_attribute. Algorithm output: the decision tree generated by the input training sample set
1 Create node M 2 If the data_list is in the category D, the M is returned as a leaf node, with the class D tag, and Return； 3 If the list_attribut is empty, the leaf node is M, and the majority voting method is adopted. The most general class, and return; 4The list_attribute information entropy value of the maximum value of information gain extracted from the property, marked as attribute_test 5 Attribute_test marked as node M 6 M node generation conditions for the attribut_test=ai branch 7Definition S_i is a subset of the samples in the data_list, S_i meet the conditions attribute_test=ai 8If S_i is empty, add a leaf node, and S_i is marked as a general class; 9Otherwise, add a node N, which is generated by the function Generate_decision_tree

Choosing a good judgment logic or attribute is difficult to construct a good decision tree. The same decision tree can be found in different decision trees. Usually, the smaller the tree structure, the better the prediction ability of the tree.

To construct a small decision tree, it is necessary to select the attributes of the branch. Constructing the smallest tree is difficult. So the heuristic method can be used to choose the strategy. The method of measuring the purity of each sample is the key to attribute selection.

These methods include: distance measure, information gain, information gain ratio, orthogonal method, correlation degree and so on. Different effects of different methods are different. Through the analysis of the actual situation, and then select the appropriate measurement methods to the impact of the results.

2.2. Decision Tree Pruning Algorithm

In the daily work life, the sample data are generally flawed. Some data in the attribute field may lack the necessary data to make the data incomplete. There may also be inaccurate data, including noise. The following mainly introduces the data noise problem.

This time you need to use the pruning technique, it is a basic method to overcome the noise data, by pruning operation it can greatly simplify the structure of the decision tree, so that the structure of the decision tree is more understandable.

At present there are two main pruning method:

Pre pruning strategy: When making decision tree, decide whether to classify or stop the subset of noisy training.

After pruning strategy: Method of fitting a simple. The initial decision tree to generate a complete data fitting, then the base of the tree bottom-up pruning of branches.

In pruning at the same time, uses a test set to test the decision tree. If a node is removed, the accuracy of the decision tree or other metrics are not the same, then remove the node; otherwise, stop.

From the theoretical analysis, pre pruning strategy without post pruning strategy. But after pruning time complexity. Decision tree pruning process usually requires the use of some thresholds or statistical parameters.

Note that all pruning and bring benefits to the data set as the minimum decision tree is not the best decision tree. When the amount of data scarce, pruning will bring side effects, this is to prevent.

2.3 ID3 Algorithm

Shannon put forward and developed the information theory in the year, it is based on mathematical methods, including the use of probability theory. Mathematical statistics to measure and research information, information entropy, etc.

By means of communication, the amount of information is measured, and the concept of self-information quantity, conditional entropy, information entropy and average mutual information is proposed.

In the initial stage of machine learning, there is only an empty decision tree, which is not known. At this point, we use the decision tree model obtained from the previous learning to classify the whole attribute space.

ID3 algorithm

Input: sample data set D, attribute set C Output: decision tree T
1 Begin 2 If the data set D is empty, return 3 Create node N

4 If there is no attribute that can be used to predict the decision tree T, then the next step
5The attribute set of the data set is empty, the nodes are transformed into leaf nodes, and the data are used in the nest. Class label for a number of instances.
6 If all instances of the data set have the same class attribute V
7 Mark node N category is V
8 The average information entropy AVG ENTROPY (A, C, T) is calculated for each attribute of the pair.
9 The average information entropy AVG ENTROPY (A, C, T) attribute value of the smallest mark Amin
10If AVG ENTROPY (A, C, T) is not smaller than AVG ENTROPY (A, C, T). In the decision tree T, most of the instances belong to the class label A
11 The average information for each Yu minimum attribute tag to node N
12 For all the properties of the smallest class V of the average information entropy, do recursive call N1=ID3
13 If N1 is empty, make an arc from N to N1 labeled V
14 Return node N.

It is shown in table 2, gives a data set that may contain noise, which contains four Outlook, Temperature, Humidity and Windy. It is divided into two categories NO and YES. ID3 algorithm to construct the decision tree to classify the data.

Attribute	Outlook	Temperature	Humidity	Windy	Class
1	Overcast	Hot	High	Not	No
2	Overcast	Hot	High	Very	No
3	Overcast	Hot	High	Medium	No
4	Sunny	Hot	High	Not	Yes
5	Sunny	Hot	High	Medium	Yes
6	Rain	Mild	High	Not	No
7	Rain	Mild	Normal	Medium	No
8	Rain	Hot	Normal	Not	Yes
9	Rain	Cool	Normal	Medium	No
10	Rain	Hot	Normal	Very	No
11	Sunny	Cool	Normal	Not	Yes
12	Sunny	Cool	Normal	Medium	No
13	Overcast	Mild	High	Very	Yes
14	Overcast	Mild	High	Very	No
15	Overcast	Mild	High	Medium	Yes
16	Overcast	Mild	Normal	Not	Yes
17	Rain	Hot	Normal	Medium	Yes
18	Rain	Mild	Normal	Very	Yes
19	Overcast	Mild	Normal	Very	Yes
20	Overcast	Mild	Normal	Medium	No
21	Sunny	Mild	High	Hot	Yes
22	Sunny	Hot	High	Medium	Yes
23	Sunny	Mild	Normal	Very	Yes

2.4 Analysis of the Advantages and Disadvantages of ID3 Algorithm

According to the specific steps of the algorithm mentioned above, and the search strategy and space of the algorithm, it can be summed up the advantages and disadvantages.

The hypothesis space of the algorithm includes all the properties of a finite discrete valued function of a complete space. Each discrete valued function represents a decision tree. One advantage of the algorithm is that: Assuming that the space is complete, the objective function is avoided.

Algorithm is not backtracking algorithm, so when the test of a property, the algorithm will not be back to the property to re test. In this way, it is possible to get a local optimal solution, rather than the global optimal solution. The algorithm is easy to fall into local optimum.

ID3 algorithm can only deal with the discrete data is another drawback of the algorithm. When studying a sample data, it requires that the attribute of the sample data is discrete, and the attribute of the decision tree node is also discrete.

ID3 algorithm will continue to increase the depth of the decision tree until the generated decision tree can completely fit the sample data. But when the sample data has the wrong data or noise data, the strategy will not be effective, resulting in the decision tree will be over fitting the data sample

There are several ways to avoid over fitting in decision tree learning, and it can be divided into two categories:

Pre pruning, before discovery algorithm creates a perfect fitting classification decision tree growth. After pruning, after the formation of over fitting in the tree to prune the tree in the algorithm.

The first method looks very direct and effective, but in practical use. The second approach is more successful.

No matter which method is used, the final determination of the correct tree size is the key problem. We can use the following method:

The classification effect of the post pruning decision tree is made by using the sample data with different sample data.

Learning from all the data samples, using statistical methods to estimate whether the extension or removal of a node can improve the problem of over fitting of the decision tree.

To sum up, the single effective and strong learning ability is the advantage of the algorithm. Disadvantage is that the size of the data set is relatively small, and the sample data set attribute must be discrete, the training data set is large, the structure of the decision tree may change.

3. Theoretical Foundation of Naive Bayesian Method

Definition 1: Set x is a sample data. The data sample class label is unknown, which we make H as one assumption. If data sample X belongs to a specific type D, hope to get $P(H \mid X)$, which means: for sample data X, supposed the probability of H which can hold, then $P(H \mid X)$ is calculated with following method.

$$P(H \mid X) = \frac{P(X \mid H)P(H)}{P(X)} \tag{1}$$

In which, $P(H \mid X)$ is priori probability or priori probability of H; $P(H \mid X)$ means the possibility of X being observed when assumption can hold; $P(H \mid X)$ means priori probability assuming H can hold in the condition X.

Take for example, suppose if data sample is composed of fruits, use fruit color and shape to describe features. If X color is red, shape is round, H is assumption; if X is apple, $P(H \mid X)$ is the confidence level of the hypothesis that X is apple can hold when X is found round and in red color.

It can be seen that $P(H \mid X)$ grows along with increasing $P(H)$ and $P(X \mid H)$; meanwhile, $P(H \mid X)$ diminishes with increasing $P(X)$. Bayesian classifier can realize better classifying result for completely dependent data and function-reliant data.

4. Naive Bias Classification Theory

Bayesian classifying process includes following steps:

1 Use n-dimensional feature vector $X = \{x_1, x_2, ..., x_n\}$ to represent all data samples; describe respectively n dimensions of sample $A_1, A_2, ..., A_n$ which has n attributes;

2 Suppose there are m classes $(C_1, C_2, ..., C_m)$; X is unknown sample data of unlabeled class; classifier predicts X belongs to the class with highest priori probability in the condition; specifically, Bayesian classifier will categorize unknown sample data to the class $C_i(1 <= i <= m)$ when and only when $P(C_i | X) > P(C_j | X)$; the biggest priori assumption is $P(C_i | X)$ is the biggest class C_i; $P(C_i | X)$ can be determined by following Bayes' theorem;

$$P(C_i | X) = \frac{P(X | C_i)P(C_j)}{P(X)} \tag{2}$$

3 Since $P(X)$ is constant, so $P(X | C_i)P(C_j)$'s value is bigger and thus $P(C_i | X)$'s value becomes bigger; the priori probability of C_i class is unknown, then we can assume the probability of those classes is equal, i.e. $P(C_1) = P(C_2) = ... = P(C_m)$; hence the question turns to seek maximized value of $P(X | C_i)$; $P(X | C_i)$ is often called the likelihood of data when C_i is given; the assumption for making $P(X | C_i)$ biggest is named the biggest likelihood assumption; otherwise, it requires maximizing $P(X | C_i)P(C_j)$. What's noted is hypothesis is not equiprobable. So the priori probability of class is estimated through $P(C_i) = s_i / s$, where si is number of training samples in class C_i; s_i is total number of training samples.

4 Since dataset has too many attributes, computing $P(X | C_i)$ will cause huge extra expenses. To avoid such amount because of calculating $P(X | C_i)$, we assume class condition can hold. As the class label of sample is known, according to the supposition that attributes are conditionally independent, we can conclude attributes are not mutually dependent.

$$P(X | C_i) = \prod_{k=1}^{n} P(X_k | C_i) \tag{3}$$

5. Example of Application of Bias Theory

After the detailed introduction above, we have a preliminary understanding of the basic idea of naive Bayesian method. A specific application example of Bayesian method is introduced in this paper. It is shown in table1.

From tabe1, data sample set to make 'age', 'income', 'sutdent' and 'credit_rating' description. Class label attribute 'buys_computer' has two values of Yes and No.

Let B_1 correspond to class buys_computer="Yes", while B_2 corresponds to class buys_computer="No"

The unknown sample data required for classification is:

X=(age=<'30',income='medium', student='Yes', credit_rating='fair')

Need to maximize $P(X \mid C_i)P(C_j)$ .The prior probability of each category attribute is calculated by the appeal sample $P(C_i)$

P(buys_computer='Yes')=9/14=0.64

P(buys_computer='No')=5/14=0.36

In order to calculate $P(X \mid C_i)$, $i = 1, 2$,The conditional probabilities are calculated respectively.

P(age<30| buys_computer='Yes')=2/9=0.22

P(age<30| buys_computer='No')=3/5=0.6

P(income='medium'| buys_computer='Yes')=4/9=0.44

P(income='medium'| buys_computer='No')=2/5=0.4

P(student='Yes'| buys_computer='Yes')=6/9=0.67

P(student='Yes'| buys_computer='No')=1/5=0.2

P(credit_rating='fair'| buys_computer='Yes')=6/9=0.67

P(credit_rating='fair'| buys_computer='No')=2/5=0.4

Assuming that each condition is independent, using the above probability, got:

P(X| buys_computer='Yes')=0.44*.022*0.67*0.67=0.044

P(X| buys_computer='No')=0.6*.04*0.2=0.019

P(X| buys_computer='Yes')* P(buys_computer='Yes')=0.64*0.044=0.028

P(X| buys_computer='No')* P(buys_computer='No')=0.35*0.019=0.007

So, according to this probability, for the data sets of samplesX, Naive Bias classification prediction buys_computer='Yes'

Table 1. Sample Data Set

ID	AGE	INCOME	STUDENT	CREDIT_RATING	BUY_COMPUTER
1	<=30	High	No	Fair	No
2	<=30	High	No	Excellent	No
3	31-40	High	No	Fair	Yes
4	>40	Medium	No	Fair	Yes
5	>40	Low	Yes	Fair	Yes
6	>40	Low	Yes	Excellent	No
7	31-40	Low	Yes	Excellent	Yes
8	<=30	Medium	No	Fair	No
9	<=30	Low	Yes	Fair	Yes
10	>40	Medium	Yes	Fair	Yes
11	<=30	Medium	Yes	Excellent	Yes
12	31-40	Medium	No	Excellent	Yes
13	31-40	High	Yes	Fair	Yes
14	>40	Medium	No	Excellent	No

From the above instance, we find it estimates the probability of the occurrence of one event from the whole time aspect. For example, in the instance, the ratio for estimating probability P(age<30| buys_computer='Yes')is n_c / n , $n = 9$ is sample quantity of all buys_computer='Yes', while $n_c = 2$ is sample quantity when age<30.

In many cases, the perceived proportion is a good estimation of probability. But when n_c is very small, estimation will become bad. In this case, the probability is almost 0; so the result would be 0 when the condition is inquired.

To overcome the difficulty, we utilize Bayesian method based on estimated possibility, which is defined m- as follows:

$$m - \text{estimate} = \frac{n_C + mp}{n + m} \tag{4}$$

In the formula, n_C is defined the same as above; P is a confirmed priori estimation probability; m's value is equal to the size of current sample, which is used to judge P with observed data. The formula aggrandizes the scale of n actual samples. Together with m virtual samples, they are distributed as per P. When other information is missing, suppose priori probability is an effective way to choose p evenly; if one attribute has k possible values, then set P=1/K. Suppose to estimate P(age<30| buys_computer='Yes'), attribute age has three possible values, so even priori probability can be set p=0.33; if m=0, m estimation is equivalent to simple percentage n_c / n; if neither n nor m is 0, priori probability P and observed percentage n_c / n can merge as per weight ratio m.

6. Incremental Learning

Incremental learning is a natural learning way through which learning system can learn new knowledge persistently from new sample from the environment and knowledge incremental learning mostly acquired can be assured to be close to human beings.

With rapid advancement of machine learning and artificial intelligence, people have developed plentiful machine learning algorithms. But such algorithms are mostly batch learning, that is, they learn all sample data at one time, which is the prerequisite. Through learning those samples, learning process ends and it's not possible to acquire new knowledge. That is contradiction between theory and reality. In actual practice, sample data can not be obtained at one time, instead, the data is gained in a continuous manner, which is called incremental problem. In that course, sample's data will change along with the time going. If a new sample data is got, it needs re-learn all data, clearly which wastes a lot of time and resources. With increment of data, the requirement for storage space becomes fierce and quick. So batch learning is suitable for that need. Incremental learning can allow to updated and modify acquired knowledge base without haste; moreover, knowledge information after update can well adapt to refreshed data, no need of relearning data.

Incremental learning has significance. With development and application of database and Internet technology, the society of all circles has accumulated tremendous data and those data are gradually increasing. How to withdraw useful information from those data and classify them is a troublesome task. Traditional batch learning style is static, unlikely to adapt to the environment; only through the way of incremental learning, can the requirement be settled.

Incremental learning algorithm is one of machine learning methods. The incremental learning classifier based on Bayes can in certain degree realize incremental learning. The incremental decision tree algorithm mentioned later in the next is a new technique which integrates Bayesian incremental learning and decision tree to solve decision tree incremental learning trouble.

7. Implementation of Incremental Decision Tree Algorithm (HID)

Due to information technology advance, information data grow explosively and data volume becomes bigger and bigger. Classification being important part of data mining, traditional data mining classifying algorithm shows big limits in the face of booming data amount, especially when information data were acquired in segments. In this case, the method to solve it is incremental learning, which utilizes priori information to deal with it.

Decision tree classifying algorithm is a technique with instance as basis. Decision tree classifying algorithm adopts top-down recursive way to compare attributes of internal nodes of decision tree. Based on different attribute value, judge branches downwards is from the node and make conclusion at leaf node of decision tree.

Incremental decision tree algorithm combines the application of naive Bayesian method and decision tree algorithm to overcome the incremental learning problem of decision tree algorithm.

7.1. Interface of Bayesian Classifier

We may as well assume the attribute space of one node of decision tree in one sample dataset is D, which in the next is regarded as parameter of Bayes classifier according to classification and attribute space D of the sample dataset. When there is new sample data, do naive Bayesian classification of sample data which arrive at the node. The node is named Bayes node.

Then the key issue is how to embed Bayes node to decision tree in later part. For realize that purpose, there are two methods: (i) embed all decision tree nodes to Bayes node; (ii) insert Bayes node to leaf node of decision tree structure. The proposed incremental decision tree algorithm in the paper takes a flexible method which enables Bayes nodes to have the ability that all leaf nodes of decision tree can be inserted there. When it's necessary for leaf nodes to insert to Bayes nodes, only part of leaf nodes are Bayes nodes.

7.2. Incremental Decision Tree Algorithm

According to current data set sample, we use decision tree algorithm to classify it. Then according to discrete attribute value, segment dataset sample in recursive way into several small sample data sets. With those data sets, construct decision tree nodes, which can be divided into Bayes nodes and common leaf nodes as per the actual procedure.

During generation, there are two cases when the generative algorithm produces one leaf node.

Firstly, when all current sample data are put in one class, this would lead to over-matching; hence it needs pre-pruning as follows: calculate and measure the difference value of node before categorization and compare it with threshold; if it's bigger than threshold, classify it; otherwise, generate leaf node; if n instances are put to the current node and n of them belong to class i, also the most likely class is A, then the association between A and n_i shows below; the difference measure of current node is $(1-n_A/n)$.

Secondly, since some samples can't be classified further in current attribute space, even if there are plentiful training data allocated to the current node, such sample data can't be divided further. This is too extreme that all training samples' discrete attribute is the same in current attribute space. Therefore, the proposed incremental decision tree algorithm produces one Bayes node, to learn valuable attributes of current instance sample subset.

The algorithm for generating initial decision tree is defined like:

GenerateTree(S,P,D)

Where S is sample instance set; P is current attribute space; D is difference threshold.

Input: sample instance set S, attribute space P and difference threshold D;

Output: decision tree T

The algorithm is carried out in following steps:

(1) Create node M

(2) If sample instances in S are in the one class B, return M as common leaf node and mark as class B; go to next step;

(3) Calculate difference degree β of current node;

(4) If there is no available attribute value pair or $\beta < D$ in S, the node needs learn about Bayes classifier's parameter; then M is defined as one Bayes node, return M.

(5) Choose from S the attribute value pair A with the smallest expectation of distribution and find out A's all sample training set S_1;

(6) Add one node generated by GenerateTree(S_1 ,P-A,D); make $S_2 = S - S_1$,add one node generated by GenerateTree(S_2 ,P-A,D) as well.

So, by constantly is to establish a decision tree. This can be increased by adding a Bayesian node or modified Bayesian parameters to carry out incremental learning of samples, to solve the incremental learning problem.

The incremental decision tree algorithm overall steps are shown below:

Input parameters: decision tree T_0 . New sample instance a

Output parameter: decision tree T

The algorithm is carried out wholly as follows:

(1) Use new added sample instance a and decision tree T_0 ; attribute matching reaches one leaf node P;

(2) If the leaf node P is a Bayes node, merge with a to modify parameter of Bayes node and return the updated decision tree T;

(3) If the leaf node is not Bayes node, then it's a common leaf node; if the class of instance a is same with the node, go back to decision tree T_0 ; otherwise compare node P's Bayesian classification accuracy A and decision tree accuracy A_2 ;

(4) If $A_1 > A_2$, change node P to Bayes node; return updated T; or turn to T_0

8. Experimental Analysis and Results

The proposed incremental decision tree made use of some data sets in database for UCI machine learning study to do comparative testing of the algorithm. It compares the precision with decision tree algorithm and Bayes classification algorithm, comparing accuracy and time consuming of the two algorithms from ordinary non-incremental and incremental perspective, validating their classifying performance.

UCI is more famous for data mining and machine learning for researchers to use an open number. According to the library, in this paper, we use some data sets of UCI to complete the experiment.

Experiment parameter setting: the difference threshold value is 0.01. The minimum sample number of nodes is 2. In experiment, the average is divided into 10 samples, which 9 samples are used as the training set, and the other one is the test set. Conduct five experiments to get the average accuracy rate.

Experimental machine configuration: Pentium(R) Dual-Core CPU E6700 3.2GHZ, memory is 4GB. The specific information used in the UCI data set is as follows.

Table 1. UCI Data Set Details

Data set	Data size	Number of attributes	Class number
Zoo	100	16	7
Banding	167	20	2
Monk1	156	6	2
Monk2	168	6	2
Vote	340	15	2
Crx	480	15	2
Soybean	680	34	18
Anneal	800	37	6
Hypo	2600	28	5

Letter	10000	15	24

We used the data set to do non-incremental learning of the proposed incremental decision algorithm, ID3 algorithm and Bayes algorithm to get experimental data results as indicated in Table 2. From the table, we note the classifying accuracy rate of incremental decision tree algorithm increased by 4.67% than ID3 method, and by 9.34% than the Bayesian algorithm. The incremental decision tree algorithm presented here realized higher precision rate than the other two algorithms.

Table 2. Comparison of Incremental Decision Tree Algorithm and Algorithm and Bias Algorithm

Data set	Time consuming(ms)			Accuracy rate (%)		
	HID	ID3	Bayes	HID	ID3	Bayes
Zoo	97	86	90	100	99	100
Banding	135	123	99	98.6	94.2	86.9
Monk1	95	108	91	96.8	83.9	80.8
Monk2	121	96	88	93.4	83.4	78.9
Vote	128	96	94	96.9	95.7	90
Crx	151	153	112	95.5	95.1	90
Soybean	283	242	212	97.1	95.8	91.1
Anneal	336	294	147	99.4	95	96.1
Hypo	386	325	282	99.5	99.8	98.8
Letter	19385	14349	1682	94.2	92	74.2

We used the data set to perform incremental learning of the proposed incremental decision algorithm, ID3 algorithm and Bayes algorithm to get experimental data results as shown in Table 3. The table reveals that the proposed algorithm increased accuracy rate of classification by 2.7% than ID3 technique and by 8.26% then Bayes algorithm.

However we find shortcomings of the proposed algorithm. To be specific, the proposed algorithm took more expenses than id algorithm and Bayes algorithm, which has its necessity according to contents of the algorithm. The method here performs learning mainly of parameters of Bayesian classifier, as a result it takes more time.

Table 3. Time Overhead and Accuracy when Incremental Occurs

Data set	Incremental ratio	Time consuming(ms)		Accuracy rate (%)	
		HID	ID3	HID	ID3
Zoo	20%	53	86	93.2	91.3
Banding	30%	76	151	97.7	94.2
Monk1	30%	69	108	95.8	83.9
Monk2	35%	71	96	93.4	83.4
Vote	40%	61	98	96.2	95.7
Crx	25%	92	153	95.5	95.1
Soybean	45%	98	242	96.4	95.8
Anneal	30%	201	294	99.4	95
Hypo	50%	126	325	99.5	99.8
Letter	20%	5983	14349	93.2	92

To sum up from the above, the actual classifying accuracy of incremental decision tree algorithm is apparently better than Bayes algorithm and ID3 algorithm and can meet requirements of incremental learning. But it requires extra time consumption, which is comprehensible. On the whole, it can get rid of the drawbacks of incremental learning by decision tree algorithm, achieving expectant objective.

9. Conclusion

The incremental decision tree algorithm was presented in the paper. Firstly, it described fully the theory of Bayesian classifying method; then introduced steps and implementation of incremental decision tree algorithm; finally with some data sets used here, experiments were made to compare the proposed method, decision tree algorithm and Bayes method and results were obtained. Based on that, it analyzed the performance of the proposed incremental decision tree algorithm with those results. In conclusion with analysis of experimental data, the approach proved its ability to solve well incremental learning problem by combining with decision tree algorithm and Bayes classification method.

References

[1] L. Luo, F. X. Hong, "Data mining ID3 decision tree classification algorithm and its improved algorithm", Computer system application, vol. 10, **(2013)**, pp. 136-138.
[2] Z. Yuhong, H. Xuegang and L. Peipei, "A method of data stream classification based on the concept of anti noise", Journal of University of Science & Technology China, vol. 4, **(2011)**, pp. 347-352.
[3] W. Aiping, W. Guowei, C. Zhiquan and L. Sikun, "Incremental extreme random forest classifier for online learning", Journal of software, vol. 9, **(2011)**, pp. 2059-2074.
[4] J. L. Yuan, Z. Luo, Y. Guang, C. Mincheng, G. Juncheng and L. Tao, "Green data center incomplete energy consumption data fill and classification algorithm", Chinese Journal of computers, vol. 12, **(2015)**, pp. 2499-2516.
[5] F. Shaorong and X. Wenjun, "Improved algorithm based on decision tree sample selection", Journal of Southwest Jiao Tong University, vol. 5, **(2009)**, pp. 643-647.
[6] S. Luan, "Decision tree algorithm in the causes of juvenile delinquency. Shandong Administration Institute", Shandong Province Economic Management Cadre College Journal, vol. 4, **(2010)**, pp. 142-145.
[7] S. Chaoli, "Research decision tree data mining algorithm based on flow", Journal of Taiyuan University of Science and Technology, vol. 4, **(2006)**, pp. 268-272.
[8] Z. Wei and N. T. Shaohua, "Based on mutual information of fuzzy decision tree and its incremental learning", Jiangxi Normal University Journal (NATURAL SCIENCE EDITION), vol. 1, **(2014)**, pp. 89-94.
[9] W. Zhihai and W. Shuying, "Research on time series classification algorithm based on incremental decision tree", Modern computer (Professional Edition), vol. 8, **(2015)**, pp. 26-30.
[10] Z. Qing, "Research and improvement of decision tree classification algorithm", Zhengzhou University, **(2002)**.
[11] Y. Weiguo, "Research on the prediction methods of association rules and decision tree", Dalian Maritime University, **(2012)**.
[12] L. Peipei, "Concept drift detection and classification methods in data stream", HeFei University of Technology, **(2012)**.
[13] N. Wenying, "Application of improved ID3 decision tree classification algorithm in performance analysis", Dalian Jiaotong University, **(2008)**.
[14] L. Yonghua, "Research on attribute reduction and classification algorithms in data mining", Northwest Normal University, **(2009)**.
[15] W. Xiaoju, "Research on data mining algorithm based on rough set", Northwest Normal University, **(2009)**.
[16] O. Jishun, "Multiple classifier dynamic", integration technology of Jiangsu University, **(2010)**.
[17] H. Yuan, "Research on the classification algorithm of massive data mining based on cloud computing", Electronic Science and Technology University, **(2011)**.
[18] L. Lingli, "A survey of classification algorithms in data mining", Journal of Chongqing Normal University (NATURAL SCIENCE EDITION), vol. 4, **(2011)**, pp. 44-47.
[19] W. Gang, H. Lihua, Z. Chenghong and X. Jie, "Data mining classification algorithm Science & technology", vol. 12, **(2006)**, pp. 73-76.
[20] W. Star, "Optimization research and application of data mining algorithm", Anhui University, **(2014)**.
[21] C. Kai, "Comparison and analysis of data mining classification algorithms based on neural network", Anhui University, **(2014)**.
[22] Liang Yonglin. Research and application of data mining classification algorithm based on fusion of multiple classifiers. Chongqing University, 2007

[23] Zheng Mingchao. Comparative analysis of classification algorithms in data mining technology. Lanzhou University of Finance and Economics, 2007
[24] Zhou Xia. Study on classification algorithm of solar wind large data mining based on cloud computing. Chengdu University of Technology, 2014
[25] Kuangnan. Data mining classification and clustering algorithm and R language. Jinan University, 2007
[26] Guo Weixing. Classification algorithm of data mining. Zhejiang University, 2008
[27] Zhang Dandan. Classification and clustering algorithm for data stream mining. Beijing Jiaotong University, 2014
[28] Gao Dongling. Application of data mining classification algorithm in student information. Qiingdao University, 2009
[29] Zhang Junxi, Wu Xiaojun, Jiang Jianghong. Semi supervised clustering of complex distributed data. computer science and exploration, 2016,07:1003-1009.
[30] Tian Weidong, Xu Jingwen. Based on fuzzy equivalence classes of frequent item set is a condensed representation method. Application Research of computers, 2016,07:1936-1940.
[31] European Huisi, Cao Jian. For cross domain recommendation system research present situation and tendency . Mini micro computer systems, 2016,07:1411-1416.
[32] Wang Center, Sun Gang, Wang Hao. An integrated classification algorithm for noise and concept drift data stream . small and micro computer system, 2016,07:1445-1449.
[33] Liu Kan, Fan Qin. The link structure of the web page clustering research. small and micro computer system, 2016,07:1450-1454.
[34] Li blunt, Li Lun, Zhang Hangjin, Zheng Zhi, Cao Fuyuan. A graph clustering algorithm based on the structure and properties . small and micro computer system, 2016,07:1469-1473.
[35] Chen Lifei, Fan Yujie, Guo Gongde. A graph similarity measure of global structure matching. small and micro computer system, 2016,07:1488-1492.
[36] Sun Zhipeng, Qian Xue Zhong, Miss Qin, Tang Jie. Based on adaptive weighted distance calculation of rough k-means algorithm . Application Research of computers, 2016,07:1987-1990.
[37] Wuyu, Lei river. By generating a portfolio model microblogging topic event Abstract. Application Research of computers, 2016,07:2026-2029.
[38] Ge Luowen Town, Bai Song Liu, Wang Yangyang, Zhao Fuqing.. Study on the application of computer automatic classification based on the title of the class level information, 2016,07:2030-2033.
[39] Guo Hairong, Zhang Hui, Zhao Xujian, Li Bo, Yang Chunming. Dynamic multi document summarization algorithm based on incremental graph clustering. computer application research, 2016,07:2034-2038.
[40] Liu Yinghua. Research on Privacy Preserving Based on fuzzy set in data publishing. computer engineering and science, 2016,06:1118-1122.
[41] Gu Jun. Research and application of the K-means algorithm based on key sentences in the hot spot discovery field. Guizhou science, 2016,03:93-96.
[42] Wang Ling, Li Shulin, Wu Lulu. Classification and regression prediction algorithm based on quantitative association rule tree . Journal of Engineering Science, 2016,06:886-892.
[43] Zhong Zhe. Rough set decision tree classification algorithm research and improvement. Xiangtan University, 2014
[44] Liu Jingwen. Research on incremental text hierarchical classification based on SVM decision tree. Beijing University of Posts and Telecommunications, 2014
[45] Zou Lingjun. Clustering algorithm of data streams. Yangzhou University, 2008
[46] Liu Yunsheng. An improved decision tree incremental algorithm. Huazhong University of Science and Technology, 2007
[47] Feng Minmin. Temporal increment algorithm based on rough set and decision tree algorithm. Zhejiang University of Technology, 2009
[48] Pan Yongli. Improvement and application of decision tree classification algorithm. Yunnan University of Finance and Economics, 2011
[49] Liang Chunquan. Study on classification algorithm of uncertain data stream. Northwest Agriculture and Forestry University, 2014
[50] L. Hanbing, “Association rules algorithm in data mining”, Hebei University of Engineering, **(2007)**.
[51] Z. Chunfen, “Multi classifier combination and its application in medical image classification”, Jiangsu University, **(2007)**.
[52] L. Dongbiao, “Study on Algorithm and application of Wuhan University of Technology”, data mining based on decision tree, **(2008)**.
[53] H. Jibing, “Building and deploying combined classifier based on decision tree”, Zhejiang University of Technology, **(2008)**.
[54] W. Liming, “Decision tree learning and research”, pruning algorithm of Wuhan University of Technology, **(2007)**.
[55] C. Peiling, “Optimization of decision tree classification algorithm”, Central South University, **(2007)**.
[56] W. Tao, “Classification method of data stream mining classification key technology research”, National defense science and Technology University, **(2007)**.
[57] W. Yu, “Decision tree and K nearest neighbor algorithm”, Tianjin University, **(2006)**.

[58] W. Yahong, "Parallel classification algorithm for streaming data", Qufu Normal University, **(2015)**.
[59] Q. Rui, "Research and improvement of data stream mining algorithm", Nanjing University of Posts and Telecommunications, **(2015)**.
[60] Z. Dongyan, "The decision tree construction algorithm based on rough set theory", HeFei University of Technology, **(2006)**.

Author

Wenyi Xu. He received his M.S degree from Jiangxi Normal University. He is a Senior Engineer in Heyuan Polytechnic. His research interests include Network Security.