ISSN:0975-9646

Efficient Processing of Decision Tree Using ID3 & improved C4.5 Algorithm

Ms. Sonal Patil.
ComputerScienceEngg
G.H.R.I.E.M.,Jalgaon.
Jalgaon

Mr .Mayur Agrawal.
ComputerScienceEngg
G.H.R.I.E.M.,Jalgaon.
Jalgaon

Ms. Vijaya R. Baviskar
ComputerScienceEngg
G.H.R.I.E.M.,Jalgaon.
Jalgaon

Abstract-Decision Support Systems (DSS) are a specific class of computerized information system that supports business and organizational decision-making activities. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions. A decision support system (DSS) is a computer program application that analyzes business data and presents it so that users can make business decisions more easily. Decision tree learning algorithm has been successfully used in expert systems in capturing knowledge. The main task performed in these systems is using inductive methods to the given values of attributes of an unknown object to determine appropriate classification according to decision rules by using C4.5 algorithm. Historically ID3 algorithm was used to construct the Decision tree. ID3 algorithm is to construct the decision tree by employing a top-down, greedy search through the given sets to test each attribute at every tree node. In order to select the attribute that is most useful for classifying a given sets, we introduce a metric information gain. C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier. At the end of stage ID3 is compare with C4.5 by improving Prediction accuracy & computational complexity of DT.

Index Terms—Decision tree system, ID3 & C4.5 algorithm

1.INTRODUCTION

Classification is one of the most common tasks in data mining to solve a wide range of real problems, such as credit scoring, bankruptcy prediction, medical diagnosis, pattern recognition, and multimedia classification. It is recognized as a powerful way for companies to develop effective knowledge based decision models to gain competitive advantages. The current classification methods include decision trees (DTs), neural networks, logistic regression, and nearest neighbor. DTs are widely used for classification because of their ability to manage noisy data and return well organized results that are easily interpreted, computationally efficient, and robust. Several algorithms, such as ID3 and C4.5, have been devised for the construction of DTs and are applied in several areas, including risk management, customer relationship management, text categorization, medical diagnosis, and credit rating of loan applicants[1]. In traditional DT algorithms, a label (target variable) of a tuple is either categorical or Boolean that is, the algorithms operate under the assumption that the labels are nominal and flat. However, several practical situations involve more complex classification scenarios, in which the label to be predicted can occur in a variety of types, for example, continuous variable, hierarchically related variable, or both. To bridge this gap in the research, Hu et al. proposed a method called the continuous label classifier (CLC) algorithm, which added a novel dynamic partition mechanism to partition continuous labels during the construction of a DT[2]. For hierarchy related problems with class labels, Chen et al. proposed an hierarchical class label classifier (HLC) algorithm that can construct a DT from data with hierarchical class labels. They also demonstrated the unsuitability of traditional methods for hierarchical class labels and showed the differences resulting from their use. The HLC and CLC algorithms address several crucial problems associated with the induction of DT from data with either continuous or hierarchical labels. However, both algorithms have the following substantial limitations: 1) The HLC algorithm can be used only for class labels and is unable to manage continuous labels.

2) for the CLC algorithm, if the partitioned ranges are excessively fragmentary because of extensive variation in continuous-label distribution, they are not useful in prediction accuracy or an understanding of the induced models. This condition makes us difficult to use these rules to form business strategies.

In addition, current algorithms cannot directly manage a situation in which the labels are continuous in the data and can be naturally organized as a hierarchical structure. This type of problem is encountered in several domains, including the insurance industry, functional genomics, supply chains, risk management, and object recognition. In functional genomics, biologists create a set of possible gene functions that are subsequently organized hierarchically. A hierarchical structure is necessary because each gene may have multiple functions. It is crucial to obtain an interpretable model to understand the interactions between various genes. In each of these cases, the labels are continuous in the data and can be naturally organized as a hierarchical structure of class labels, which defines an abstraction over class labels. To the best of our knowledge, current research has not adequately addressed the issue of data classification using hierarchical continuous labels or the construction of DTs for hierarchical continuous-label classification. We used a simple example to illustrate the concept underlying the construction of DTs using hierarchical continuous labels and explain the differences that may occur because of hierarchical continuous labels[2].

Structured continuous-label classification is a variety of classification in which the label is continuous in the data, but the goal is to classify data into classes that are a set of predefined ranges and can be organized in a hierarchy. In the hierarchy, the ranges at the lower levels are more specific and inherently more difficult to predict, whereas the ranges at the upper levels are less specific and inherently easier to predict. Therefore, both prediction specificity and prediction accuracy must be considered when building a decision tree (DT) from this kind of data[3]. This method proposes a novel classification algorithm for learning DT classifiers from data with structured continuous labels. This approach considers the distribution of labels throughout the hierarchical structure during the construction of trees without requiring discretization in the preprocessing stage. We compare the results of the proposed method with those of the C4.5 algorithm using eight real data sets. The empirical results indicate that the proposed method outperforms the C4.5 algorithm with regard to prediction accuracy, prediction specificity, and computational complexity.

1.1 Decision Tree Induction

Decision tree induction is a very popular and practical approach for pattern classification. Decision tree induction is the learning of decision trees from class labelled training tuples. A decision tree is a flow chart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node holds a class label.

The decision tree classifier has two phases:

i) Growth phase or Build phase.

ii) Pruning phase.

The tree is built in the first phase by recursively splitting the training set based on local optimal criteria until all or most of the records belonging to each of the partitions bearing the same class label. The tree may overfit the data[3]. The pruning phase handles the problem of over fitting the data in the decision tree. The prune phase generalizes the tree by removing the noise and outliers. The accuracy of the classification increases in the pruning phase. Pruning phase accesses only the fully grown tree. The growth phase requires multiple passes over the training data. The time needed for pruning the decision tree is very less compared to build the decision tree.

The table specified below represents the usage frequency of various decision tree algorithms .

Table 1.1 Frequency usage of decision tree algorithms

Algorithm	Usage frequency (%)
CLS	9
IDE	68
IDE3+	4.5
C4.5	54.55
C5.0	9
CART	40.9
Random Tree	4.5
Random Forest	9
SLIQ	27.27
PUBLIC	13.6
OCI	4.5
CLOUDS	4.5

1.1.1 ID3 (Iterative Dichotomiser 3)

This is a decision tree algorithm introduced in 1986 by Quinlan Ross. It is based on Hunts algorithm. The tree is constructed in two phases. The two phases are tree building and pruning ID3 uses information gain measure to choose the splitting attribute. It only accepts categorical attributes in building a tree model. It does not give accurate result when there is noise. To remove the noise pre-processing technique has to be used.

To build decision tree, information gain is calculated for each and every attribute and select the attribute with the highest information gain to designate as a root node. Label the attribute as a root node and he possible values of the attribute are represented as arcs[4]. Then all possible outcome instances are tested to check whether they are falling under the same class or not. If all the instances are falling under the same class, the node is represented with single class name, otherwise choose the splitting attribute to classify the instances. Continuous attributes can be handled using the ID3 algorithm by discretizing or directly, by considering the values to find the best split point t by taking a threshold on the attribute values. Id3does not support pruning.

1.1 .2 C4.5

This algorithm is an extension to ID3 developed by Quinlan Ross. It is also based on Hunt's algorithm.C4.5 handles both categorical and continuous attributes to build a decision tree.In order to handle continuous attributes, C4.5 splits the attribute values into two partitions based on the selected threshold such that all the values above the threshold as one child and the remaining as another child. It also handles missing attribute values. C4.5 uses Gain Ratio as an attribute selection measure to build a decision tree. It removes the biasness of information gain when there are many outcome values of an attribute. At first, calculate the gain ratio of each attribute. The root node will be the attribute whose gain ratio is maximum. C4.5 uses pessimistic pruning to remove unnecessary branches in the decision tree to improve the accuracy of classification[3].

1.1.3 CART

CART stands for Classification And Regression Trees introduced by Breiman. It is also based on Hunt's algorithm. CART handles both categorical and continuous attributes to build a decision tree. It handles missing values. CART uses Gini Index as an attribute selection measure to build a decision tree .Unlike ID3 and C4.5 algorithms, CART produces binary splits. Hence, it produces binary trees. Gini Index measure does not use probabilistic assumptions like ID3, C4.5. CART uses cost complexity pruning to remove the unreliable branches from the decision tree to improve the accuracy.

2. History

Now-a-days the data stored in a database and which is used for application is huge. This explosive growth in data and database has generated an urgent need for new techniques and tools that can intelligently automatically transform the processed data into useful information and knowledge. Hence data mining has become a research are with increasing importance. Classification and prediction are two forms of data analysis that can be used to extract models describing important data classis or to predict future data trends. Such analysis can help provide us with better understanding of the data at large. Whereas classification predicts categorical (discrete, unordered) labels, prediction models continuous valued functions. Many classification and prediction methods have been proposed by researcher in machine learning pattern recognition and statistics. Most algorithms are memory resident, typically assuming a small data size. Recent data mining research has built on such work, developing scalable classification and prediction techniques capable of handling large disk-resident data.

In classification, the cases are placed in differing groups. The procedures behind this methodology create rules as per training and testing individual cases. A number of algorithms have been developed for classification based data mining. Some of them include decision tree, k-Nearest Neighbor, Bayesian and Neural-Net based classifiers[5]. At present, the decision tree has become an important data mining method. The basic learning approach of decision tree is greedy algorithm, which use the recursive top-down approach of decision tree structure. Quinlan in 1979 put forward a well-known Iterative Dichotomiser 3 algorithm, which is the most widely used algorithm in decision tree. But that algorithm has a defect of tending to select attributes with many values. It has also problem of over classification which leads to have less accuracy.

Several challenges are facing researchers and developer. Therefore, several papers and articles have tried to cover these issues. Such as Improvement on ID3 Algorithm, Reoptimization of ID3, C4.5 algorithm.

2.2 Literaure Survey

2.2.1 Data Mining

Data mining is the process of discovering interesting knowledge, such as associations, patterns, changes, significant structures and anomalies, from large amounts of data stored in databases or data warehouses or other information repositories. It has been widely used in recent years due to the availability of huge amounts of data in electronic form, and there is a need for turning such data into useful information and knowledge for large applications[3]. These applications are found in fields such as Artificial Intelligence, Machine Learning, Market Analysis, Statistics and Database Systems, Business Management and Decision Support.

2.2.2 Classification

Classification is a data mining technique that maps data into predefined groups or classes. It is a supervised learning method which requires labelled training data to generate rules for classifying test data into predetermined groups or classes. It is a two-phase process. The first phase is the learning phase, where the training data is analyzed and classification rules are generated[3]. The next phase is the classification, where test data is classified into classes according to the generated rules. Since classification algorithms require that classes be defined based on data attribute values, we had created an attribute “class” for every student, which can have a value of either “Pass” or “Fail”.

2.2.3. Clustering

Clustering is the process of grouping a set of elements in such a way that the elements in the same group or cluster are more similar to each other than to those in other groups or clusters . It is a common technique for statistical data analysis used in the fields of pattern recognition, information retrieval, bioinformatics, machine learning and image analysis[4]. Clustering can be achieved by various algorithms that differ about the similarities required between elements of a cluster and how to find the elements of the clusters efficiently. Most algorithms used for clustering try to create clusters with small distances among the cluster elements, intervals, dense areas of the data space or particular statistical distributions.

2.2.4 Selecting Classification over Clustering

In clustering, classes are unknown apriori and are discovered from the data. Since our goal is to predict students’ performance into either of the predefined classes “Pass” and “Fail”, clustering is not a suitable choice and so we have used classification algorithms instead of clustering algorithms.

2.3 Issues Regarding Classification

2.3.1. Missing Data

Missing data values cause problems during both the training phase and to the classification process itself. For example, the reason for non-availability of data may be due to :

- Equipment malfunction
- Deletion due to inconsistency with other recorded data
- Non-entry of data due to misunderstanding
- Certain data considered unimportant at the time of entry
- No registration of data or its change

This missing data can be handled using following approaches :

- Data miners can ignore the missing data
- Data miners can replace all missing values with a single global constant
- Data miners can replace a missing value with its feature mean for the given class
- Data miners and domain experts, together, can manually examine samples with missing values and enter a reasonable, probable or expected value

The chances of getting missing values in the training data are very less. The training data is to be retrieved from the admission records of a particular institute and the attributes considered for the input of classification process are mandatory for each student. The tuple which is found to have missing value for any attribute will be ignored from training set as the missing values cannot be predicted or set to some default value[6]. Considering low chances of the occurrence of missing data, ignoring missing data will not affect the accuracy adversely.

2.3.2. Measuring Accuracy

Determining which data mining technique is best depends on the interpretation of the problem by users. Usually, the performance of algorithms is examined by evaluating the accuracy of the result. Classification accuracy is calculated by determining the percentage of tuples placed in the correct class. At the same time there may be a cost associated with an incorrect assignment to the wrong class which can be ignored.

2.4 ID3 Algorithm

In decision tree learning, ID3 (Iterative Dichotomiser 3) is an algorithm invented by Ross Quinlan used to generate a decision tree from the dataset. ID3 is typically used in the machine learning and natural language processing domains. The decision tree technique involves constructing a tree to model the classification process[8]. Once a tree is built, it is applied to each tuple in the database and results in classification for that tuple. The following issues are faced by most decision tree algorithms :

- Choosing splitting attributes
- Ordering of splitting attributes
- Number of splits to take
- Balance of tree structure and pruning
- Stopping criteria

The ID3 algorithm is a classification algorithm based on Information Entropy, its basic idea is that all examples are mapped to different categories according to different values of the condition attribute set; its core is to determine the best classification attribute form condition attribute sets. The algorithm chooses information gain as attribute selection criteria; usually the attribute that has the highest information gain is selected as the splitting attribute of current node, in order to make information entropy that the divided subsets need smallest[6]. According to the different values of the attribute, branches can be established, and the process above is recursively called on each branch to create other nodes and branches until all the samples in a branch belong to the same category. To select the splitting attributes, the concepts of Entropy and Information Gain are used.

2.4.1. Entropy

Given probabilities p1, p2, …, ps, where pi = 1, Entropy is defined as

H(p1, p2, …, ps) = - (pi log pi)

Entropy finds the amount of order in a given database state. A value of H = 0 identifies a perfectly classified set[7]. In other words, the higher the entropy, the higher the potential to improve the classification process.

2.4.2 Information Gain

ID3 chooses the splitting attribute with the highest gain in information, where gain is defined as difference between how much information is needed after the split. This is calculated by determining the differences between the entropies of the original dataset and the weighted sum of the entropies from each of the subdivided datasets. The formula used for this purpose is:

G(D, S) = H(D) - P(Di)H(Di).

2.5 C4.5

C4.5 is a well-known algorithm used to generate a decision trees. It is an extension of the ID3 algorithm used to overcome its disadvantages. The decision trees generated by the C4.5 algorithm can be used for classification, and for this reason, C4.5 is also referred to as a statistical classifier. The C4.5 algorithm made a number of changes to improve ID3 algorithm . Some of these are:

- Handling training data with missing values of attributes
- Handling differing cost attributes
- Pruning the decision tree after its creation
- Handling attributes with discrete and continuous values

Let the training data be a set S = s1, s2 ... of already classified samples. Each sample Si = xl, x2... is a vector where xl, x2 ... represent attributes or features of the sample.

The training data is a vector

C = c1, c2..., where c1, c2... represent the class to which each sample belongs to.

At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits data set of samples S into subsets that can be one class or the other. It is the normalized information gain (difference in entropy) that results from choosing an attribute for splitting the data. The attribute factor with the highest normalized information gain is considered to make the decision. The C4.5 algorithm then continues on the smaller sub-lists having next highest normalized information gain.

3. Problem definition

The ability of software systems in supporting the decision making process.

Decision Support Systems (DSS) are a specific class of computerized information system that supports business and organizational decision-making activities. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions.

A decision support system (DSS) is a computer program application that analyzes business data and presents it so that users can make business decisions more easily.

4. Methodology & working

In existing decision support system, proved two theorems presenting the McDiarmid’s bound for both the information gain, used in ID3 algorithm, and for Gini index, used in Classification and Regression Trees (CART) algorithm. The results of the work guarantee that a decision tree learning system, applied to data streams and based on the McDiarmid’s bound, has the property that its output is nearly identical to that of a conventional learner.

4.1 Working

Historically, the first decision tree algorithm was ID3 (Iterative Dichotomiser), later C4.5 and Classification and Regression Trees algorithms were developed. In the beginning they were applied to solve problems characterized by static data. In most algorithms, e.g., ID3, entropy was taken as a measure of the impurity in a collection of training examples. As a measure of the effectiveness of an attribute in classifying the training data, the so-called information gain was used. Its interpretation is the expected reduction in entropy caused by splitting the examples according to this attribute. So we use c4.5 algorithm to improve the efficiency of the decision tree as compare to ID3 algorithm.

4.2 Methodology

DECISION MAKING TREES

This system functioning is based on the implementations of the decision making trees and operations perform on the basis of it.

4.2.1 Decision Tree Algorithm:-

Decision tree learning algorithm has been successfully used in expert systems in capturing knowledge. The main task performed in these systems is using inductive methods to the given values of attributes of an unknown object to determine appropriate classification according to decision rules.

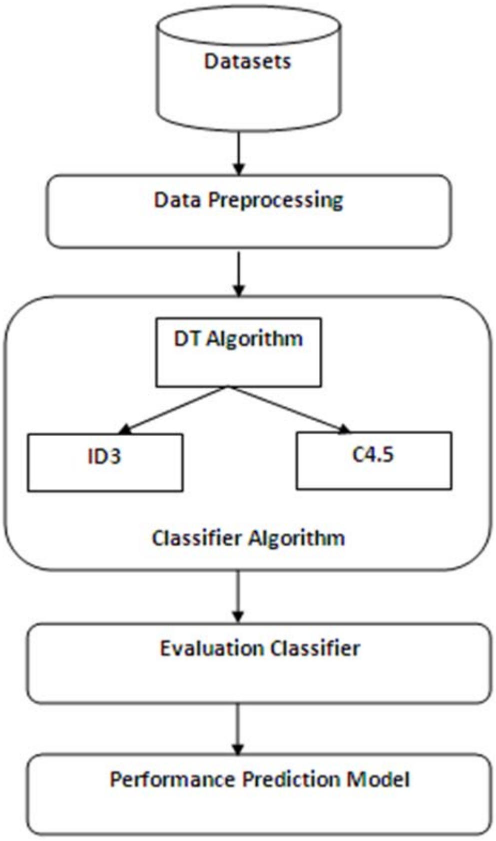

Fig. 4.1 Processing model

We examine the decision tree learning algorithm ID3, C4.5 and implement this algorithm using Java programming. We first implement basic ID3 and C4.5 in which we dealt with the target function that has discrete output values.

1. ID3

ID3 is a simple decision tree learning algorithm developed by Ross Quinlan (1983). The basic idea of ID3 algorithm is to construct the decision tree by employing a top-down, greedy search through the given sets to test each attribute at every tree node. In order to select the attribute that is most useful for classifying a given sets, we introduce a metric---information gain.

2. C4.5

C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier.

Advantages of C4.5 over ID3:-

- Pruning trees after creation - C4.5 goes back through the tree once it's been created and attempts to remove branches that do not help by replacing them with leaf nodes.
- Handling training data with missing attribute values - C4.5 allows attribute values to the goal is to make the algorithm as accurate as practically possible.
- To reduce space complexity we are partitioning the data set into n partitions where n = (Total no of rows)/15
- We have ensured that no exception is directly thrown on the EJB container and every exception is converted into user comprehensible message.

4.2.2 Attribute Selection Measure

The information gain measure is used to select the test attribute at each node in the tree. The attribute with highest information gain is chosen as test attribute for the current Node. This attribute minimizes the information needed to classify the samples in resulting partition and reflect the least “impurity” in these partitions.

Let S be set consisting of s data sample. Suppose the class label attribute has m Distinct values defining m distinct class. Ci (for i =1... m). Lets Si be the number of Sample of S in class Ci. The expected information needed to classify a given sample is given by equation (2.1)

I (S1, S2, • • • , Sm) = −m X i pilog2(pi) ------------ (2.1)

Where Pi is probability that an arbitrary sample belongs to classify Ci and estimated by si/s. Note that a log function to base 2 is used since the information in encoded in bits .

Entropy:

It is minimum number of bits of information needed to encode the classification of arbitrary members of S.

Lets attribute A have v distinct value a1,............., av.Attribute A can be used to Partition S into v subsets, S1, S2,........, Sv, where Sj contains those samples in S that have value aj of A. If A were selected as the test attribute, then these subset would corresponds to the branches grown from the node contains the set S. Let Sij be the number of class Ci, in a subset by Sj, The entropy or expected information based on partitioning into subset by A, is given by equation (2.2)

E(A) =vXj Sij + • • • + Smj S I(Sij + • • • + Smj) (2,2)

The first term acts as the weight of the jth subset and is the number of samples in the subset divided by the total number of sample in S. The smaller the entropy value, the greater purity of subset partitions as shown in fig (2.1).

I(S1, S2, • • • , Sm) = −mXi pilog2(pi)------- (2.3)

Where Pi is the probability that a sample in Sj belongs to class Ci.

Information Gain

It is simply the expected reduction in entropy caused by partitioning the examples. According to the attribute. More precisely the information gain, Gain(S, A) of an attribute A, relative collection of examples S, is given by equation (2.4)

Gain (A) = I (S1, • • •, Sm) − E (A) ------- (2.4)

In other words gain (A) is the expected reduction in entropy caused by knowing the Value of attribute A. The algorithm computes the information gain of each attribute. With highest information gain is chosen as the test attribute for a given set.

5.APPLICATIONS

5.1 The applications of decision tree in various areas :

The decision tree algorithms are largely used in all area of real life. The application areas are listed below

1. Business: Decision trees are use in visualization of probabilistic business models. It also use in customer relationship management and uses for credit scoring for credit card users.
2. Intrusion Detection: Decision trees use for generate genetic algorithms to automatically generate rules for an intrusion detection expert system. Abbes et al. proposed protocol analysis in intrusion detection using decision tree. Energy Modeling: Energy modeling for buildings is one of the important tasks in building design. Decision tree is use for it.
3. E-Commerce: Decision tree is widely use in e-commerce. It is use for generated online catalog which is essence for the success of an e-commerce web site. Image Processing: Park et al. proposed perceptual grouping of 3-D features in aerial image using decision tree classifier. Macarthur et al. proposed use of decision tree in content-based image retrieval.
4. Medicine: Medical research and practice are the important areas of application for decision tree techniques. Decision tree is most useful in diagnostics of various diseases. It also use for Heart sound diagnosis.
5. Industry: Production quality control (faults identification), non-destructive tests are areas where decision tree algorithm is useful.
6. Intelligent Vehicles: The job of finding the lane boundaries of the road is important task in development of intelligent vehicles. Gonzalez and Ozguner proposed lane detection for intelligent vehicles by using decision tree.
7. Remote Sensing: Remote sensing is a strong application area for pattern recognition work with decision trees. Some researcher proposed algorithm for classification for land cover categories in remote sensing. And binary tree with genetic algorithm for land cover classification.
8. Web Applications Chen et al. presented a decision tree learning approach to diagnosing failures in large Internet sites. Bonchi et al. proposed decision trees for intelligent web caching.

CONCLUSION:

According to our observations, the performances of the algorithms are strongly depends on the entropy, information gain and the features of the data sets. There are various work has been done using the Decision tree Algorithm. But they all are like Static in Nature. Some recent improve algorithm reduce problem like replication, handle continuous data, biased to multi value attribute ,and continous label classification.

This work can be extended in one direction, i.e., datatype extensions, structure-type extensions, or learning process extensions. The data can be extended to include ordinal data, which have mixed characteristics of categorical and numerical data. In practice, it may be necessary to predict multiple numerical labels, such as stock prices, profits, and revenue simultaneously.

1. The Application can be extended to implement the Dataset extractor which can directly extract the datasets in XML format from warehouse schema.

2. Both the algorithms can be published on the web as web services so as to share the logic with other applications.

REFERENCES:

[1] Hsiao-Wei Hu, Yen-Liang Chen, and Kwei Tang," A Novel Decision-Tree Method for Structured Continuous-Label Classification", IEEE Transactions On Cybernetics, Vol. 43, No. 6, December 2013.

[2] Leszek Rutkowski, Fellow, IEEE, Lena Pietruczuk, Piotr Duda, and Maciej Jaworski," Decision Trees for Mining Data Streams Based on the McDiarmid's Bound", IEEE Transactions On Knowledge And Data Engineering, Vol. 25, No. 6, June 2013.

[3] Mr. Brijain R Patel, 2Mr. Kushik K Rana," A Survey on Decision Tree Algorithm For Classification", IJEDR 2014.

[4] J. R. Quinlan," Improved Use of Continuous Attributes in C4.5", Journal of Artcial Intelligence Research 4 (1996) 77-90.

[5] Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha and Vipul Honrao," Predicting students performance using ID3 and C4.5 classification algorithms", International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.5, September 2013.

[6] Ihsan A. Kareem , Mehdi G. Duaimi ," Improved Accuracy for Decision Tree Algorithm Based on Unsupervised Discretization", IJCSMC, Vol. 3, Issue. 6, June 2014.

[7] Boris Delibašić, Member, IEEE, Milan Vukićević, Miloš Jovanović, and Milija Suknović." White-Box or Black-Box Decision Tree Algorithms: Which to Use in Education?", IEEE Transactions On Education, Vol. 56, No. 3, August 2013.

[8] Nishant Mathur, Sumit Kumar, Santosh Kumar, and Rajni Jindal," The Base Strategy for ID3 Algorithm of Data Mining Using Havrda and Charvat Entropy Based on Decision Tree", International Journal of Information and Electronics Engineering, Vol. 2, No. 2, March 2012.