Analysis of Various Decision Tree Algorithms for Classification in Data Mining

Bhumika Gupta, PhD
Assistant Professor,
C.S.E.D
G.B.P.E.C, Pauri
Uttarakhand, India

Aditya Rawat
B.Tech IV Year
G.B.P.E.C, Pauri
Uttarakhand, India

Akshay Jain
B.Tech IV Year
G.B.P.E.C, Pauri
Uttarakhand, India

Arpit Arora
B.Tech IV Year
G.B.P.E.C, Pauri
Uttarakhand, India

Naresh Dhami
B.Tech IV Year
G.B.P.E.C, Pauri
Uttarakhand, India

ABSTRACT
Today the computer technology and computer network technology has developed so much and is still developing with pace.Thus, the amount of data in the information industry is getting higher day by day. This large amount of data can be helpful for analyzing and extracting useful knowledge from it. The hidden patterns of data are analyzed and then categorized into useful knowledge. This process is known as Data Mining. [4].Among the various data mining techniques, Decision Tree is also the popular one. Decision tree uses divide and conquer technique for the basic learning strategy. A decision tree is a flow chart-like structure in which each internal node represents a "test" on an attribute where each branch represents the outcome of the test and each leaf node represents a class label. This paper discusses various algorithms of the decision tree (ID3, C4.5, CART), their features, advantages, and disadvantages.

Keywords
Decision Tree, ID3, C4.5, Entropy, Information Gain.

1. INTRODUCTION
Data mining comprises extracting information from a data set and transforming it to a structure that is understandable [4].It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics and database systems. Data mining allows you to sift through all the chaotic and repetitive noise in your data. It also helps to understand the relevant information and make good use of that information to assess likely outcomes. Thus data mining accelerates the pace of making informed decisions. There are six classes in data mining namely Anomaly Detection, Association Rule Learning, Clustering, Classification, Regression. Classification is a data mining function that assigns items in a collection to target categories or classes. Classification aims at predicting the target class for each case in the data. For example, a classification model can help to identify bank loan applications as safe or risky. The various classification techniques used in the field of data mining are decision tree induction, rule-based method, memory-based learning, Bayesian networks, neural networks and support vector machines.

The most widely applied supervised classification technique is Decision Tree. Decision Tree induction comprises of learning and classification. These steps are simple and fast and thus Decision Tree can be applied to any domain [10]. The goal of the decision tree is to form a model and then predict the value of a target variable by giving several inputs. A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible outcomes. A decision tree is a classifier in the form of a tree structure where each node is either:

- Leaf node- Leaf node is an indicator of the value of target attribute(class) of examples, or
- A decision node- A decision node specifies all possible tests on a single attribute-value, with one branch and sub-tree for each possible outcome of the test [6].

2. CONCEPT
2.1 Theory
Decision Tree learning is one of the most widely used and practical methods for inductive inference over supervised data. Based on various attributes a decision tree represents a procedure that classifies the categorical data. Besides this decision tree are also for processing a large amount of data and thus find use in data mining applications. For constructing decision trees no domain knowledge or parameter setting is required. Thus decision trees are sufficient as well as appropriate for exploratory knowledge discovery and their representation of acquired knowledge in tree form is intuitive and easy to understand.

2.2 Why use Decision Trees
- Decision trees can be visualized and are simple to understand and interpret.
- They require very little data preparation whereas other techniques often require data normalization, the creation of dummy variables and removal of blank values.
- The cost of using the tree (for predicting data) is logarithmic in the number of data points used to train the tree.
- Decision trees can handle both categorical and numerical data whereas other techniques are specialized for only one type of variable.
- Decision trees can handle multi-output problems.
- Uses a white box model i.e. the explanation for the condition can be explained easily by Boolean logic because there are mostly two outputs. For example yes or no.

- Decision trees can perform well even if assumptions are somewhat violated by the dataset from which the data is taken.

2.3 Types of Decision Trees

Decision trees used in data mining are mainly of two types:

- **Classification tree** in which analysis is when the predicted outcome is the class to which the data belongs. For example outcome of loan application as safe or risky.
- **Regression tree** in which analysis is when the predicted outcome can be considered a real number. For example population of a state.

Both the classification and regression trees have similarities as well as differences, such as procedure used to determine where to split.

There are various decision trees algorithms namely ID3(Iterative Dichotomiser 3), C4.5, CART(Classification and Regression Tree), CHAID(CHi- squared Automatic Interaction Detector), MARS. Out of these, we will be discussing the more popular ones which are ID3, C4.5, CART.

2.3.1 ID3(Iterative Dichotomiser)

ID3 is an algorithm developed by Ross Quinlan used to generate a decision tree from a dataset [12]. To construct a decision tree, ID3 uses a top-down, greedy search through the given sets, where each attribute at every tree node is tested to select the attribute that is best for classification of a given set [10]. Therefore, the attribute with the highest information gain can be selected as the test attribute of the current node. ID3 is based on Occam's razor. In this algorithm, small decision trees are preferred over the larger ones. However, it does not always construct the smallest tree and is, therefore, a heuristic algorithm [6].

For building a decision tree model, ID3 only accepts categorical attributes. Accurate results are not given by ID3 when there is noise and when it is serially implemented. Therefore data is preprocessed before constructing a decision tree [1]. For constructing a decision tree information gain is calculated for each and every attribute and attribute with the highest information gain becomes the root node. The rest possible values are denoted by arcs. After that, all the outcome instances that are possible are examined whether they belong to the same class or not. For the instances of the same class, a single name class is used to denote otherwise the instances are classified on the basis of splitting attribute.

2.3.1.1 Advantages of ID3

- The training data is used to create understandable prediction rules.
- It builds the fastest as well as a short tree.
- ID3 searches the whole dataset to create the whole tree.
- It finds the leaf nodes thus enabling the test data to be pruned and reducing the number of tests.
- The calculation time of ID3 is the linear function of the product of the characteristic number and node number [9].

2.3.1.2 Disadvantages of ID3

- For a small sample, data may be over-fitted or over-classified.
- For making a decision, only one attribute is tested at an instant thus consuming a lot of time.
- Classifying the continuous data may prove to be expensive in terms of computation, as many trees have to be generated to see where to break the continuum.
- One disadvantage of ID3 is that when given a large number of input values, it is overly sensitive to features with a large number of values [2].

2.3.2 C4.5

C4.5 is an algorithm used to generate a decision tree which was also developed by Ross Quinlan. It is an extension of Quinlan's ID3 algorithm. C4.5 generates decision trees which can be used for classification and therefore C4.5 is often referred to as statistical classifier [11]. It is better than the ID3 algorithm because it deals with both continuous and discrete attributes and also with the missing values and pruning trees after construction. C5.0 is the commercial successor of C4.5 because it is a lot faster, more memory efficient and used for building smaller decision trees.C4.5 performs by default a tree pruning process. This leads to the formation of smaller trees, more simple rules and produces more intuitive interpretations.

C4.5 follows three steps in tree growth [3]:

- For splitting of categorical attributes, C4.5 follows the similar approach to ID3 algorithms. Continuous attributes always generate binary splits.
- Selecting attribute with the highest gain ratio.
- These steps are repeatedly applied to new tree branches and growth of the tree is stopped after checking of stop criterion. Information gain bias the attribute with more number of values. Thus, C4.5 uses Gain Ratio which is a less biased selection criterion.

2.3.2.1 Advantages of C4.5

- C4.5 is easy to implement.
- C4.5 builds models that can be easily interpreted.
- It can handle both categorical and continuous values.
- It can deal with noise and deal with missing value attributes.

2.3.2.2 Disadvantages of C4.5

- A small variation in data can lead to different decision trees when using C4.5.
- For a small training set, C4.5 does not work very well.

2.3.3 CART

It stands for Classification And Regression Trees. It was introduced by Breiman in 1984. CART algorithm builds both classification and regression trees. The classification tree is constructed by CART by the binary splitting of the attribute. Gini Index is used as selecting the splitting attribute. The CART is also used for regression analysis with the help of regression tree. The regression feature of CART can be used in forecasting a dependent variable given a set of predictor variable over a given period of time. CART have an average speed of processing and supports both continuous and nominal attribute data.

2.3.3.1 Advantages of CART

- CART can handle missing values automatically using surrogate splits.
- Uses any combination of continuous/discrete variables.
- CART automatically performs variable selection.
- CART can establish interactions among variables.

- CART does not vary according to the monotonic transformation of predictive variable [7].

2.3.3.2 Disadvantages of CART

- CART may have unstable decision trees.
- CART splits only by one variable.
- Non-parametric.

2.3.4 Random Forest

Random Forest decision tree was developed by Leo Breiman [13]. A Random Forest is a collection of simple tree predictors, such that each tree produces a response when a set of predictor values are given as input. Similar to CART algorithm. Random Forest also works both for classification and regression problems. When solving classification problems, the response or the output appears in the form of a class membership, which associates or classifies, a set of independent predictor values with the matching category present in the dependent variable. When solving regression problems the output or response of the tree is an approximation of the dependent variables given the predictors.

Random Forest is a bagging tool that leverages the ability of multiple varied analyses, organization strategies, and ensemble learning to supply correct models, perceptive variables, importance ranking and laser-sharp coverage on the record-by-record basis for deep data understanding [8].

2.3.4.1 Advantages of Random Forest

- It recognizes outliers and anomalies in knowledgeable data.
- It is one of the most accurate learning algorithms available. For many datasets, it produces highly accurate classifiers.
- It gives an estimate of the important variables in classification.

2.3.4.2 Disadvantages of Random Forest

- Sometimes the classification made by Random Forests are difficult to be interpreted by humans.
- Random Forest sometimes overfits with datasets with noisy classification/regression tasks.

3. ATTRIBUTE SELECTION MEASURES

For selecting the splitting criterion that "best" separates the data partition, D, of class-labeled training tuples into individual classes, we used attribute selection measure which is heuristic for such selection. If we were to split D into smaller partitions according to the outcomes of the splitting criterion, ideally each partition would be pure(i.e., all the tuples that fall into a given partition would belong into the same class) [5]. The result of this scenario is actually the "best" splitting criterion of all the criteria taken. Attribute selection measure determines how to split the tuples at a given node and are therefore also known as splitting rules.

The splitting attributes can be continuous-valued or it can be restricted to binary trees. For continuous-valued attributes, a split point must be determined as part of the splitting criterion whereas for the binary trees a splitting subset must be determined. The tree node for partition is labeled with the splitting criterion, branches are grown for each outcome of criterion and the tuples are partitioned accordingly. The most popular attribute selection measures are – Entropy (Information Gain), Gain Ratio and Gini Index.

3.1 Entropy

Entropy is a measure of uncertainty associated with a random variable. The entropy increases with the increase in uncertainty or randomness and decreases with a decrease in uncertainty or randomness. The value of entropy ranges from 0-1.

$$Entropy(D)=\sum_{i=1}^{c} -p_i \log_2(p_i)$$

where p_i is the non-zero probability that an arbitrary tuple in D belongs to class C and is estimated by $|C_{i,D}|/|D|$. A log function of base 2 is used because as stated above the entropy is encoded in bits 0 and 1.

3.2 Information Gain

ID3 uses information gain as its attribute selection measure. Claude Shannon studied the value or "information content" of messages and gave information gain as a measure in his Information Theory [5]. Information Gain is the difference between the original information gain requirement (i.e. based on just the proportion of classes) and the new requirement (i.e. obtained after the partitioning of A).

$$Gain(D,A)=Entropy(D)-\sum_{j=1}^{v} \frac{|Dj|}{|D|} Entropy(D_j)$$

Where,

D: A given data partition

A: Attribute

V: Suppose we partition the tuples in D on some attribute A having v distinct values

D is split into v partition or subsets, $\{D_1,D_2,..D_j\}$ were D_j contains those tuples in D that have outcome a_j of A.

The attribute that has the highest information gain is chosen.

3.3 Gain Ratio

The information gain measure is biased towards tests with many outcomes. That is it prefers to select attributes having a large number of values. As each partition is pure, the information gain by partitioning is maximal. But such partitioning cannot be used for classification.

C4.5(a successor of ID3) uses this attribute selection measure named Gain Ratio which is an extension to the information gain. Gain Ratio differs from information gain, which measures the information with respect to a classification that is acquired based on some partitioning [5]. Gain Ratio applies kind of information gain using a "split information" value defined as:

$$SplitInfo_A = -\sum_{j=1}^{v} \frac{|Dj|}{|D|} \log_2\left(\frac{|Dj|}{|D|}\right)$$

The Gain Ratio is then defined as:

$$Gain\ Ratio(A)=\frac{Gain\ (A)}{SplitInfoA\ (D)}$$

A splitting attribute is selected which is the attribute having the maximum Gain Ratio. The gain ratio becomes unstable if the split information tends to 0. A constraint is added to avoid such condition, whereby the information gain of test selected must be large- at least as great as the average gain over all tests examined [5].

3.4 Gini Index

Gini Index is an attribute selection measure used by the CART decision tree algorithm. The Gini Index measures the impurity D, a data partition or set of training tuples as:

$$Gini(D)=1-\sum_{i=1}^{m} p_i^2$$

Where p_i is the probability that a tuple in D belongs to class Ci and is estimated by $|C_{i,D}|/|D|$. The sum is computed over m classes. The attribute that reduces the impurity to the maximum level (or has the minimum gini index) is selected as the splitting attribute.

4. ILLUSTRATION SHOWING ATTRIBUTESELECTION MEASURES

In this paper, we have used the database of an Electronic store to see whether a person buys a laptop or not. Figure 1 shows table having class-labeled training tuples from the electronic store. Each attribute taken is of a discrete value. The class-labeled attribute buys _laptop, has two distinct values (yes, no). Therefore there are two distinct classes and the value of m is equal to 2.

We assume:

- Class P: buys_laptop = "yes"
- Class N:buys_laptop="no"

As there are 9 yes and 5 no in the buys_laptop attribute, therefore 9 tuples belong to class P and 5 tuples belong to class N.

Entropy is calculated as:

$$Entropy(D)= -\frac{9}{14}log_2\left(\frac{9}{14}\right)-\frac{5}{14}log_2\left(\frac{5}{14}\right)=0.940$$

Now the information gain is calculated as:

$$Gain(age,D)=Entropy(D)-$$

$$\sum_{v\in\{youth,middle-aged,Senior\}}\frac{|Sv|}{|S|}Entropy(S_v)$$

$$=Entropy(D)-\frac{5}{14}Entropy(S_{youth})-\frac{4}{14}Entropy(S_{middle-aged})-$$

$$\frac{5}{14}Entropy(S_{senior})$$

$$Gain(salary,D)=0.029$$

$$Gain(graduate,D)=0.151$$

$$Gain(credit\_raiting,D)=0.048$$

Calculation of Gain Ratio:

Firstly the SplitInfo is calculated. The salary attribute splits the data in Figure 1 into three partitions high, low and medium.

$$SplitInfo_{salary}(D)=-\frac{4}{14}xlog_2\left(\frac{4}{14}\right)-\frac{6}{14}xlog_2\left(\frac{6}{14}\right)-\frac{4}{14}xlog_2\left(\frac{4}{14}\right)$$

=1.557

As the Gain(salary)=0.029.

Therefore, GainRatio(salary)= 0.029/1.557 = 0.019

Calculation of Gini Index:

The Gini Index to compute the impurity of D for the database in Figure 1 is:

$$Gini(D)=1-\left(\frac{9}{14}\right)^2-\left(\frac{5}{14}\right)^2=0.459$$

The Gini Index value computed based on the salary attribute partitioning is given as:

$$Gini_{salary\ \epsilon\ (low,medium)}(D)=\frac{10}{14}Gini(D_1)+\frac{4}{14}Gini(D_2)$$

$$=\frac{10}{14}\left(1-\left(\frac{7}{10}\right)^2-\left(\frac{3}{10}\right)^2\right)+\frac{4}{14}\left(1-\left(\frac{2}{4}\right)^2-\left(\frac{2}{4}\right)^2\right)$$

=0.443

$$=Gini_{salary\ \epsilon\{high\}}(D).$$

We calculated the Gini index values for the other subsets also and the result was 0.458 for the subset({low, high} and {medium}) and it was 0.450 for the subset({medium, high} and {low}).

Therefore, the best binary split for salary attribute was found to be on ({low, medium} or {high}) because it minimizes the Gini index and has the value of 0.443.

The attribute age when split over the subset({youth, senior}) gives the minimum Gini index overall, with a reduction in impurity of 0.459-0.357 = 0.102. Now according to the Gini index, the binary split "age ∈ {youth, senior?} becomes the splitting criterion as it results in the maximum reduction in impurities of tuples in D.

Thus, the database of Electronics store shows that the attribute age has the maximum or highest Information Gain and that the age attribute also has the minimum Gini index, therefore, resulting in a maximum reduction in impurity of the tuple in this. Thus the decision tree for the given data is formed in Figure 3. by taking age as the splitting attribute.

ID	age	salary	graduate	credit_rating	class: buys_laptop
1	youth	high	no	average	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	average	yes
4	senior	medium	no	average	yes
5	senior	low	yes	average	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	average	no
9	youth	low	yes	average	yes
10	senior	medium	yes	average	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	average	yes
14	senior	medium	no	excellent	no

Figure 1. Class-labeled labeled training tuples from the Electronics Store database.

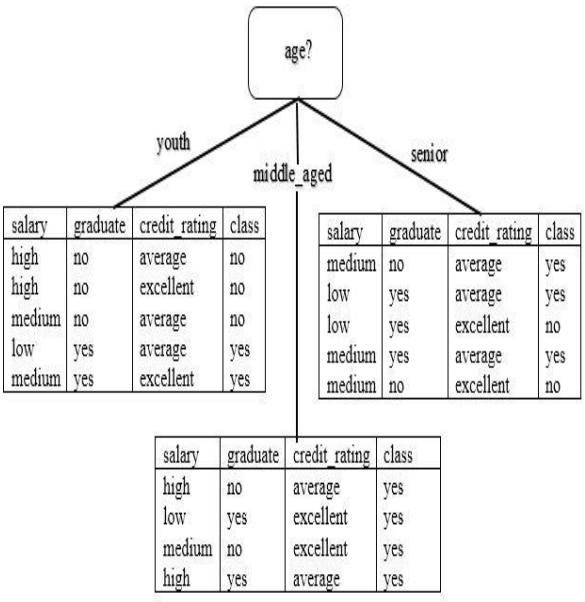

Figure 2. Tuples for the Electronic Store database partitioned according to the attribute age.

The attribute age has the highest information gain and thus becomes the splitting attribute at the root node of the decision tree. Branches are grown for each outcome of age. These tuples are shown partitioned according to the age.

A decision tree for the concept buys _laptop, indicating whether a customer at an electronic store is likely to buy a laptop or not is shown in Figure 3. Each internal (non-leaf) node of the decision tree represents a test on an attribute. Each leaf node of the decision tree represents a class (either buys_laptop=”yes” or busy _laptop=” no”).

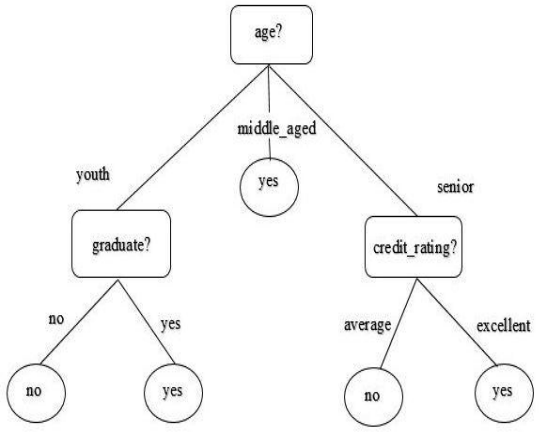

Figure 3. A decision tree for the concept buys _laptop in an electronic store.

5. APPLICATIONS OF DECISION TREES IN VARIOUS AREAS OF DATA MINING

The various decision tree algorithms find a large application in real life. Some areas of application include:

- **E-Commerce**: Used widely in the field of e-commerce, decision tree helps to generate online catalog which is a very important factor for the success of an e-commerce website.
- **Industry**: Decision Tree algorithm is very useful for producing quality control(faults identification) systems.
- **Intelligent Vehicles**: An important task for the development of intelligent vehicles is to find the lane boundaries of the road. Gonzalez and Ozguner have proposed lane detection for intelligent vehicles using decision trees.
- **Medicine**: Decision Tree is an important technique for medical research and practice. A decision tree is used for diagnostic of various diseases. And is also used for hard sound diagnosis.
- **Business**: Decision Trees also find use in the field of business where they are used for visualization of probabilistic business models, used in CRM(Customer Relationship Management) and used for credit scoring for credit card users and for predicting loan risks in banks.

6. CONCLUSION

This paper analyses various decision tree algorithms that are used in data mining. We found that each algorithm has got its own advantages and disadvantages as per our study. The efficiency of various decision tree algorithms can be analyzed based on their accuracy and the attribute selection measure used. The efficiency of the algorithms also depends on the time taken information of the decision tree by the algorithm. We found that both C4.5 and CART are better than ID3 when missing values are to be handled whereas ID3 cannot handle missing or noisy data. But we also analyzed that ID3 produces faster results. The paper also gives an idea of the attribute selection measure used by various decision trees algorithms like ID3 algorithm uses information gain, the C4.5 algorithm uses gain ratio and CART algorithm uses GINI Index as the attribute selection measure. The paper also gives the methods for calculation of these attribute selection measures. In all, we find that these algorithms for decision tree induction are to be used at different times according to the situation.

7. REFERENCES

[1] Anuj Rathee and Robin Prakash Mathur, “Survey on Decision Tree Classification algorithms for the evaluation of Student Performance”,(IJCT),ISSN:2277-3061,March-April,2013.

[2] Badr HSSINA, Abdelkarim MERBOUHA, Hanane EZZIKOURI, Mohammed ERRITALI,“A comparative study of decision tree ID3 and C4.5”, (IJACSA).

[3] Devinder Kaur, Rajiv Bedi and Dr. Sunil Kumar Gupta, “Implementation of Enhanced Decision Tree Algorithm on Traffic Accident Analysis”, (IJSRT), ISSN: 2379-3686, 15th September 2015.

[4] G.Kesavaraj, Dr. S.Sukumaran, “A Study On Classification Techniques in Data Mining”, IEEE-31661, July 4-6, 2013.

[5] Han J., Kamber M., and Pei J. (2012) Data Mining: Concepts and Techniques, 3rd edition. The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor.

[6] Hemlata Chahal, “ID3 Modification and Implementation in Data Mining”, International Journal of Computer Applications (0975-8887), Volume 80-No7, October 2013.

[7] Jatinder Kaur and Jasmeet Singh Gurm, “Optimizing the Accuracy of CART algorithm by Using Genetic Algorithm”, (IJST), Volume 3 Issue 4, Jul-Aug, 2015.

[8] M.S. Mythili and Dr. A.R.Mohamed Shanavas, “An Analysis of students’ performance using classification algorithms”, (IOSR-JCE), e-ISSN:2278-0661,p-ISSN:2278-8727, Jan. 2014.

[9] Qing-Yun Dai, Chun-ping Zhang and Hao Wu,“Research of Decision Tree Classification Algorithm in Data Mining”, International Journal of Database Theory and ApplicationVol.9, No.5(2016),pp.1-8.

[10] T.Miranda Lakshmi, A.Martin, R.Mumtaj Begum, and Dr. V.Prasanna Venkatesnan,“An Analysis on Performance of decision Tree Algorithms using Student’s Qualitative Data”, I.J. Modern Education and Computer Science, June 2013.

[11](2017, March 4), C4.5[Online].Available: http://en.wikipedia.org/wiki/C4.5_algorithm.

[12](2017, March 4), ID3[Online].Available: http://en.wikipedia.org/wiki/ID3_algorithm.

[13](2017, March 4), Random Forest[Online].Available:http://en.wikipedia.org/wiki/Random_Forest.