HE ID3 DECISION TREE ITHM IN WEKA

Sonia Singh
Dept. of Computer Science
University of Delhi
Mob. 9540512809
Address: 125 LIG DDA FLATS,PulPrahladPur, New Delhi-110044
14sonia.singh@gmail.com

Priyanka Gupta
Dept. of Computer Science
University of Delhi
Mob. 9716772205
Address: H.NO. 5-C,MIG Flats, VivekVihar, B-Block Phase-II,Delhi-110095
shinepriyanka@gmail.com

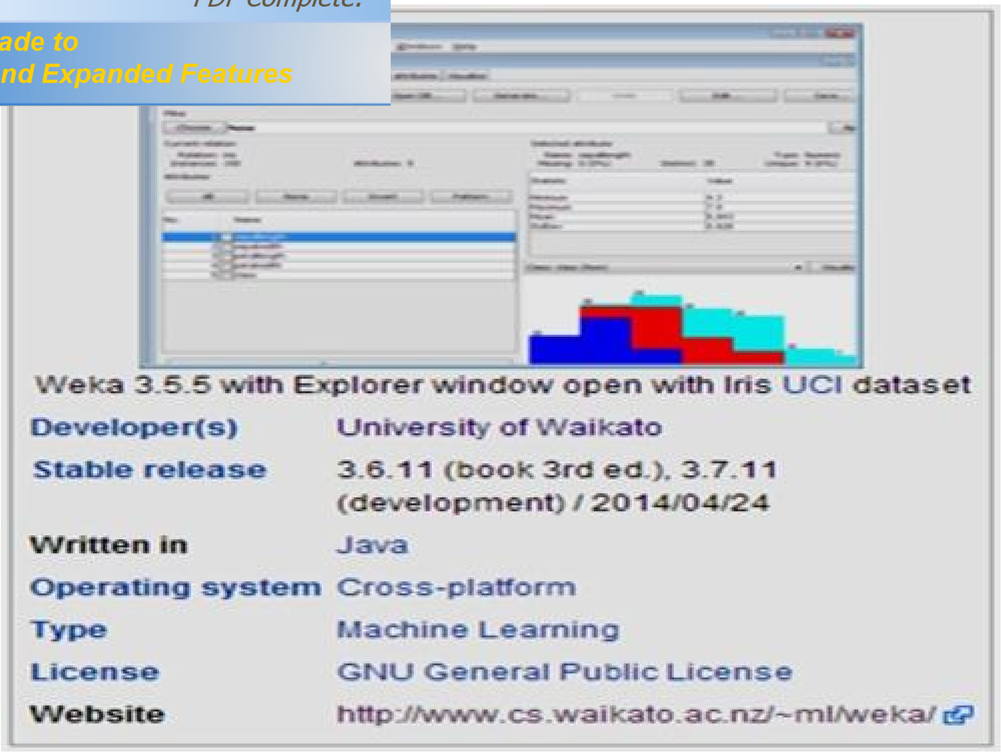
Manoj Giri
Dept. of Computer Science
GGSIP University
Mob. 9958232291
Address: Qtr NO. 954, Sector-7 PushpVihar, New Delhi-110017
manojgiri0889@gmail.com

Abstract— Data mining is used to extract the required data from large databases. The data mining algorithm is the mechanism that makes various models, trees and algorithm to extract the useful information or pattern or we can say knowledge from large database . To create a model, an algorithm first learns the rules from a set of data then looks for specific required patterns and trends according to those rules. Decision-tree learning is one of the most effective form to represent and evaluate the performance of algorithms, due to its various eye-catching features: simplicity, comprehensibility, no parameters, and being able to handle mixed-type data . ID3 is a simple decision tree erudition algorithm developed by Ross Quinlan (1983) . Decision tree helps to take the decision for better analysis of data for splitting the node to make the dataset homogeneous. Decision tree algorithm is used to select the best path to follow in the standard division. This paper introduces the use of ID3 algorithm of decision tree. It helps in taking the better decision to analyse the data. In this paper the ID3 decision tree learning algorithm is implemented with the help of example data vertebrates which is implemented in WEKA in ARFF (attribute relation file format) format .The resultant of the work will be the classified decision tree and the decision rules which are generated in the form of IT-THEN rules and then interpreted further.

Keywords:* *Data Mining, Decision Tree, Classification, ID3, WEKA, ARFF.

I. INTRODUCTION

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 tree rules. We have examined the decision tree learning algorithm ID3. We used a free data mining software available under the GNU General Public Licenseï **Weka** in figure 1[10] in which we have made the training records in ARFF (attribute relation file format) and then there training records are being examined by Weka.



Weka 3.5.5 with Explorer window open with Iris UCI dataset

Developer(s)	University of Waikato
Stable release	3.6.11 (book 3rd ed.), 3.7.11 (development) / 2014/04/24
Written in	Java
Operating system	Cross-platform
Type	Machine Learning
License	GNU General Public License
Website	http://www.cs.waikato.ac.nz/~ml/weka/

Figure:1 Weka Visualization

Weka contains a collection of visualization tools and algorithms for data analysis and predictive modelling, together with graphical user interfaces for easy access to this functionality. It is freely available under the GNU General Public License, portable since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform, comprehensive collection of data pre-processing and modelling techniques are available and easy to use because of the presence of graphical user interface.

Weka supports several standard data mining tasks, more specifically, data pre-processing features, clustering technique algorithms (k-means, farthest first, hierarchical etc), classification, regression, visualization, and feature selection. All of Weka's techniques are predicated on the assumption that the data is available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported).

II. DECISION TREE

A decision tree is a tree in which each branch node represents a choice between a number of alternatives, and each leaf node represents a decision. Decision tree are commonly used for gaining information for the purpose of decision -making. Decision tree starts with a root node on which it is for users to take actions. From this node, users split each node recursively according to decision tree learning algorithm. [4]The final result is a decision tree in which each branch represents a possible scenario of decision and its outcome. 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 tree rules

[4]. Decision trees classify instances by traverse from root node to leaf node. We start from root node of decision tree, testing the attribute specified by the node, and then moving down the tree branch according to the attribute value in the given set. This process is repeated to the sub-tree level. To make a decision, one starts at the root node, and asks questions to determine which arc to follow, until one reaches a leaf node and the decision is made.

This basic structure is shown in Figure 2.

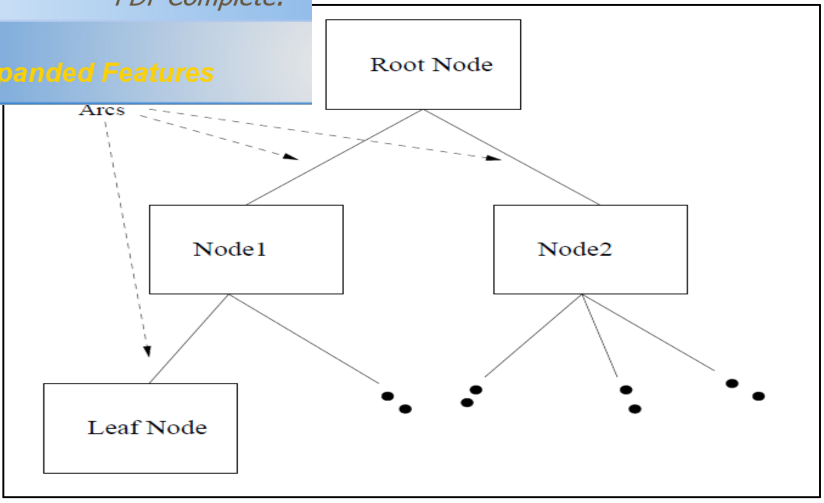


Figure: 2 Basic Decision Tree Structure

Decision tree learning algorithm is suited because [5]:
1. Instance is represented as attribute-value pairs. For example, attribute 'Body Temperature' and its value 'cold blooded' and 'warm blooded' Figure 3.

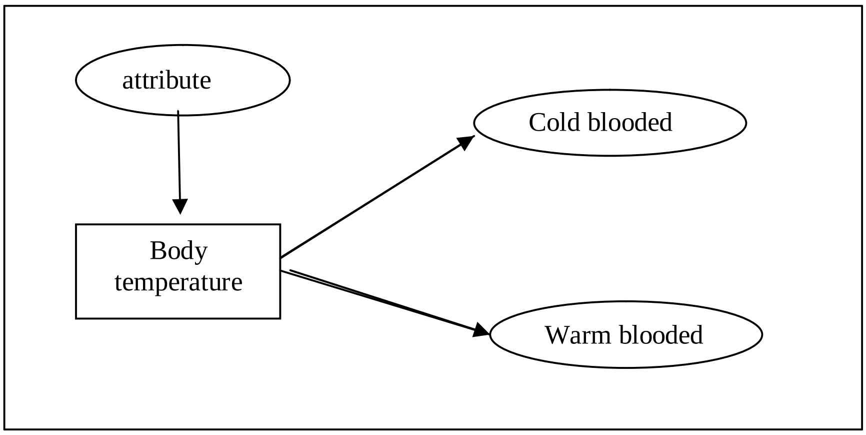


Figure:3 Attribute and its value

2. The target function or class label has discrete output values. It can easily deal with instance which is assigned to a Boolean decision, such as 'true' and 'false', 'p(positive)' and 'n(negative)' as an Output.

3. The training data may contain errors. This can be dealt with pruning techniques that we will not cover here.

The 3 widely used decision tree learning algorithms are:

1. ID3 2. CARTand 3. C4.5.
   We will cover ID3 in this report.

III. ID3 ALGORITHM

ID3(Iterative Dichotomizer) is a simple decision tree learning algorithm developed by Ross Quinlan (1983) which determines the classification of objects by testing the values of the their properties. It builds the tree in a top down fashion, starting from a set of objects and a specification of properties. At each node of the tree, a property is tested and the results used to partition the object set. This process is recursively done till the set in a given subtree is homogeneous with respect to the ontains objects belonging to the same category. This then e property to test is chosen based on information theoretic gain and minimize entropy. In simpler terms, that property the most homogeneous subsets. For choosing a splitting node we first need to calculate the Entropy of that node and then calculate the information gain from which we decide whether we have to split the node or not.

For computing Entropy [5]:
First, let us assume, without loss of generality, that the resulting decision tree classifies instances into two categories, we'll call them P (positive) and N (negative).
Given a set S, containing these positive and negative targets, the entropy of S related to this Boolean classification is:

Entropy(S) = {- P (positive) log2P (positive)-P (negative) log2P (negative)}

P (positive): proportion of positive examples in S
P (negative): proportion of negative examples in S
For example, if S is (0.5+, 0.5-) then Entropy(S) is 1, if S is (0.67+, 0.33-) then Entropy(S) is 0.92, if P is (1+, 0 -) then Entropy(S) is 0. Note that the more uniform is the probability distribution, the greater is its information. You may notice that entropy is a measure of the impurity in a collection of training sets.

Information Gain is then calculated to take the decision of splitting the node.

Info Gain (▽) = Entropy (parent node)- entropy (child node)

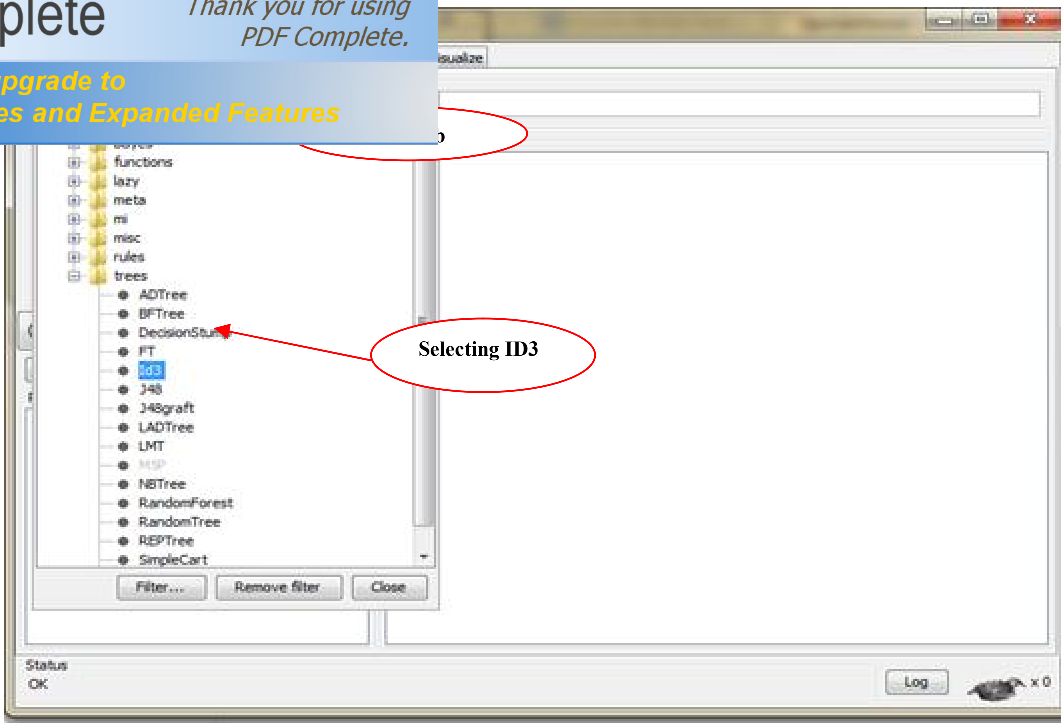
Let say in an example we can split on two attributes A and B .A has entropy value 0.487 and B has entropy value 0.376, so we will choose B as a splitting node because it has a larger Info gain which leads to more homogenous (pure) node[9].

IV. METHODOLOGY

We have applied ID3 algorithm using a well known Data Mining software available under GNU General Public License WEKA on two nominal types of data set Weather and Vertebrates. Weather is very popular and can be extracted from uci repository and vertebrates is a self made data set referenced from the book ñIntroduction to Data Miningò by Vipin Kumar, Pang-Ning Tan and Michael Steinbach from chapter number 4 ñClassification:ò[9]. Weather has 5 attributes and Vertebrates has 8 attributes. We applied ID3 algorithm in Weka figure 4 by choosing ñclassifyò tab and explore the ñtreesò and forms the decision tree. The result is then interpreted in terms of IF-THEN rules and an basic tree is then developed further to represent the path.

We will perform ID3 on 2 data sets in weka.

1. Weather data set
2. Vertebrates data set



Selecting ID3

Figure: 4 ID3 in Weka

V. EVALUATING THE DATA SET

1. Weather data set

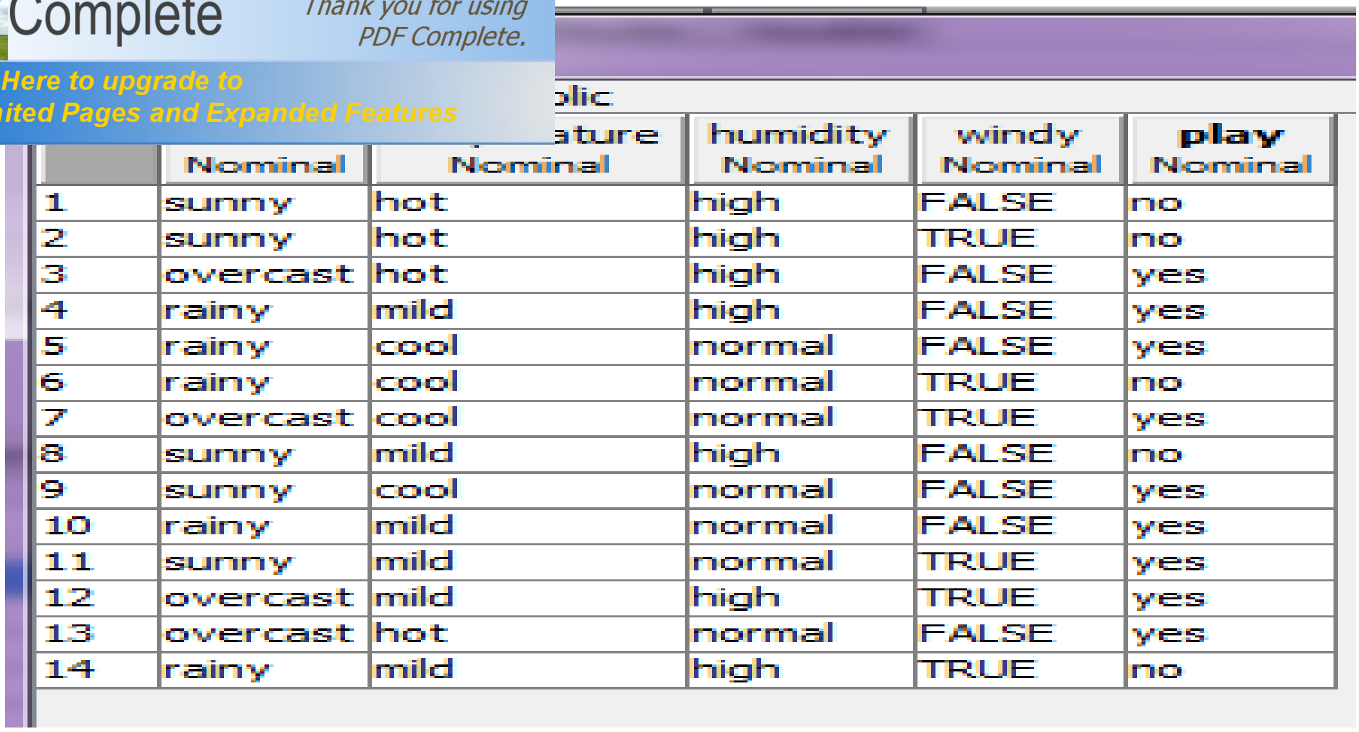
Weather is a very popular data set available in uci repositories which is used by many learning algorithms to evaluate the performance of various algorithms and trees also. It is presented in the figure 5 below in edit section of weka explorer.

The *run information* part contains general information about the scheme used, the number of instances (14) and attributes (5) as well as the attributes names as presented below

=== **Run information** ===

Scheme: weka.classifiers.trees.Id3
Relation: weather. Symbolic
Instances: 14
Attributes: 5
outlook
temperature
humidity
windy
play
Test mode: 10-fold cross-validation

Here, 10 fold cross validation is used as test mode in which the complete data set is divided into equal partitions and each single partition is then act as a test set and it is compared with the all other partitions which together act as training set. It is efficient as each partition is tested against all the training records.



blic

		ature	humidity	windy	play
	Nominal	Nominal	Nominal	Nominal	Nominal
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
3	overcast	hot	high	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
7	overcast	cool	normal	TRUE	yes
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
10	rainy	mild	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes
13	overcast	hot	normal	FALSE	yes
14	rainy	mild	high	TRUE	no

Figure: 5 Weather data set

The *Second* part of the output is represented by the ID3 decision tree as
== Classifier model (full training set) ===

Id3

outlook = sunny
| humidity = high: no
| humidity = normal: yes
outlook = overcast: yes
outlook = rainy
| windy = TRUE: no
| windy = FALSE: yes
Time taken to build model: 0.01 seconds

This tree is interpreted using the If-Then rules:

1. If (outlook=sunny AND humidity = high) THEN => will not play
2. If (outlook = sunny AND humidity = normal) THEN=>will play
3. If (outlook = overcast) THEN=> will play
4. If (outlook=rainy AND windy=true) THEN=> will not play
5. If (outlook=rainy AND windy=false) THEN=> WILL PLAY

The Decision tree of the above rules generated is represented in figure 6.

Beside this, Weka provides some complementary information about the percent of correctly as well as incorrectly classified instances. In this example, out of a total of 14 instances, 12 have been correctly classified meaning 85.714 %. This summary is presented below, along with some important statistical parameters. One of it is *Kappa statistic*, a measure of agreement between two individuals,

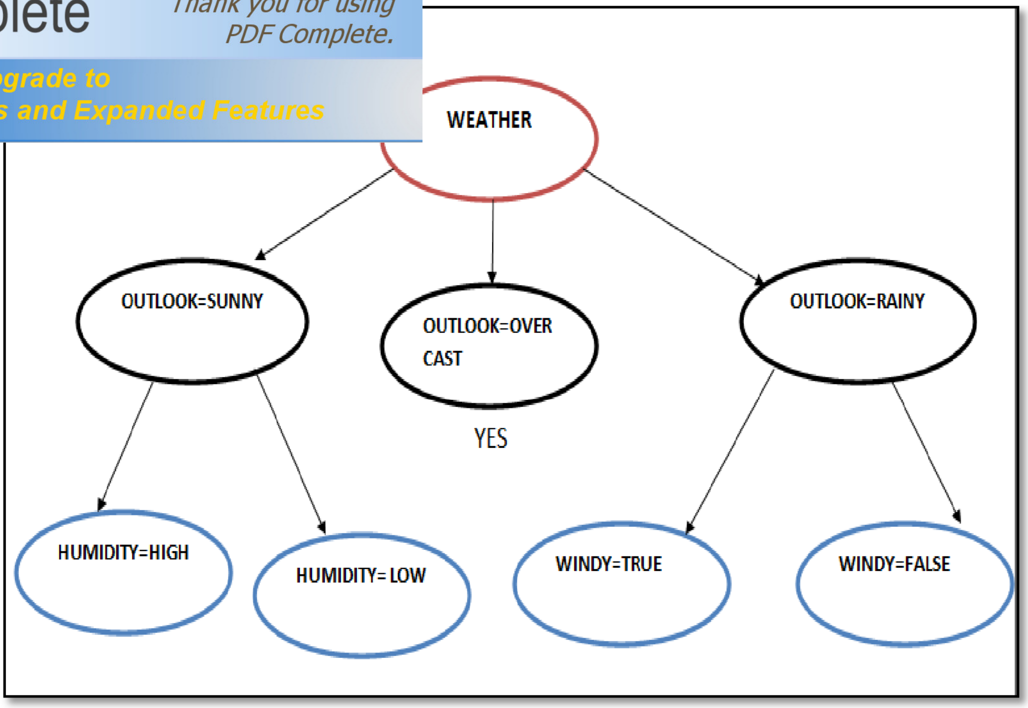


Figure: 6 Decision tree of Weather data set

with a 0.6889 value; other parameters are *mean absolute error*- a quantity used to measure how close forecasts or predictions are to the eventual outcomes, *rootmean squared error* - a good measure of the modelôs accuracy*, root relative squared error* -the average of the actual values, *relative absolute error* - similar to the relative squared error.

```
Correctly Classified Instances          12              85.7143 %
Incorrectly Classified Instances         2              14.2857 %
Kappa statistic                      0.6889
Mean absolute error                      0.1429
Root mean squared error                      0.378
Relative absolute error                 30      %
Root relative squared error                76.6097 %
Total Number of Instances               14
```

The fourth part of the output, presented below, contains information regarding the detailed accuracy by class. Here is detailed information concerning the next statistical parameters:

- *TP Rate (True positive rate)* ï the report of the positive instances classified as positive here it is 85.7%.
- *Precision* ï the number of correctly classified instances divided by the whole classified instances number is 0.857
- *Recall* ï the same with TP Rate
- *FP Rate (False Positive Rate)* – the report of the negative classified instances as positive is 16.8%
- *F-Measure* is a measure of a test's accuracy and is determined using the formula:

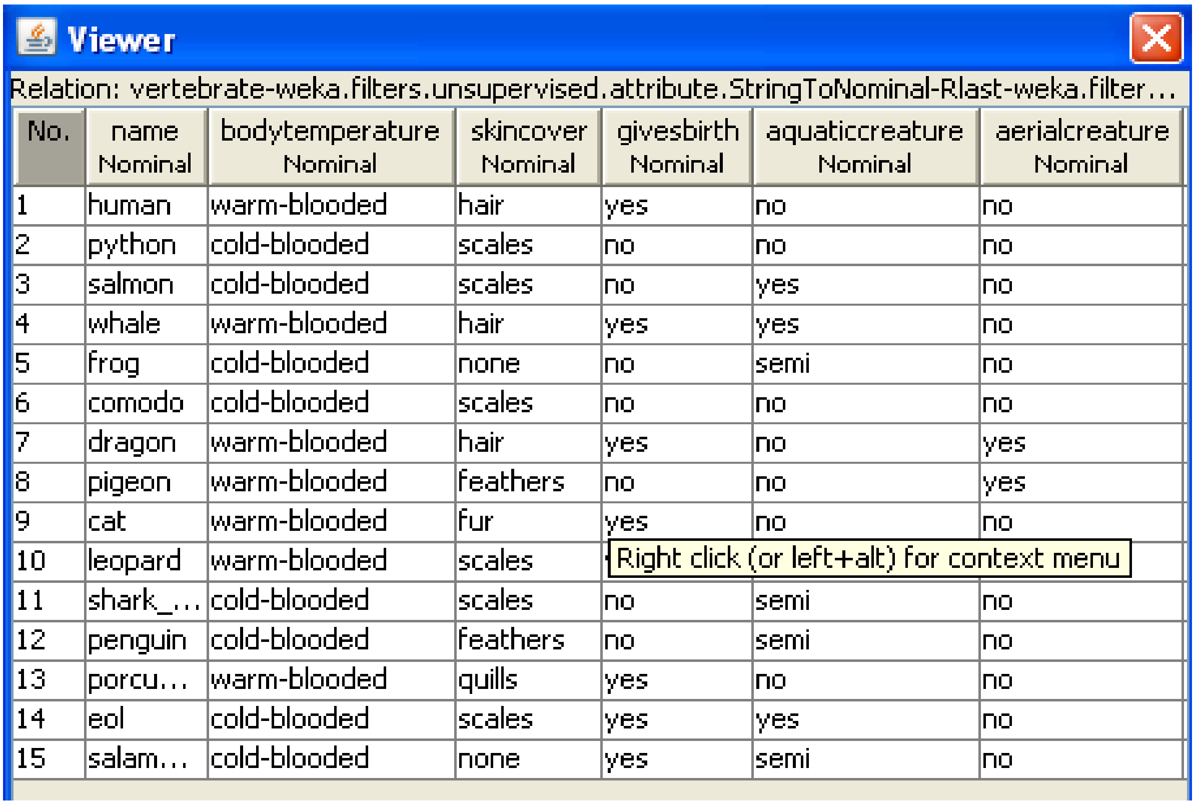
(2* TP Rate * Precision) / (TP Rate + Precision) here the value is 85.7%.

ROC Area (Receiver Operating Characteristic Area) ï The ROC curve is given by the TP Rate and FP Rate. The area under the ROC Curve (AUC) is a method of measuring the performance of the ROC curve. If AUC is 1 then the prediction is perfect; if it is 0.5 then the prediction is random. Here it is 0.844.

```
                                                easure   ROC Area   Class
                 0.889     0.2          0.889     0.889    0.889           0.844        yes
                 0.8       0.111        0.8       0.8      0.8             0.844        no
Weighted Avg.    0.857                  0.168     0.857    0.857           0.857        0.844
```

Vertebrates data set

The vertebrates data set is a self made data set taken from the reference of Data Mining book ñINTRODUCTION TO DATA MININGò by Vipin Kumar, Pang-Ning Tan and Michael Steinbach from chapter number 4 ñClassification:ò[9] . It is presented in the figure 7 below in edit section of weka explorer.



Relation: vertebrate-weka.filters.unsupervised.attribute.StringToNominal-Rlast-weka.filter...

No.	name Nominal	bodytemperature Nominal	skincover Nominal	givesbirth Nominal	aquaticcreature Nominal	aerialcreature Nominal
1	human	warm-blooded	hair	yes	no	no
2	python	cold-blooded	scales	no	no	no
3	salmon	cold-blooded	scales	no	yes	no
4	whale	warm-blooded	hair	yes	yes	no
5	frog	cold-blooded	none	no	semi	no
6	comodo	cold-blooded	scales	no	no	no
7	dragon	warm-blooded	hair	yes	no	yes
8	pigeon	warm-blooded	feathers	no	no	yes
9	cat	warm-blooded	fur	yes	no	no
10	leopard	warm-blooded	scales			
11	shark_...	cold-blooded	scales	no	semi	no
12	penguin	cold-blooded	feathers	no	semi	no
13	porcu...	warm-blooded	quills	yes	no	no
14	eol	cold-blooded	scales	yes	yes	no
15	salam...	cold-blooded	none	yes	semi	no

Right click (or left+alt) for context menu

Figure: 7Vertebrates data set

The *run information* part contains general information about the scheme used, the number of instances (15) and attributes (8) as well as the attributes names as presented below

=== Run information ===

Scheme: weka.classifiers.trees.Id3
Relation: vertebrate-weka.filters.unsupervised.attribute.StringToNominal-RFIRST-weka.filters.unsupervised.attribute.Remove-R1
Instances: 15
Attributes: 8
bodytemperature
Skincover
givesbirth

classlabel
Test mode: 10-fold cross-validation

The *Second* part of the output is represented by the ID3 decision tree as
=== Classifier model (full training set) ===
Id3
skincover = hair: mammal
skincover = scales
| aquaticcreature = yes: fish
| aquaticcreature = no: reptile
| aquaticcreature = semi: reptile
skincover = feathers: bird
skincover = fur: mammal
skincover = quills: mammal
skincover = none: amphibian

Time taken to build model: 0 seconds

Interpretation using the If-Then rules:

1. If (skin cover=hair)THEN =>mammal
2. If (skin cover=scales AND aquatic creature=yes)THEN => fish
3. If (skin cover=scales AND aquatic creature=no)THEN => reptile
4. If (skin cover=scales AND aquatic creature=semi)THEN => reptile
5. If (skin cover=feather)THEN=> bird
6. If(skin cover=fur)THEN => mammal
7. If(skin cover=quills)THEN=> mammal
8. If(skin cover=none) => amphibian

The Decision tree of the above rules generated is represented in figure 8.

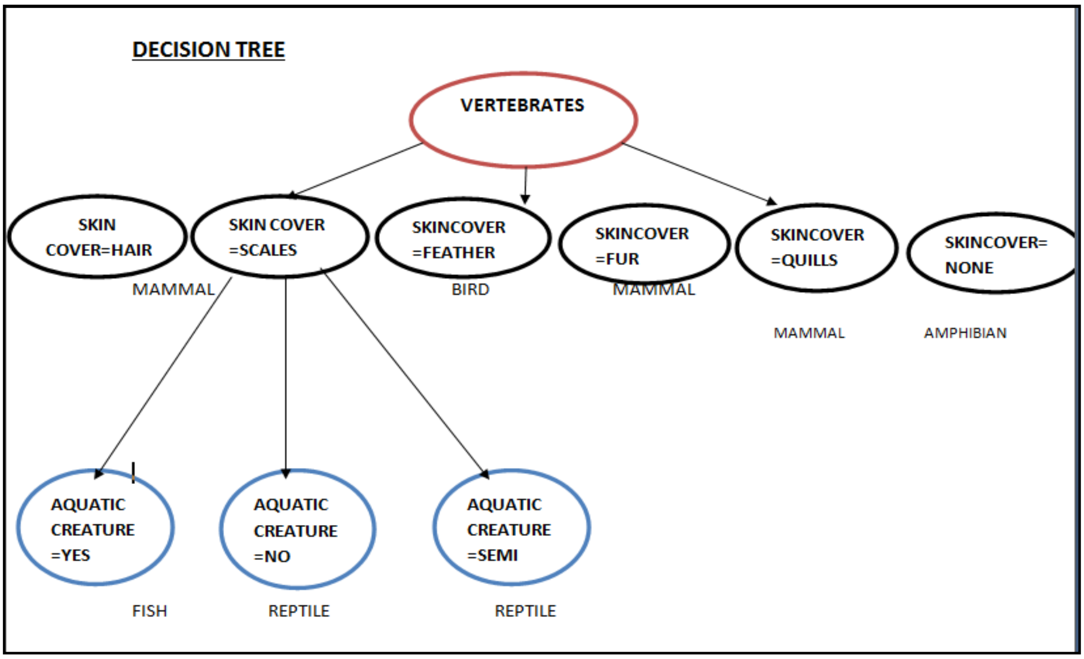


Figure 8: Vertebrates decision tree

```
Correctly Classified Instances          11            73.3333 %
Incorrectly Classified Instances         1             6.6667 %
Relative absolute error             12.5809 %
Root relative squared error            48.7896 %
Unclassified Instances                  3             20      %
Total Number of Instances               15
=== Detailed Accuracy by Class ===


 TP Rate          FP Rate   Precision   Recall   F-Measure     ROC Area     Class
    1                0          1          1          1           0.8         mammal
    1               0.1       0.667        1         0.8        0.792         reptile
    1                0          1          1          1            1          amphibian
  0.667              0          1        0.667       0.8        0.833         fish
    1                0          1          1          1            1             bird
Weighted Avg.   0.917      0.017      0.944      0.917          0.917      0.874
```

The Summary and Detailed accuracy can be easily interpreted as in interpreted in weather data set.

VI. CONCLUSIONS

This paper presents an example of a decision tree building process, one of the most common data mining techniques. We have tried to highlight the way the stored data about past events can be used in future. For that we can use a decision tree built with ID3 algorithm implemented in specialized software in data mining - Weka. The studies and their implementation conducted here conclude that the decision tree learning algorithm ID3 works well on any classification problems having dataset with the discrete values [8].

VII. FUTURE SCOPE

In future this paper will provide a detailed study and understanding of the use of free data mining software available under the GNU General Public Licenseï **Weka.** Moreover It has explained that the decision tree generated in Weka Can be easily interpreted by forming IF-THEN rules which can be further used to form a simple decision tree. In future people who are studying Data Mining can gain knowledge of different kinds of data and their properties by examine them in Weka.

REFERENCES:

[1] E. Thomas, ñ*Data mining: definitions and decision tree examples*,ò Stony Brook, State University of Newyork.*International Journal of Information and Electronics Engineering*, Vol. 2, No. 2, March 2012

[2] Data Mining Algorithms *(Analysis Services - Data Mining)*. [Online]. Available: http://technet.microsoft.com/en us/library/ms175595.aspx

[3] RupaliBhardwaj and Sonia Vatta*"Implementation of ID3 algorithm"*,.*International Journal of Advanced Research in Computer Science and Software Engineering*Vol 3, June 2013.

kumar and Rajini Jindal *"The Base Strategy for ID3* *ournal of Information and Electronics Engineering.*Vol 2,

[5] Wei Peng, Juhua Chen and Haiping Zhou, of ID3,*' An Implementation Decision Tree Learning Algorithm'*, University of New South Wales, School of Computer Science &Engineering,Sydney, NSW 2032, Australia .

[6] Tutorial CSE 5230: *"The ID3 Decision Tree Algorithm*ò, *MONASH UNIVERSITY*. 2014.

[7] Tom M. Mitchell, Book on *Machine Learning,* Singapore, McGrawHill, Published in 1997, ISBN 10: **0070428077 / 0-07-042807-7** ,ISBN 13: **9780070428072**.

[8] J. Han and M. Kamber, ñ*Data Mining: Concepts and Techniques (2nd edition)*,ò Morgan Kaufmann Publishers, 2006

[9] Vipin Kumar, Pang-Ning Tan and Michael Steinbach, ñ*Introduction to Data Mining*ò Pearson, Published on 05/02/2005, ISBN-10: 0321321367 , ISBN-13: 9780321321367.

[10] http://www.wikipedia.com/

[11] http://www.google.com/