

# COMPARATIVE STUDIES OF ARTIFICIAL INTELLIGENCE (AI) TECHNIQUES

*Lubabatu Sada Sodangi and Suleiman Isah Sani*

Department of Computer Studies  
College of Science and Technology  
Hassan Usman Katsina Polytechnic – Katsina Nigeria  
lubabatusodangi@yahoo.com, ssmiadua@gmail.com,

**Abstract:** Determining the accuracy and duration of AI techniques is important for the reason that the Application of such techniques in Data mining has become wider. There is a range of techniques used in AI for finding hidden or unknown information in data sets (or data groups), and most of these techniques have their own sub fields. Artificial intelligence techniques are extensive and too numerous to list. In this paper, a machine learning algorithm that combines two widely used algorithms Decision trees [J48] and Neural networks [Multilayer perceptron] are compared and studied using Waikato Environment for Knowledge Analysis(Weka) and Iris dataset from UCI machine learning repository. These two algorithms determine and compare the classifier errors and time taken to build the model in the explorer window and compare the Error rate of each algorithm in experimenter window. The result shows that Decision Trees [J48] runs faster but produce high error rate and low accuracy whereas Neural Network [Multilayer perceptron] is more preferable in Data mining although it takes longer training time, yet it yields low error rate and high accuracy.

Keywords: Data mining, Neural networks, Decision trees, J48, Multilayer perceptron, Weka.

## 1. INTRODUCTION

Artificial Intelligence (also called AI) is one of the most recent fields in science and computing; it encompasses a large number of subfields [1]. AI is a branch of science that creates machines or computer systems which help humans find solutions to complex problems. It could also be defined as the development of computer systems that have the ability to perform various tasks that normally require human intelligence [2]. Hence, we could say that AI deals with the study

and development of computer systems that exhibit some form of intelligence and attempts to apply this knowledge to the design of computer-based systems that can understand a natural language [3].

Over the past five decades, the field of AI seems to have developed significantly. It has succeeded in developing into an industry of its own and has successfully contributed to other science and non-science fields like medicine, accounting and auditing, philosophy, economics, and psychology and so on [4]. There is a range of techniques used in AI for finding hidden or unknown information in data sets (or data groups), and most of these techniques have their own sub-fields. AI techniques are extensive and too numerous to list. Some examples of these AI techniques are Decision trees (e.g. J48 tree, ID3 tree, ID5 tree), Neural networks (e.g. multilayer perceptron), Fuzzy logic; Expert systems and Evolutionary computation to mention a few. Individually, each technique is a problem solving agent but each of the techniques differs from each other because they all have different levels of performance.

This paper analyses the result of Iris dataset in the explorer window by means of J48 and Multilayer perceptron classifiers using 10 and 90 fold cross validation. It also compares the performance of each technique in Experimenter window using 10 and 90 fold cross validation.

The aim of this paper is to compare the performance of Decision trees (J48 tree) and Neural networks (multilayer perceptron) using Waikato Environment for Knowledge Analysis (Weka) simulation tool. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant (iris setose, iris versicolour and iris virginica). Each instance is described by 4 attributes – sepal length, sepal

width, petal length and petal width. All attributes in this dataset are 'real' values.

In this Study, Iris dataset obtained from [UCI Machine Learning repository] was used in WEKA Machine learning to compare the performances of two AI techniques (J48 & MLP). Waikato Environment for Knowledge Analysis (WEKA) was developed in the University of Waikato, New Zealand. It is a collection of machine learning algorithms written in java and comprises an interface to many standard machine learning techniques [5].

Weka is useful in data mining because:

1. It has a broad set of data pre-processing tools, learning algorithms and evaluation methods.
2. It can be used to relate a learning method to a dataset and analyse its output to extract essential information about the data [5].

Weka is chosen in this study among other simulation tools because it provides an environment called [Experimenter] for comparing learning algorithms.

## 2. METHODS

There are various classification algorithms that vary in terms of performance for supervised learning. Some of these classification algorithms perform better than the others on a particular dataset. In this paper, two Artificial Intelligence techniques J48 and Multilayer perceptron are chosen, and their performances are compared to each other in terms of classifier errors, time taken to build their models and error rate using Iris Dataset.

### 2.1. J48

J48 is an application developed from C4.5 Decision tree learners. Professor Ross Quilan of the University of Sydney implemented C4.5 in 1993[6]. It is an inductive Machine learning algorithm based on classification problems.

J48 is a strong classifier that essentially performs the following on a given Dataset [7]:

- a. Deals with continuous attributes
- b. Handle nominal values
- c. Convert trees to rules and
- d. Can handle any mixture of nominal and numeric properties.

J48 was chosen in this comparison because it can be used for classification and it provides less

accuracy rate for correctly classified instances [8].

### 2.2. Multilayer Perceptron (MLP)

Multilayer perceptron is a type of neural network algorithm that can solve complex problems that are complicated for humans to easily identify or detect. It has the ability to learn how to perform actions based on a given Dataset. It can also assign output nodes to represent each class in classification and it will have more than one output nodes [9]. It is also a well-known classification technique and is known as a practical tool for classification in Data Mining. Multilayer perceptron can prepare data into required output because it is a supervised network. It has been also shown to approximate any input-output map into one or more hidden layers [7].

### 2.3 Iris Dataset

[10] developed Iris dataset as an example of discriminant analysis. It is a multivariate dataset and consist of 50 samples of which comes the three species of Iris (setosa, virginica, and versicolor). [10] introduced a linear discriminant model to distinguish the species from each other. This was by measuring the four features from each sample the length(sepal)and the width(petals) in centimeters [10]. Iris dataset is a classification data that consist of 150 instances, four attributes and a class that contains three classes of 50 instances each. Each class refers to a type of Iris plant of setosa, versicolour and virginica. The dataset has no missing values and has nominal attributes. The problem is to identify whether an iris flower is versicolor, setosa or virginica by examining its sepal length and width and petal width and length [11].

## 3. SIMULATION

Encoding is a method of converting data from one form to another, e.g. from Non numeric to numeric. In Data mining, Non-numeric inputs are converted to numeric for a machine to

identify each instance. This is done by filters in WEKA. In this study the classification problem of the dataset needs no encoding because it has nominal (descrete) attributes, which goes well with classification learning algorithm.

Therefore, the data was first copied to notepad, where headings were inserted after which it was imported to Microsoft excel (spread sheet) for a clear and complete view and saved as Comma Separated Value (CSV). The data was loaded into the WEKA explorer and saved as Attribute-relation (ARFF) file format file. The ARFF described the Iris dataset as follows:

```
@relation iris
@attribute 'sepal length' numeric
@attribute 'sepal width' numeric
@attribute 'petal length' numeric
@attribute 'petal width' numeric
@attribute class {Iris-setosa, Iris-versicolor, Iris-virginica}
@data 5.1, 3.5, 1.4, 0.2, Iris-setosa
```

### 3.2. Implementation

For the whole of the experiment, the Iris dataset was split randomly where 66% was used as the training set and 34% as test set. A 10 fold cross validation and 90 fold cross validation was used in the explorer window, to compare which number of folds is better in terms of the time taken to build the model, also to evaluate classifier errors. Likewise in the experimenter window, same number (10 and 90) cross validation was used to compare the two techniques in terms of which had a lower error rate.

**THE EXPLORER WINDOW:** The ‘explorer’ was selected from the four graphical user interface options in WEKA panel. The Iris dataset was loaded in the explorer screen after which, a screen appeared that explained the dataset, confirming the dataset had 5 attributes, 150 instances and had no missing values. The class attribute was described by a histogram at the lower right corner of the window.

**Experimenter:** Experimenter was selected from the four GUI options at WEKA panel. At the extreme top of the experimenter screen are three panels (*setup*, *run*, *analyse*) where the experiment was started at the first panel *setup*, *run* and then *analysed*.

This experiment uses a simple interface. A new experiment was then started by clicking the

“new” tab to define default parameters. At *Result destination*, CSV file was chosen to save the result in the ‘work’ folder. The Iris dataset was then loaded by clicking the *add new* button.

## 4. RESULTS

After loading the Iris dataset into the explorer window and experimenter window, the result history list was viewed by right clicking on the classifier at the lower left. A new line was added whenever the classifier was run. In this paper, two lines were added at the lower left, the first line for a 10 fold cross validation and the second for 90 fold cross validation respectively.

### 4.1 The Explorer Window

At the beginning of the explorer window, it showed the summary of the dataset, and the number of fold cross validation used. Using 10 fold cross validation it took the Multilayer perceptron classifier 0.23 seconds to build the model where it showed 146 instances (97.3333%) as the correctly classified instances and 4 instances (2.6667%) as the incorrectly classified instances. Using same Multilayer Perceptron classifier but with 90 fold cross validation it took 0.21 seconds to build the model, and the result showed 147 instances (98%) as correctly classified instances and 3 instances (4%) as the incorrectly classified instances.

The summary of the dataset and number of fold cross validation used was shown at the beginning of the explorer window. Using 10 fold cross validation and 90 fold cross validation it took J48 classifier 0.04 seconds and 0.01 seconds to build the model respectively. However, with 10 fold cross validation, it showed that 96% were correctly classified instances (i.e. 144 instances) and 4% were incorrectly classified instances (i.e. 6 instances). Using same J48 classifier but with a 90 fold cross validation, the result showed that 98% was correctly classified (i.e. 147 instances) and 2% as the incorrectly classified instances (i.e. 3 instances).

The simulation result are shown in table 1. The results are classified into a number of sub items in the table below for clear analysis and evaluation. The first column lists the names of the classifiers followed by second column where correctly and incorrectly classified instances will be partitioned in numeric and percentage value.

Kappa statistics is” used to measure the accuracy of any particular measuring case and its usual to differentiate between the reliability of the data collected and their validity” [12]

Table 2 summarise the difference of errors resulting from the training of the two selected algorithms. The widely used errors indicators are used in this study that is absolute errors, root mean square and relative errors. It is observed that the highest error belongs to J48 (90) with an average value of the two error readings 0.10485

**Table 1: Simulation result of each algorithm**

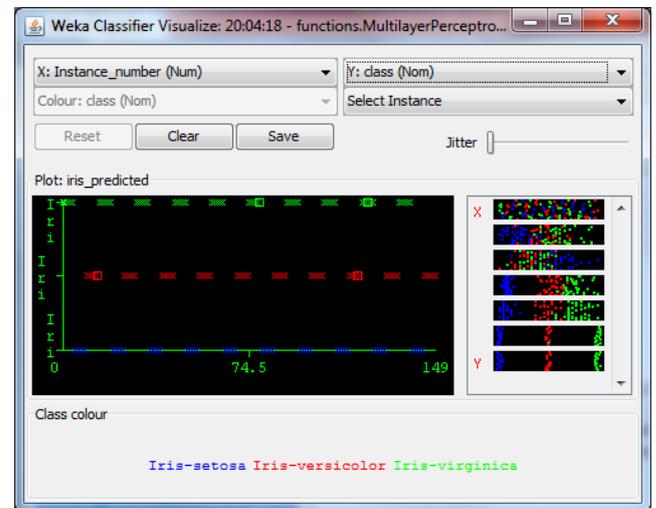
Algor ithm	Correctly classified instances %(value)	Incorrectl y classified instances %(value)	Time taken(se conds)	Kap pa stati stics
J48(1 0)	96 (144)	4 (6)	0.04 seconds	0.94
J48(9 0)	95.3333 (143)	4.6667 (7)	0.01 seconds	0.93
MLP (10)	97.3333(1 46)	2.6667(4)	0.23sec	0.96
MLP (90)	98(147)	2(3)	0.21sec	0.94

**Table 2: Training and simulation errors**

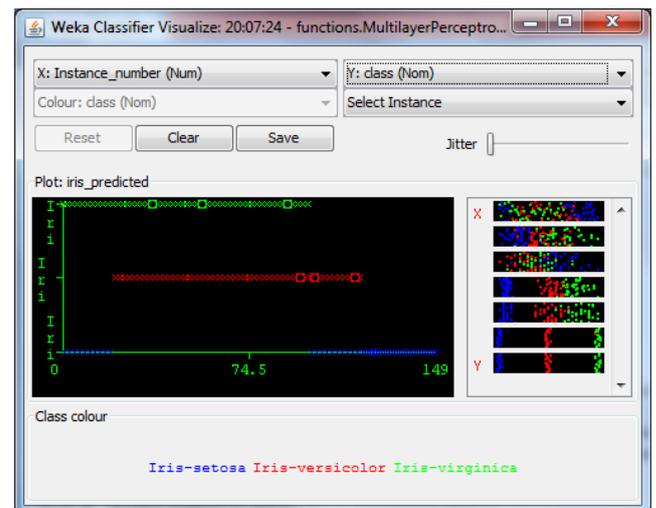
Algorith m	Mean Absolut e Error	Root Mean Squar e Error	Relativ e Absolut e Error(%) )	Root Relativ e Square Error(%) )
J48(10)	0.035	0.158 6	7.8705	33.6353
J48(90)	0.0387	0.171	8.6786	36.1271
MLP(10)	0.0327	0.129 1	7.3555	7.3555
MLP(90)	0.0382	0.144	8.559	30.4328

Figures 1-4 below displays the result of each Algorithm classifier errors. This showed a window menu that displays options- *visualise classifiers error* was selected from the options. A window is displayed showing the data points in blue, red and green for Iris setosa, Iris versicolor and Iris virginica respectively. The classifier errors were shown clearly when X-axis is chosen

as instance-number (num) and Y-axis as predicted class (nom) where the correctly classified errors are displayed in crosses and the incorrectly classified are shown as boxes.



**Figure 1: 10 folds cross validation MLP classifier errors**



**Figure 2: 90 folds cross validation MLP classifier errors**

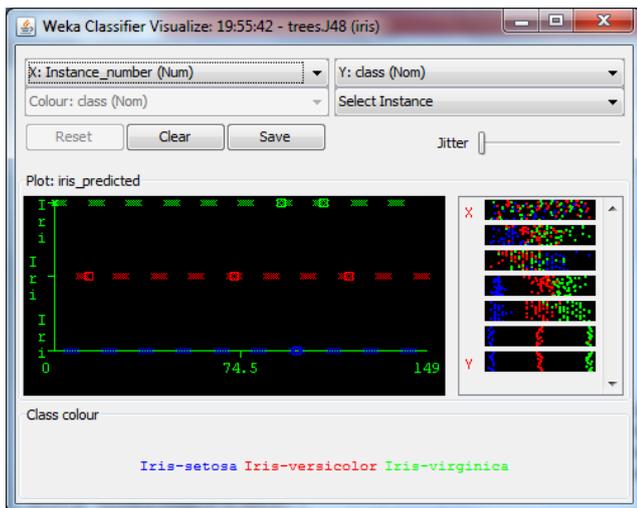


Figure 3: 10 fold cross validation J48 classifier errors

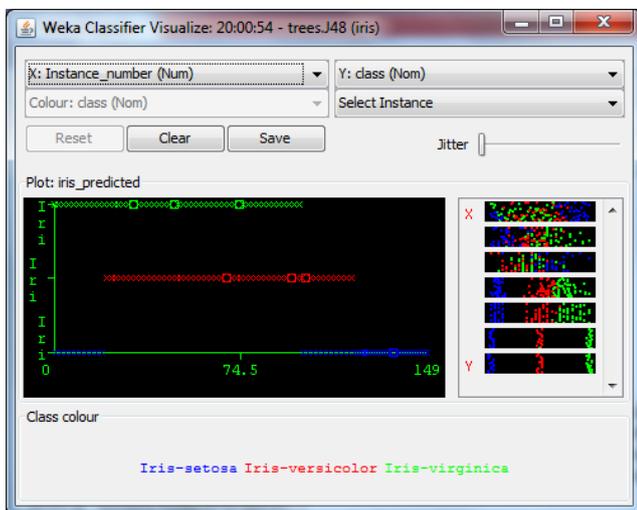


Figure 4: 90 fold cross validation J48 classifier errors

## 4.2 Experimenter

The result of the performance of J48 and Multilayer perceptron were shown in the large panel on the right of *analyse* window. Initially, the two algorithms compared per cent correct statistics using 10 fold cross validation. From the little table, the two algorithms were shown horizontally and were numbered 1 & 2. The second row displayed (100) and 900 meaning that the experiment ran 10 times 10 fold cross validation and 10 times 90 fold cross validation respectively.

The per cent correct for J48 was 94.73% using 10 cross validation (see fig.1) and 95.111% using 90 cross validation (see fig.2). From the third row, the symbols (v/ /\*) specified that the result was either better (v) or worse (\*) than the baseline scheme. Being v at first indicated that

the first algorithm (1) trees.J48 at 0.05 significance level performed well.

Using 10 fold cross validation, MLP produced 96.93% as correct and was equivalent to baseline (J48), likewise using 90 fold cross validation MLP produced 97.00% as correct. From the third row, second column, the 1 at the middle (0/1/0) indicates that the second algorithm (2) funct [MLP] was equivalent to first algorithm baseline (J48) once.

The result of the experimenter window is summarised in the table below:

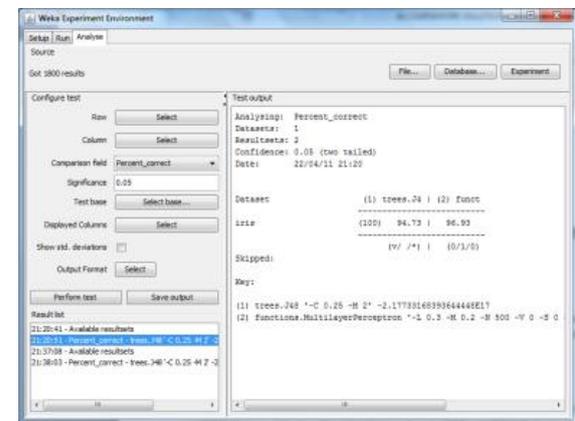


Figure 5: 10 fold cross validation

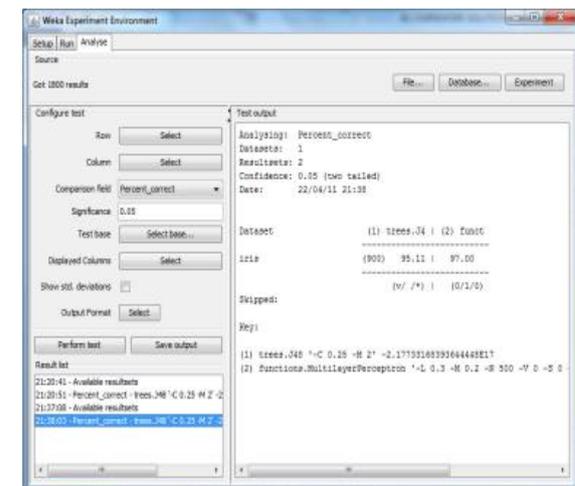


Figure 6: 90 fold cross validation

## 5. CRITICAL ANALYSIS OF THE RESULTS

When the Iris dataset was loaded into the explorer window and run using 10 fold cross validation, trees.J48 classifier took 0.04 seconds to build the model and showed 144 instances [96%] as correctly classified instances and 6 instances [4%] as incorrectly classified instance whereas functions. MLP took 0.23 seconds to build the model and showed 146 instance [97.3333%] as correctly classified instance and 4 instances [2.6666%] as incorrectly classified instance.

Similarly, using 90 fold cross validation for the both classifiers, J48 took 0.01 seconds to build the model and showed 147 instances [98%] as correctly classified instances and 3 instances [2%] as incorrectly classified instances. Whereas, MLP took 0.21 seconds to build the model and showed 148 instances [98.6667%] as correctly classified instances and 2 instances [1.3333%] as incorrectly classified.

At the experimenter window, with 10 fold cross validation, trees.J48 had 94.73% correct and function MLP had 96.93% correct and was equivalent to J48. Similarly, using 90 fold cross validation to compare per cent correct, J48 produced 95.11% correct, whereas MLP produced 97.00% correct and was equivalent to the baseline scheme (J48).

From the table 2 above, It is observed that the two algorithms differs by 0.5 – 1% on their performances, this observation is similar to [13] statement in their paper they stated that “other popular classifiers like decision trees, k-nearest neighbour and multilayer perceptron also have similar performance with 0.5%-1% less accurate”.

On accuracy metrics, Multilayer perceptron performed better with an accuracy of 97.0% and 96.93% using 90 and 10 folds respectively. This was stated by [8] in their paper that the “highest accuracy belongs to neural networks (neural network classifier) and subsequently decision tree with pruning”. Moreover [8] elaborates that “the total time/duration of building a model is an essential constraint in comparing the classification algorithm” in this experiment, it is observed that Decision trees algorithms requires the shortest time to build the model using 90

folds cross validation whereas MLP algorithm that has lower error rate.

Furthermore, our result shows that for the two algorithms on the iris dataset the most of the well-known classification algorithms can work better than the others but Multilayer perceptron outperforms all of them with low error rate and high accuracy. [8] Also stated that an algorithm with low error rate is more favourable in health and bioinformatics fields.

## 6. CONCLUSION

This Study showed that, whereas the decision trees algorithm runs faster, it produces a high error rate and low accuracy. It can be concluded that J48 is more preferable using 90 fold cross validation to produce low error rate and high accuracy. On the other hand, although MLP may have low error rate, it takes longer training time to produce higher accuracy, it can be concluded that MLP is more preferable in Data mining because of the high accuracy, and MLP is more accurate using 10 fold cross validation.

In this study, the overall observation is that the best algorithm based on accuracy and duration to build a model is J48 using 90 fold cross validation. This result recommends that Multilayer perceptron is more preferred in terms of accuracy whereas J48 is considered as the fastest in building the model

## REFERENCE

1. <http://www.cs.berkeley.edu/~russell/intro.html> [Accesed on 05/08/2012]  
[http://www.thinkquest.org/pls/html/f?p=52300:30:807824381164306:::P30\\_CATEGORY\\_ID:CP\\_ARTIFICIAL\\_INTELLIGENCE](http://www.thinkquest.org/pls/html/f?p=52300:30:807824381164306:::P30_CATEGORY_ID:CP_ARTIFICIAL_INTELLIGENCE) [Accesed on 07/09/2012]
2. <http://www.brainmetrix.com/intelligence-definition/>[Accesed on 05/10/2012]
3. Miklos A. V and Alexander K, (1997): Artificial Intelligence in Accounting and Auditing: Towards New Paradigms : Volume 4 Newark, Markus Wiener Publisher, Princeton, USA
4. Scuce, D., (2010). Experimenter. In: A. Seewald, ed. WEKA manual: for version 3-5-5. New Zealand: University of Waikato.ch.5

5. Quinlan, J. R. (1993) C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA
6. Norvig, R. and Russell, S. (2010) :Artificial intelligence: A modern approach.3<sup>rd</sup> edition. New Jersey: Pearson
7. Othman, M. and Yau, T :“Comparison of different classification techniques using WEKA for breast cancer ”. International Conference on Biomedical Engineering; 2007. pp. 520–523.
8. Lawrence, J. (1994) :Introduction to Neural Networks: Design, Theory and Applications. California Scientific Software,
9. R. A. Fisher (1936) :“The use of multiple measurements in taxonomic problems”. Annals of Eugenics 7 (2): 179–188. doi:10.1111/j.1469-1809.1936.tb02137.x [Accessed on 05/09/2012]
10. [www.ics.uci.edu/~mlearn/](http://www.ics.uci.edu/~mlearn/) [Accessed 5 september 2011].
11. Kappaat<http://www.dmi.columbia.edu/homepages/chuangj/kappa>
12. Hasan , M, .et al . (2006): Link Prediction using Supervised Learning (e-journal) available through Google scholar (accessed 9th September, 2012).
13. Witten, I. H. and Frank, E. (2005) : Data mining: Practical machine learning tools and techniques. 2<sup>nd</sup> Edition. Morgan Kauffman: San Francisco