ISBN 978-93-84422-22-6
Proceedings of 2015 International Conference on Image Processing, Production and Computer Science (ICIPCS'2015)
Istanbul (Turkey), June 3-4, 2015 pp. 120-126

Comparative Analysis on the Performance of Artificial Intelligence (AI) Classification Algorithms

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

Abstract: *There is range of AI Algorithms used in data mining to determine the hidden or unknown information in a datasets. AI techniques are wide and too many to mention and most of these techniques have their own subfields. This paper determines the accuracy and duration of two AI classification algorithms: Bayesian network (naive Bayes) and neural network (multilayer perceptron).This is important because the application of such techniques in Data mining has been studied extensively since 1950s (1). This paper analyses the result of iris dataset from UCI machine learning repository using Waikato Environment for Knowledge Analysis (Weka). The two algorithms were used in explorer window to determine and compare the classifier errors and time taken to build the model and also to compare the Error rate of each algorithm in experimenter window. It is observed from the results that naïve Bayes runs faster but produce high error rate and low accuracy while Neural Network [Multilayer perceptron] is more desirable in Data mining although it takes longer training time, however it gives low error rate and high accuracy.*

Keywords: *Data mining, Neural networks, Multilayer perceptron, naive Bayes, Weka.*

1. Introduction

Artificial Intelligence (AI) as one of the wider fields in science and computing; it comprises a great number of subfields [1] and can be defined as the development of computer systems that are capable of performing various tasks that typically entails human intelligence [2]. AI is a branch of science that develops machines or computer systems that assist humans to find solutions to complex problems. Therefore, AI could also deals with the study

And advancement of computer systems that revealed some form of intelligence and tends to apply this knowledge to the design of computer-based systems that can recognize a natural language [3].

For the past fifty years, there is rapid growth in the field of AI. It has succeeded in developing into a field of its own and has positively contributed to other branches of science and non-science fields like medicine, accounting and auditing, philosophy, economics, and psychology and so on [4]. There are various algorithms used in AI for determining hidden or unknown information in data sets (data groups), and most of such techniques have their own sub-fields. AI techniques are broad and too numerous to mention. Some examples of these AI techniques are Decision trees (e.g. J48 tree, ID3 tree, ID5 tree), Bayesian network (naïve bayes), Fuzzy logic; Neural networks (e.g. multilayer perceptron), Expert systems and Evolutionary computation etc. Solely, every technique is a problem solving agent however each of the techniques differs from each other since they all have unique levels of performance.

In this study, Iris dataset from UCI machine learning repository and Waikato environment for knowledge analysis (WEKA) which is a machine learning algorithm studied the two commonly used algorithms Bayesian network and neural networks. These two algorithms were used to find and compare the classifier errors and time taken to build the model in the explorer window using 10 and 90 fold cross validation and to compare the error rate of both algorithms in experimenter window.

The data set has 3 classes of 50 instances each, in which each class represent a type of iris plant (iris setose, iris versicolour and iris virginica). Every instance is described by 4 attributes – sepal length, sepal width, petal length and petal width. All the attributes in the dataset are ‘real’ values.

Iris dataset obtained from [UCI Machine Learning repository] was used in this research to compare the performances of two AI techniques (naïve bayes & MLP). University of Waikato, New Zealand developed Waikato Environment for Knowledge Analysis (WEKA) which is a collection of machine learning algorithms written in java and contains an interface to many standard machine learning techniques [5]. Weka is useful in data mining due to following reasons:

1. It has a wide-range of data pre-processing tools, machine learning algorithms and evaluation methods.

2. It can be used to simulate a learning method to a dataset and analyze its output to obtain vital facts about the data [5].

Weka simulation tool is chosen in this research because its [Experimenter] environment can be used for comparing learning algorithms.

2. Methods

Supervised learning comprises of various classification algorithms that differ in terms of performance. The performance of these classification algorithms varies depending on the type of dataset. In this paper, the performance of two AI techniques Bayesian network [naïve bayes] and neural networks [Multilayer perceptron] are compared to each other in terms of classifier errors, duration of building models and error rate using Iris Dataset.

2.1. Naïve Bayes

Naive Bayes classifier belongs to a group of simple probabilistic classifiers in machine learning . It offers a good performance on text categorization problems [6]. Naïve bayes models are known in the statistics and computer science works using various names like[7]: Bayesian classifier, simple Bayes and independence Bayes.

Bayesian network is a graphical model that essentially performs the following for data analysis[8]:

a. The model known as classification technique can converts dependencies amongst all variables [8].
b. It can readily handles conditions for missing data entries [8].
c. A Bayesian network are known in the statistics works to learn causal relationships, to understand the problem domain and hence to predict the consequences of intervention [8].
d. As a result of the causal and probabilistic semantics of the model, it is an ideal representation for relating prior knowledge and data [8].
e. A Bayesian network in conjunction with Bayesian statistical methods gives an effective methodology for avoiding data over fitting [8].

2.2. Multilayer Perceptron (MLP)

Multilayer perceptron is a well-known classification technique and is known as a practical tool for classification in Data Mining. It’s a neural network algorithm that can work/solve complicated problems that are difficult for humans to easily detect. It is capable of learning how to perform tasks on a given Dataset as well as allocating output nodes to represent each class in classification and it will have more than one output nodes [9]. Multilayer perceptron is a supervised network; therefore it can prepare data into required output.

2.3 Iris Dataset

Iris dataset was developed by [10] as an example of discriminant analysis. 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 denotes to a type of Iris plant of setosa, versicolour and virginica. The dataset has nominal attributes and contains no missing values. The difficulty is to find out if an iris flower is versicolor, setosa or virginica by observing its sepal length and width and petal width and length [11]. It is a multivariate dataset and consist of 50 samples of which comes the three species of Iris (setosa, virginica, and versicolor). A linear discriminant model was introduced to distinguish the species from each other [10]. This was by examining the four features from every sample of the width (petals) and the length (sepal) in centimeters [10].

3. Simulation

Data preparation: In Data mining, non-numeric inputs are converted to numeric through a process called-Encoding which is a method of changing data from one form to another. Filters in WEKA converts Non-numeric inputs to numeric for a machine to recognize each instance. The classification problem of the dataset needs no encoding in this study; this is because it has nominal (discrete) attributes, which goes well with classification learning algorithm.

For that reason, the data was first copied to notepad, then headings were put in, afterwards the data was fetched into the 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 kept as Attribute-relation (ARFF) file format file. The ARFF defined the Iris dataset as:

@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

3.1. Implementation

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

The Explorer Window: From the four graphical user interface options in WEKA panel, the ‘explorer’ was selected. When the Iris dataset was loaded in the explorer screen, a screen appears that described the dataset, explaining the dataset had 5 attributes, 150 instances and had no missing values. The class attribute was shown by a histogram at the lower right corner of the window.

Experimenter: Experimenter was carefully chosen from the four GUI options at WEKA panel. At the extreme top of the experimenter screen are three panels (setup, run, analyze) as the experiment was started at the panel setup, run and then analyzed.

The experiment uses a simple interface. A new experiment was run by clicking the “new” tab to outline default parameters. The Iris dataset was then loaded by clicking the add new button. At Result destination, CSV file was chosen to save the result in the ‘work’ folder.

4. Results

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

4.1 The Explorer Window

After loading the dataset at Explorer window, it showed the summary of the dataset, and the number of fold cross validation used. Using 10 fold cross validation, Multilayer perceptron classifier spends 0.23 seconds to build the model where it displayed 146 instances (97.3333%) as the correctly classified instances and 4 instances (2.6667%) as the incorrectly classified instances. Using 90 fold cross validation, same Multilayer Perceptron classifier spends 0.21 seconds to build the model, and the result displayed 147 instances (98%) as correctly classified instances and 3 instances (4%) as the incorrectly classified instances.

The explorer window also describe the summary of the dataset and number of fold cross validation used at the beginning of the window. Using 10 fold cross validation and 90 fold cross validation naive Bayes classifier spends 0.04 seconds and 0.01 seconds to build the model respectively. On the other hand, with 10 fold cross validation, it displayed that 96% were correctly classified instances (i.e. 144 instances) and 4% were incorrectly classified instances (i.e. 6 instances). Using same naive Bayes classifier but with a 90 fold cross validation, the result displayed that 98% was correctly classified (147 instances) and 2% as the incorrectly classified instances (3 instances).

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

Kappa statistics is used to determine the accurateness of any specific measuring case and its usual to distinguish amongst the reliability of the data collected and their validity" [12]

Table 2 below describes the change of errors obtained from the training of the two selected algorithms. This study uses the commonly errors indicators: absolute errors, root mean square and relative errors. The result shows that the highest error belongs to naïve bayes (90) with an average value of the two error readings 0.10485

TABLE I: Simulation result of each algorithm

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

TABLE II: Training and simulation errors

Algorithm	Mean Absolut e Error	Root Mea Squa e Error	Relativ e Absolut e Error(%)	Root Relativ e Square Error(%)
Naïve bayes(10)	0.035	0.158 6	7.8705	33.6353
naive bayes(90)	0.0387	0.171	8.6786	36.1271
MLP(10)	0.0327	0.129	7.3555	7.3555
MLP(90)	0.0382	0.144	8.559	30.4328

Figures 1-5 below shows the result of every Algorithm classifier errors. This displays a window menu that illustrates options- visualize Classifiers error was chosen from the options. A window is presented that shows the data points in blue, red and green for Iris setosa, Iris versicolor and Iris virginica respectively. The classifier errors were displayed clearly when X-axis is selected as instance-number (num) and Y-axis as predicted class (nom) where the correctly classified errors are shown in crosses and the incorrectly classified are displayed as boxes.

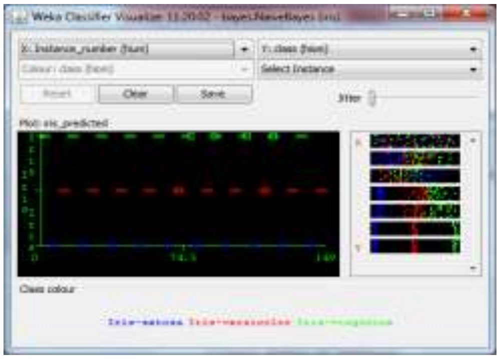

Fig. 1: 10 fold cross validation NB classifiers

Fig. 2: 90 folds cross validation NB classifiers

Fig. 3: 10 folds cross validation MLP classifiers

Fig. 4: 90 folds cross validation MLP classifiers

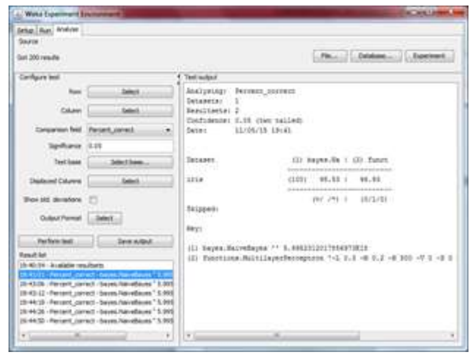

Fig. 5: 10 & 90 folds cross validation Experimenter table

4.2 Experimenter

The performance of the two algorithms: naïve Bayes and Multilayer perceptron were displayed in the large panel of analyze window. In the beginning, the two algorithms compared per cent correct statistics using 10 and 90 fold cross validation. From the tiny table, the two algorithms were displayed horizontally and were numbered 1 & 2. The second row displayed (100) which means the experiment ran 10 times 10 fold cross validation.

The per cent correct for naïve bayes was 95.53% using 10 and 90 cross validation (see fig.1) and 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 shows that, the first algorithm (1) naïve Bayes at 0.05 significance level did well.

Similarly, Using 10 fold cross validation, MLP produced 96.93% as correct and was equivalent to baseline (naïve bayes), likewise using 90 fold cross validation it produced same figure (96.93%) 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 (naïve bayes) once.

The result of the experimenter window is explained in the table above.

5. Critical Analysis of the Results

After loading the Iris dataset into the explorer window and it was run using 10 fold cross validation, naïve bayes 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.

Likewise, using 90 fold cross validation for the both classifiers, naïve bayes spend 0.01 seconds to build the model and displayed 147 instances [98%] as correctly classified instances and 3 instances [2%] as incorrectly classified instances. While, MLP spend 0.21 seconds to build the model and displayed 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, naïve bayes had 94.73% correct and function MLP had 96.93% correct. Likewise, using 90 fold cross validation to compare per cent correct, naïve bayes produced 95.11% correct, whereas MLP produced 97.00% correct and was equivalent to the baseline scheme (naïve bayes).

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

On accuracy performance, Multilayer perceptron did 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". Furthermore [8] explains 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 Naïve bayes algorithms requires the shortest duration to build the model using 90 folds cross validation while MLP algorithm has lower error rate.

Moreover, our result demonstrates that for the two algorithms on the iris dataset the best of the well-known classification algorithms can work better than the others but Multilayer perceptron performs better with low error rate and high accuracy. [8] Also stated that an algorithm with low error rate is more preferable in health and bioinformatics fields.

6. Conclusion

In this study, the general observation is that the best algorithm based on accuracy and duration to build a model is naïve bayes using 90 fold cross validation. It is also noticed that, although the naïve bayes algorithm runs faster, it produces a high error rate and low accuracy. It can be concluded that naïve bayes is more suitable using 90 fold cross validation to provide low error rate and high accuracy. On the other hand, although MLP may have low error rate, it takes longer training time to provide higher accuracy, it can be concluded that MLP is more favorable/preferable in Data mining due to the high accuracy, and MLP is more accurate using 10 fold cross validation.

This result suggests that Multilayer perceptron is more favored in terms of accuracy although naïve bayes is considered as the fastest in building the model.

7. Reference

[1] http://www.cs.berkeley.edu/~russell/intro.html [Accesed o 05/08/2012]

[2] http://www.brainmetrix.com/intelligence-definition/[Accesed on 05/10/2012]

[3] A.V Miklos and K. Alexander , (1997): Artificial Intelligence in Accounting and Auditing: Towards New Paradigms : Volume 4 Newark, Markus Wiener Publisher, Princeton, USA.

[4] D. Scuce, (2010). Experimenter. In: A. Seewald, ed. WEKA manual: for version 3-5-5.New Zealand: University of Waikato.ch.5.

[5] J. R. Quinlan, (1993) C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.

[6] R. Norvig, and S. Russell, (2010) :Artificial intelligence: A modern approach.3rd edition. New Jersey: Pearson.

[7] D. J. Hand, and K. Yu, (2001). "Idiot's Bayes — not so stupid after all?". *International Statistical Review* **69** (3): 385–399. doi:10.2307/1403452. ISSN 0306-7734Lawrence, J. (1994) :Introduction to Neural Networks: Design, Theory and Applications. Califonia Scientific Software.

http://dx.doi.org/10.2307/1403452

[8] D. Heckerman, (1995). A tutorial on learning Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research.

[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 [Accesed on 05/09/2012].

http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x

[10] www.ics.uci.edu/~mlearn/ [Accessed 5 september 2011].

[11] Kappaathttp:/www.dmi.columbia.edu/homepages/ chuangj/kappa.

[12] Hasan , M, . (2006): Link Prediction using Supervised Learning (e-journal) available through Google scholar (accessed 9th September, 2012).

[13] I. H Witten, and E. Frank, (2005): Data mining: Practical machine learning tools and techniques. 2nd Edition. Morgan Kauffman: San Francisco.