The Combination and Comparison of Neural Networks with Decision Trees for Wine Classification

Rohitash Chandra[1], Kaylash Chaudhary[2], Akshay Kumar[3]

[1, 2, 3] *School of Science and Technology, The University of Fiji, Lautoka, Fiji.*

rohitashc@unifiji.ac.fj, kaylashc@unifiji.ac.fj,

akshay@unifiji.ac.fj

Abstract. *This work presents the comparison and combination of neural networks with decision trees on the application of wine classification. Neural networks are first trained and then combined with decision trees in order to extract knowledge learnt in the training process. Artificial neural networks are used for the classification of Italian wines obtained from a region which has three different wine cultivars. Wines are classified according to their respective cultivar using the chemical analysis of the thirteen major chemical constituents. The trained network classifies a sample of wine according to the knowledge the network acquired by learning from previous wine samples. After successful training, knowledge is extracted from these trained networks using decision trees in the form of 'if-then' rules. We then use decision trees to train on the same dataset and compare the performance of neural networks, and decision trees in both knowledge extraction from neural networks and classification of wines on their own. Our results show that artificial neural networks perform better when compared to decision trees however, the extraction of knowledge from neural networks do not outperform the performance of decision trees alone. The general paradigm can be applied to other categories of food classification and processing.*

Keywords

Wine classification, artificial neural networks, decision trees, and knowledge extraction.

1 Introduction

Artificial intelligence has been successfully applied in food analysis. They have been used for food processing and classification as demonstrated in [1, 2, and 3]. Artificial neural networks are artificial intelligence paradigms; they are machine learning tools which are loosely modelled after biological neural systems. They learn by training from past experience data and make generalization on unseen data. They have been applied as tools for modelling and problem solving in real world applications such as speech recognition, gesture recognition, financial prediction, and medical diagnostics [4, 5, 6 and 7]. Backpropagation employs gradient descent learning and is the most popular algorithm used for training neural networks. Neural networks were recently viewed as 'black boxes' as they could not explain how they arrived to a particular solution. Knowledge extraction is the process of extracting valuable information from trained neural networks in the form of 'if-then' rules as shown in [8, 9]. The extracted rules describes the knowledge acquired by neural networks while learning from examples.

Wine has social and commercial importance. They are used in number of social, business and religious occasions. Wine is a widely consumed beverage in the world. The evaluation of the quality of the wine is important for merchants, manufactures and consumers. Wine is produced through the fermentation of fruit juices such as grapes, berries and apples and is classified in five major categories. These categories are Table wines, Sparkling wines, Dessert wines, Aperitiv wines, and Pop wines. Wines are characterized into these groups through verification methods, taste, wine style, quality, and vintage i.e. year in which the grapes are harvested. Artificial intelligence can be used in wine classification according to its quality, type, and nutritional value by using chemical analysis with machine learning techniques. In this work, we classify wines according to their cultivar based on the chemical analysis which examines thirteen chemical constituents. The chemical constituents are Alcohol, Malic acid, Ash, Alcalinity, Magnesium, Phenols, Flavanoids, Nonflavanoid Phenols, Proanthocyanins, Colour intensity, Hue, OD280/OD315 of diluted wines, and Proline.

We train artificial neural networks to classify wines according to their respective cultivar. We then use decision trees to extract knowledge from trained neural networks in order to understand the knowledge represented by the trained networks. Finally, we apply decision trees to build a tree structure for classification on the same sets of data sample we used to train neural networks earlier. In this way we combine neural networks and decision trees through training and knowledge extraction and then compare the combined paradigm to the performance of decision trees alone. The extracted knowledge from neural networks is transformed as rules which will help experts in understanding which chemical constituents have a major role in wine classification. The rules contain information for sorting wines according to their cultivar based on the chemical analysis and knowledge acquired by neural networks from training on previous samples. This paradigm is general and can be applied for classification and categorization of food using chemical and physical analysis. We end this paper with conclusions from our work and possible directions for future research.

2 Definition and Methods

2.1 Artificial Neural Networks

Artificial Neural networks learn by training on past experience using an algorithm which modifies the interconnection weight links as directed by a learning objective for a particular application. A *neuron* is a single processing unit which computes the weighted sum of its inputs. The output of the network relies on cooperation of the individual neurons. The learnt knowledge is distributed over the trained networks weights. Neural networks are characterized into feedforward and recurrent neural networks. Neural networks are capable of performing tasks that include pattern classification, function approximation, prediction or forecasting, clustering or categorization, time series prediction, optimization, and control. Feedforward networks contain an input layer, one or many hidden layers and an output layer. Fig. 1 shows the architecture of a feedforward network. Equation (1) shows the dynamics of a feedforward network.

$$S_j^l = g_j\left(\sum_{i=1}^{m} S_i^{l-1} w_{ji}^l - \theta_j^l\right) \tag{1}$$

where S_j^l is the output of the neuron j in layer l. S_i^{l-1} is the output of the neuron i in layer l-1 (containing m neurons) and w_{ji}^l the weight associated with that connection with j. θ_j^l is the internal threshold/bias of the neuron and g_j is the sigmoidal discriminant function.

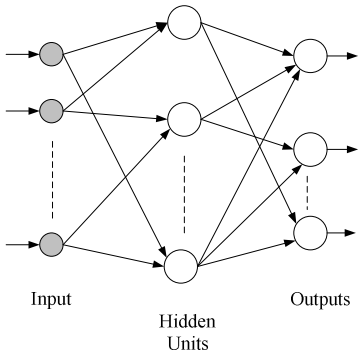

Fig. 1. The architecture of the feedforward neural network with one hidden layer. The diagram shows the interconnections of neurons from one layer to another by links known as weights. Dashed lines indicate that more neurons can be used in each layer depending on the application.

Backpropagation is the most widely applied learning algorithm for neural networks. It learns the weights for a multilayer network, given a network with a fixed set of weights and interconnections. Backpropagation employs gradient descent to minimize the squared error between the networks *output values* and *desired values* for those outputs. The goal of gradient descent learning is to minimize the sum of squared errors by propagating error signals backward through the network architecture upon the presentation of training samples from the training set. These error signals are used to calculate the *weight* updates which represent the knowledge learnt in the network. The performance of backpropagation can be improved by adding a momentum term and training multiple networks with the same data but different small random initializations prior to training. In gradient descent search for a solution, the network searches through a weight space of errors. A limitation of gradient descent is that it may get trapped in a local minimum easily. This may prove costly in terms for network training and generalization performance.

In the past, research has been done to improve the training performance of neural networks which has significance on its generalization. Symbolic or expert knowledge is inserted into neural networks prior to training for better training and generalization performance as demonstrated [8]. The generalization ability of neural networks is an important measure of its performance as it indicates the accuracy of the trained network when presented with data not present in the training set. A poor choice of the network architecture i.e. the number of neurons in the hidden layer will result in poor generalization even with optimal values of its weights after training. Until recently neural networks were viewed as black boxes because they could not explain the knowledge learnt in the training process. The extraction of rules from neural networks shows how they arrived to a particular solution after training.

2.1.2 Knowledge Extraction from Neural Networks: Combining Neural Networks with Decision trees

The goal of knowledge extraction is to find the knowledge stored in the network's weights in symbolic form. One main concern is the fidelity of the extraction process, i.e. how accurately the extracted knowledge corresponds to the knowledge stored in the network. There are two main approaches for knowledge extraction from trained neural networks: (1) extraction of 'if-then' rules by clustering the activation values of hidden state neurons and (2) the application of machine learning methods such as decision trees on the observation of input-output mappings of the trained network when presented with data. We will use decision trees for the extraction of rules from trained neural networks. The extracted rules will explain the classification and categorization of wines according to their cultivator.

In knowledge extraction using decision trees, the network is initially trained with the training data set. After successful training and testing, the network is presented with another data set which only contains inputs samples. Then the generalisation made by the network upon the presentation is noted with each corresponding input sample in this data set. In this way, we are able to obtain a data set with input-output mappings made by the trained network. The generalisation made by the output of the network is an indirect measure of the knowledge acquired by the network in the training process. Finally, the decision tree algorithm is applied to the input-output mappings to extract rules in the form of trees.

Decision trees are machine learning tools for building a tree structure from a training dataset of instances which can predict a classification given unseen instances. A decision tree learns by starting at the root node and selects the best attributes which splits the training data. The root node then grows unique child nodes using an entropy function to measure the information gained from the training data. This process continues until the tree structure is able to describe the given data set. Compared to neural networks, they can explain how they arrive to a particular solution. We will use decision trees to extract rules from the trained neural networks.

3 Results and Discussion

3.1 The Dataset of Wine Samples

Our dataset consists of 178 samples of wines with attribute information given by their chemical constituents: Alcohol, Malic acid, Ash, Alkalinity, Magnesium, Phenols, Flavanoids, Nonflavanoid Phenols, Proanthocyanins, Colour intensity, Hue, OD280/OD315 of diluted wines, and Proline. These data samples are chemical analysis information of wines grown in the same region in Italy but derived from three different cultivators. The values in all attributes were continuous. We randomly chose 80% of the available data for training and set aside the remaining 20% for testing for ten trial experiments.

3.2 Artificial Neural Networks for the Classification of Wines

We ran 10 trial experiments with randomly selected 80% of the available data for training and the remaining 20% for testing the networks generalisation performance. The learning rate of the network in gradient descent learning was 0.5. The network topology used was as follows: 13 neurons in the input layer for each chemical constituent, 4 neurons in the hidden layer and 3 neurons in the output layer representing each wine cultivar as shown in Fig. 2. We carried out some sample experiments on the number of hidden neurons to be used in the networks for this application. The results demonstrate that 4 neurons in the hidden layer were sufficient for the network to learn the training samples. We pre-processed the dataset by normalizing all attribute values (which were inputs to the network) in the range of -2 to 2. All neural networks were trained until one of the three following stopping criteria was satisfied:

1. On 100% of the training examples, the activation of every output unit was within 0.2 of the desired output, or
2. a network had been trained for 500 epochs, or
3. a network classified at least 98% of the training examples correctly, but had not improved it's ability to classify the training after ten epochs

The neurons in the network had a sigmoidal discriminant function and all networks were trained using the standard quadratic error function. The individual results for training time, prediction performance on the training and test sets, respectively, are shown in Table 1. The mean and 90% confidence interval for training and generalization classification performance (both measured in percentage of correctly classified instances) are 99.7±0.5% and 98.7±1.2%, respectively. In this work, we used the C++ programming language in programming neural networks as described by their dynamics in Section 2. Data mining and machine learning software tools such as ‘Weka’ can also be used for classification using neural networks.

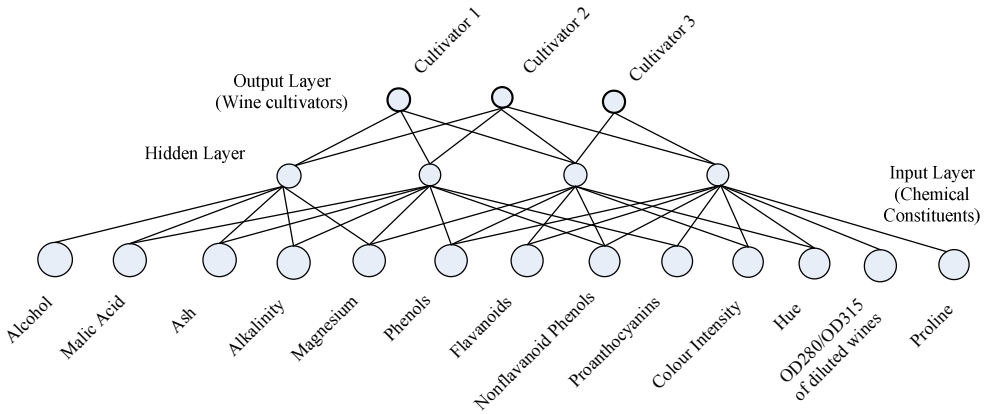

Fig. 2. The figure shows the neural network topology used for training a dataset of wine samples which contain attribute information of thirteen different chemical constituents. Each neuron in the input layer represents a particular chemical constituent as labelled. The neurons in the output layer represent the three major wine cultivars. We found that 4 neurons in the hidden layer were sufficient for training after running some trial experiments. Please note that all weight link interconnections are not shown in this diagram.

Tab. 1. Training and extraction of knowledge from neural networks.

Experiment No.	Training Time (Epochs)	Training Performance	Generalization Performance	Knowledge Extraction
1	88	97%	99.3%	97.0%
2	92	100%	100.0%	94.1%
3	124	100%	100.0%	94.1%
4	101	100%	99.3%	94.1%
5	219	100%	98.6%	94.1%
6	82	100%	92.1%	97.0%
7	90	100%	99.3%	94.1%
8	227	100%	98.6%	94.1%
9	139	100%	100.0%	100.0%
10	500	100%	99.3%	97.0%

3.2.1 Extraction of Knowledge from Neural Networks in the form of Decision Trees

We extracted decision trees from trained neural networks using the J48 decision tree algorithm from the 'Weka' data mining software. For the decision induction, we used the attributes and classification of the 80% training data and the attributes of the remaining 20% test data, but with their label determined by the trained networks. A typical decision tree extracted from experiment No. 9 in Table 1 is shown in Fig. 3. We show this particular decision tree because experiment No. 9 has the best

generalization performance from all experiments in Table 1. The mean and 90% confidence interval obtained from 10 experiments of knowledge extraction from all networks using decision trees is 95.6±1.3%.

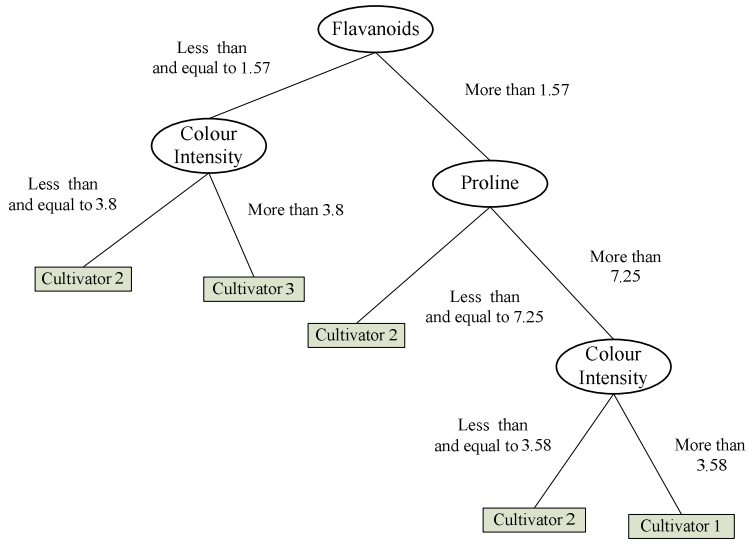

Fig. 3. The figure shows the decision tree extracted from the trained neural network in experiment No. 9 of Table 1. The extracted rules show that three chemical constituents (Flavanoids, Proline and Colour Intensity) play a vital role in the classification of wines according to the relevant cultivator.

The knowledge obtained from the trained network about the presence and absence of the Renosterveld vegetation as shown in Fig. 3 is summarized in the following rules:

1. If the attribute Flavanoids is less than and equal to 1.57, and the Colour Intensity is less than and equal to 3.8, then the wine is from cultivar 2.

2. If the attribute Flavanoids is less than and equal to 1.57, and the Colour Intensity is more than 3.8, then the wine is from cultivar 3.

3. If the attribute Flavanoids is more than 1.57, and the attribute Proline is less than and equal to 7.25, then the wine is from cultivar 2.

4. If the attribute Flavanoids is more than 1.57, and the attribute Proline is more than 7.25, and the Colour Intensity is more than 3.58, then the wine is from cultivar 1.

5. If the attribute Flavanoids is more than 1.57, and the attribute Proline is more than 7.25, and the Colour Intensity is less than and equal to 3.58, then the wine is from cultivar 2.

The prediction performance of the above rules on unseen field data is 100% based from experiment 9 of Table 1. These rules describe the wine category according to the cultivar based on the existing dataset we used for training neural networks. Notice that the rules contain only three attributes which are Flavanoids, Colour Intensity and Proline from the thirteen attribute information in the dataset. The rules suggest that the other attributes are insignificant and does not contribute much to the classification of wines according to their respective cultivar.

3.3 Decision Trees for Wine Classification

We used decision trees for classification on the original dataset which was used for training neural networks previously. This is done to compare the classification performance of neural networks with decision trees. We used the J48 decision tree algorithm from the 'Weka' data mining software. We ran 25 experiments with random fold where we used 80% of the data for training and the remaining 20% for testing. The results for 10 trial experiments are shown in Table 2. The summary of all the experiments with their respective mean and 90% confidence interval is shown in Table 3.

Tab. 2. Decision trees for the classification of wines

Experiment No.	Training Performance	Generalization Performance
1	98.6%	97.1%
2	98.6%	97.1%
3	98.6%	97.1%
4	98.6%	97.1%
5	98.6%	94.1%
6	98.6%	97.1%
7	98.6%	97.1%
8	98.6%	97.1%
9	98.6%	100.0%
10	97.7%	94.1%

Tab. 3. Summary of results

Classification Approach	Training Performance	Generalization Performance	Training Time (Epochs)
Neural Networks	99.7±0.5%	98.7±1.2%	166±67 epochs
Knowledge extraction from trained networks	--	95.6±1.3%	--
Decision Trees	98.5±0.2%	96.8±1.7%	--

We have used wine samples to train neural networks in order to classify then into the categories according to their respective cultivar. Usually customers rank the quality of food products according to the farms that produce them. This paradigm of wine classification can help experts in food processing and screening departments. Using similar approach, we can also classify wines according to their quality, nutritional value, and wine type (Red wine, Rose wine etc.). This artificial intelligence method can also be applied in processing and screening of other food items, drinks and beverages.

4 Conclusion

Neural networks have been successful in the classification of wines according to their cultivar. The prediction accuracy can be increased by having more training instances in the dataset. Decision trees have been useful in knowledge extraction from trained neural networks. They have been a means for knowledge discovery. We have obtained rules which explain the classification of wines according to their cultivars; these rules explain the knowledge acquired in neural networks by learning from previous samples of wine. The extracted rules show that three major constituents, (Flavanoids, Colour Intensity and Proline) have been useful in wine classification. The results demonstrate that artificial neural networks on their own have shown the best generalisation performance in wine classification. However, they cannot explain how they arrive to a solution. The extraction of knowledge from these networks helps experts obtain useful rules. The classification performance of the extracted rules do not

outperform to the performance of decision trees alone. This means that decision trees are the best paradigm for the classification of wines. One can alleviate the time taken for training neural networks and extraction of knowledge from them by using decision trees on their own for classification. This knowledge discovery paradigm can be applied to other food processing domains. The work done in this paper is the foundation for the implementation of decision support systems and screening facilities for food processing.

5 Acknowledgements

We thank Stefan Aeberhard for providing the wine sample dataset at the UMI machine learning repository. The authors also thank the University of Fiji for showing interest in their research work.

References

[1] Singhal P., Zhang J., Michel H.E., Singh B.R.: Fish-Quality Analysis using Artificial Neural Networks and Spectroscopic Data, *Proceeding of the International Conference on Artificial Intelligence and Soft Computing*, 2002.

[2] Pardo M., Sberveglieri G., Masulli F., Valentini G.: Decompositive classification models for electronic noses, *Anal. Chimica Acta*, 446, pp. 223-232, 2001.

[3] Patel V. C., McClendon R. W., Goodrum J. W.: Color Computer Vision and Artificial Neural Networks for the Detection of Defects in Poultry Eggs, *Artificial Intelligence Review*, pp. 163-176, 1998.

[4] Robinson A.J.: An application of recurrent nets to phone probability estimation, *IEEE Transactions on Neural Networks*, 5(2), pp. 298-305, 1994.

[5] Giles C.L., Lawrence S., Tsoi A.C.: Rule inference for financial prediction using recurrent neural networks, *Proceedings of the IEEE/IAFE Computational Intelligence for Financial Engineering. New York City*, USA, pp. 253-259, 1997.

[6] Marakami K., Taguchi H.: Gesture recognition using recurrent neural networks, *Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through Technology*, Louisiana, USA, pp. 237-242, 1991.

[7] Omlin C.W., Snyders S.: Inductive bias strength in knowledge-based neural networks: application to magnetic resonance spectroscopy of breast tissues, *Artificial Intelligence in Medicine*, 28(2), 2003.

[8] Chandra R., Omlin C. W.: Knowledge discovery using artificial neural networks for a conservation biology domain, *Proceedings of the 2007 International Conference on Data Mining*, Las Vegas, USA, In Press, 2007.

[9] Fu L.: Rule generation from neural networks, *IEEE Transactions on Systems, Man and Cybernetics*, 24(8), pp. 1114-1124, 1994.