Australian Journal of Basic and Applied Sciences, 5(9): 1093-1098, 2011
ISSN 1991-8178

Comparison Of Network Pruning And Tree Pruning On Artificial Neural Network Tree

[1]S. Kalaiarasi, [1]S. Sayeed and [2]J. Hossen

[1]Faculty of Information Science and Technology (FIST), Multimedia University (MMU), Malaysia.
[2]Faculty of Engineering and Technology (FET), Multimedia University (MMU), Malaysia.

Abstract: Artificial Neural Network (ANN) has not been effectively utilized in data mining because of its "black box" nature. This issue was resolved by using the Artificial Neural Network Tree (ANNT) approach in the authors' earlier works. The ANNT approach derives symbolic knowledge (rules) to provide some explanation of how the classification or prediction of the ANN is obtained. To enhance extraction, pruning will be incorporate with this approach where two pruning techniques are evaluated to see which method is best to use with ANNT. The first technique is to prune the neural network and the second technique is to prune the tree. The first technique analytically measures the amount of information gained by each of the ANN links as a result learning (training). The one with the lowest information is pruned. Rules are extracted from the pruned network. The second pruning will prune the tree that is built from the neural network and then the tree is converted to rules. These two techniques are evaluated with the ANNT algorithm in the insurance domain to see which method of pruning is most suitable with ANNT in terms of comprehensibility and accuracy.

Key words: Data mining, Artificial neural network, Network pruning, Tree network, Rule extraction.

INTRODUCTION

Data mining is a rapidly evolving research area that is at the intersection of several disciplines, including machine learning, statistics, pattern recognition and artificial intelligence (Zhi-Hua Z. *et al*., 2000). In this area, many conventional techniques such as decision trees are very popular (Han J., 1996). Although Neural Network (ANN) has been successfully used in a wide variety of areas (Zhi-Hua *et al*., 2000; LeCun *et al*., 1990), it has not been well exploited in data mining. This is because the main purpose of data mining is to gain comprehensible knowledge for human being. But in ANN, the knowledge accumulated is not comprehensible, that's why it is regarded as a "black box". It is difficult to explain how the network arrives at a particular conclusion due to the complexity of the network architecture.

In data mining, it is important to know how the ANN arrived at a particular decision. In other words inspecting the internal knowledge of ANN is essential for making the decision process explicit. Extracting rules from trained neural networks is one of the solutions. Rule extraction is used for interpreting ANN and mining the relationship between input and output variables in data.

There are several algorithms for rule extraction from trained ANN. They can be categorized as decompositional, pedagogical and eclectic (Andrews, *et al*., 1995). Decompositional techniques extract rules from each unit in ANN and aggregate them. Pedagogical techniques treat the network as a 'black box' and make no attempt to disassemble its architecture to examine how it works, instead this approach extract rules by examining the relationship between the inputs and outputs. Eclectic approach incorporates both decompositional and pedagogical techniques.

The ideal rule extraction algorithm with respect to the guideline given by Andrews et.al. 1995, has the following criteria. First, it can be applied to any network architecture, secondly, it doesn't need any special training and third, it can be applied to any types of data (continues or discrete). So far, there are a few algorithms that satisfy all the items. Dectext and TREPAN, which is similar to this research in extracting rules using decision tree approach, only satisfy the first and the second item.

Kalaiarasi *et al*., 2005, used Artificial Neural Network Tree (ANNT), i.e ANN training preceded by Decision Tree rules extraction method. ANNT satisfies the three items listed above. ANNT is decompositional approach, where else Dectext and TREPAN is a pedagogical technique.

With ANNT approach, it is proven that ANN can be utilized in data mining applications where comprehensibility is important. With this, the "black box nature" of ANN can be overcome. To further improve on ANNT approach, pruning is used in this work.

This paper is organized as follows: Section II provides a brief description on ANNT algorithm and Network pruning in section III. Section IV presents a short description about Tree pruning. Experimental results in Section V. The final conclusions are drawn in Section VI.

Corresponding Author: S. Sayeed and S. Kalaiarasi, Faculty of Information Science and Technology (FIST), Multimedia University (MMU), Malaysia. Tel: +606 2523296/ +606 2523344;
E-mail: {shohel.sayeed, kalaiarasi}@mmu.edu.my

Artificial Neural Network Tree (Annt) Algorithm:

In the ANNT approach (Kalaiarasi, *et al*., 2005; Kalaiarasi, *et al*., 2006) first the network is trained and rules are extracted from this trained network using the decision tree method. For ANNT illustration, a network with 4 hidden units (figure 1) is trained to classify the lenses (UCI Repository of Machine Learning Databases, 2011) to be fitted on a patient. There are 24 instances belonging to three classes, i.e. to fit hard lenses, to fit soft lenses or no need lenses, described by 4 nominal attributes.

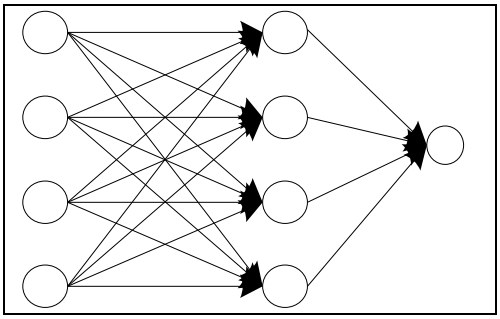

Fig. 1: Trained network.

From the trained network, using the weights and activation pattern at hidden-output (figure 2), a hidden-output tree was build (figure 3).

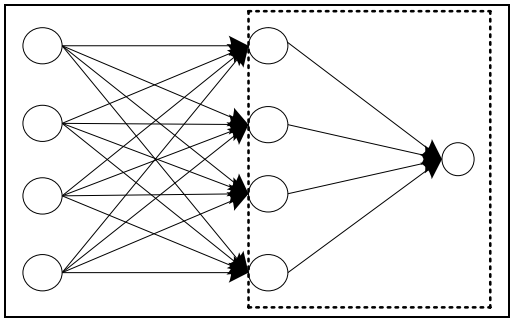

Fig. 2: Hidden-Output network.

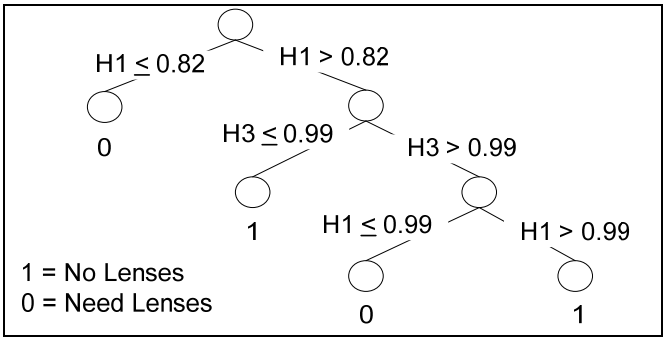

Fig. 3: Hidden-Output tree.

From the decision tree (figure 3), intermediate rules are extracted as shown in figure 4

If Hidden unit 1 (h1) > 0.82 & H3 ≤ 0.99 or

H1 > 0.82 & H3 > 0.99 & H1 > 0.99 Then 1

Fig. 4: Intermediate rules.

Using each hidden unit as output (figure 5), build hidden-output tree (figure 6), and from this tree extract the input-hidden rules.

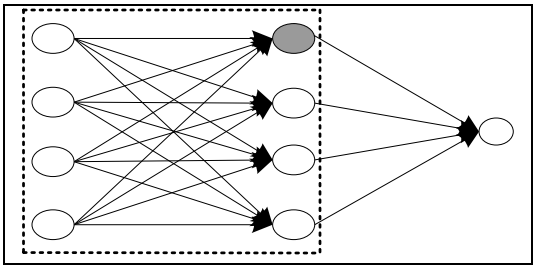

Fig. 5: Input-hidden network.

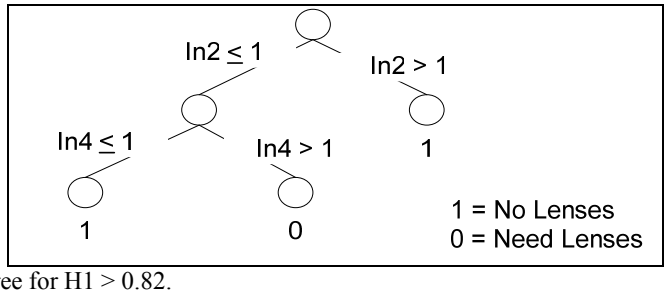

Fig. 6: Input-Hidden tree for H1 > 0.82.

If Input unit (In 2) > 1 or In2 ≤ 1 & In 4 ≤ 1

Then 1

Fig. 7: Input-Hidden rules.

The final rule is obtained by substituting the input rules in the intermediate rules.

Network Pruning:

Pruning eliminates uncontributing links and relieves a network's computational load. Many algorithms for ANN pruning have been proposed in the past few years (Setiono, R., 1996; Karnin, E. D., 1990), where it is often mentioned that pruned network achieve higher accuracy rate on new patterns not used for training. In rule extraction, which is the focus of this paper, pruning aims to extract rules from the contributing links so that the rules extracted will be meaningful and concise and also reduce the number of rules. In others words, pruning is used to simplify the representation of the acquired knowledge.

The concept of information theory has been used by a number of researchers to explain ANN learning algorithms (Haykin, S., 1999; Rolls E.T., 2003). In this research, information theory base pruning method is used to prune the network.

Information theory is used to measure the classification performed by the network links and the information embedded in data, that is information theory is used to calculate the information gained of each link after learning. The link with lowest information will be pruned because low information indicates that the link has irrelevance or marginal contribution to network output

Information Gain:

Two main concepts of information theory (Quilan, J., 1993) are entropy and information. Entropy is the uncertainty of our knowledge about which outcome will actually happen: the less sure we are about the outcome, the higher the entropy. If the outcome is known for sure, then the entropy will be zero. Information is gained by reducing entropy, for example by making an observation of an outcome. Before the observation, our knowledge of the outcome is limited (uncertain), after observation the uncertainty (entropy) is reduced to zero. The difference before and after observation is the information gained.

The overall entropy of a given dataset with respect to the classes computed as follows:

$$\text{Entropy(classes)} = -\sum(\frac{n_i}{N})\log(\frac{n_i}{N}) \quad (1)$$

where $_{ni}$ is the number of patterns belonging to specific class i, i= 1, 2,….c and N is the number of records in the overall data set.

The entropy of the attribute, Entropy (attribute), is computed as follows.

$$\text{Entropy(attribute)} = -\sum \frac{k_j}{N}(\sum(\frac{n_i}{k_j})\log(\frac{n_i}{k_j})) \quad (2)$$

where k_j is the number of subclasses j, j=1,2….s, n_i is the number of patterns belonging to the specific subclass i, i=1,2….c and N is the number of records in the overall data set.

Next, the information gain for each of the attribute is calculated with respect to the data set's entropy using the following formula.

$$\text{Information Gain(attribute)} = \text{Entropy(classes)} - \text{Entropy(attribute)} \quad (3)$$

For continuous variables, the entropy measure can be defined as:

$$\text{Entropy} = -\int p(x)\log p(x)dx \quad (4)$$

Information Pruning (Infoprune):

Learning is by definition an acquisition of information, so ANN learning is a process of accumulating and distributing information by training process. In training, the input data's are converted to information and distributed among the links. In the proposed pruning approach, we are using information of each link and prune the less informative links. In other words we are measuring the level of information a link possesses at a given point of time (after training) that is we are attempting to find if all the links in the network are contributing to the classification. This information (each link), is added to the weight of the link to compute the saliencies. A low saliency is assumed to imply that the link's contribution to the classification is marginal. The pruning is done by ranking the link according to saliency and then removing the least significant ones.

Assume H_i is the output of hidden node i, with n_i number of cases belonging to the i – th class (i = 1,2), the entropy before observation (for simplicity the calculation for discrete data is shown).

The overall entropy of a given dataset (H_i) with respect to the classes computed using Eqn., 1 as follows:

$$\text{Entropy}(H_i) = -\sum(\frac{n_i}{H_i})\log(\frac{n_i}{H_i}) \quad (5)$$

To calculate the information on each links (e.g. link S_i) use Eqn. 2-3

$$\text{Entropy(link Si)} = -\sum \frac{k_j}{H_i}(\sum(\frac{s_i}{k_j})\log(\frac{s_i}{k_j})) \quad (6)$$

$$\text{InfolinkS}_i = \text{Entropy}(H_i) - \text{Entropy}(\text{linkS}_i) \quad (7)$$

where k_j is the number of subclasses j, j=1,2….s, s_i is the number of links belonging to the specific subclass i, i=1,2….c and H_i is the total number of hidden i output. The same method is used to calculate the information of links between the hidden to output unit. For detailed discussion, refer to Quilan, J., 1993.

Next the saliencies of the of each link is calculated

$$\text{Saliencies} = \text{InfolinkS}_i \times \text{WlinkS}_i \quad (8)$$

Info Prune Algorithm:

The proposed pruning algorithm using information theory (Infoprune) is given, which will be used in this research to prune the uncontributing links. Since ANNT is based on information gain and entropy (decision tree) to extract the rules, InfoPrune is suitable for ANNT pruning.

Step 1) Train a network

Step 2) Compute saliencies of each link

Step 3) Sort links by saliency and delete the lowest saliency link

Step 4) Back to step 1 (until 85% link deleted)
Step 5) Check the accuracy of the network, the best architecture is saved

Tree Pruning:

The reduced error pruning based on statistical confidence estimation (Quilan, J., 1993) is used to prune the decision tree. This pruning technique calculates the confidence interval for the error. In this research the tree that is build will be pruned before the rules are extracted, where else in network pruning, the network will be pruned and then only converted to tree then to rules.

Experimental Results:

In this experiment, the performance of network pruning and tree pruning are studied in connection with rule extraction from ANN in insurance domain.

Insurance data set was collected from Prudential and Great Eastern Insurance company within Malaysia to determine whether a client will pay premium more than RM200 a month. There are a total of 203 samples in the data set, of which 130 will pay. This data set is described by eight attributes that are age, education level, annual salary, occupation risk, race, sex, smoking habit, and marital status. Out of the eight attributes, two are continuous value and the rest are discrete.

This data set was randomly divided into two subsets: the training set (60%), and the test set (40%). Five groups each with six networks were trained. Each group will have same training data set with randomized initialize weight. The average results of the groups are regarded as the result of the group. It should be mentioned that the architecture of the networks has not been finely tuned (have been standardized for all the data set for comparison). The hidden neurons are activated using bipolar sigmoid activation function with linear output units. The learning rate is fixed at 0.001 with maximum epoch fix at 2000 with 9 hidden neurons.

For network pruning, an ANN is trained until it reaches the maximum number of iterations or the Mean Squared Error drops to less than 0.01. Then, the network is pruned a link at a time and the accuracy of the network is stored. After pruning a link, the network is again retrained 20 iterations and this process will continue until 85% of the entire links are removed. The architecture with the best classification will be saved to extract the rules.

Table 1: Classification Accuracy of Different Techniques.

Techniques	Min	Max	Mean	SD
ANN	78.3	84.2	80.9	2.2
ANNT	77.8	84.0	81.1	2.1
Network Pruning	79.8	84.0	82.0	1.7
Tree Pruning	77.8	83.5	79.3	3.7

Table 1 shows the performance classification (accuracy) of ANN, the knowledge translated from ANN in the form of rules (ANNT), ANNT from network pruning and ANNT from tree pruning. The tables represent the minimum (Min) classification, maximum (Max) classification, average (Mean) classification (in percentage) and standard deviation (SD) of the overall average of the five groups for testing data set for a insurance domain. The minimum and the maximum classification are included in the table to check the consistency of ANN with ANNT. The overall performance of ANNs is determined by measuring the classification accuracy achieved by the ANN on an unseen pattern (test set) for each data set. The average generalization accuracy of ANNs is 80.9%, where else ANNT is 81.1%. ANNT with network pruning is 2.7% better than ANNT from tree pruning.

Table 2: Gives the Comprehensibility and Fidelity Performance of Different Techniques.

Techniques	Number of Rules	Fidelity
ANNT	47.4	96.1
Network Pruning	41.6	97.5
Tree Pruning	25.5	95.4

Table 2 compared the comprehensibility and fidelity performance of ANNT with different pruning methods. The rule comprehensibility is measured in terms of the number of rules Fidelity measures how well ANNT represents the trained ANN in symbolic form. Each of these values represents the average of the five groups. Both of the pruning techniques reduce the number of rules where the decrease is 46.5% for tree pruning compared to 4.2% decrease using network pruning. ANNT combined with either of the pruning techniques increases the comprehensibility of the rules extracted. That is, pruning reduces the number of redundant rules.

As shown in Table 2, the fidelity of the rules extracted from network pruning increases by 1.6% and the rules extracted from tree decrease by 0.4%. In other words, the rules extracted using network pruning represent the ANN better then tree pruning.

Conclusion:

Artificial Neural Network (ANN) has not been effectively utilized in data mining because of its "black box" nature. This issue was resolved using the ANNT. To enhance this algorithm, in this paper two pruning techniques are proposed and evaluated. The first pruning technique is to prune the neural network and the second technique is to prune the tree. The idea behind these pruning methods is to evaluate which method of pruning is suitable to improve ANNT in terms of comprehensibility and accuracy. The experiment results show that, network pruning improves the accuracy and the fidelity of the rules extracted for this domain. On the other hand tree pruning improve the comprehensibility of the rules. Whether network pruning or tree pruning is better depends on to the needs of the user or application in which the model is being applied. If the user wants to increase comprehensibility, then tree pruning is better with ANNT compared to network pruning. If user want to increase the accuracy or fidelity, network pruning is better.

REFERENCES

Andrews, R., J. Diederich and A.B. Tickle, 1995. Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge-Based Systems, 8(6): 373-389.

Han, J., 1996. Data Mining techniques, Tutorial at the 1996 ACM-SIGMOD International Conference on Management of data, Montreal, Canada.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation. Addison Wesley, 2nd edition, 1999.

Kalaiarasi, S., G. Sainarayanan, A. Chekima and J. Teo, 2005. Data Mining using Artificial Neural Network Tree, in Proceedings of IEEE 1st International Conference on computers, Communications & Signal Processing with Special Track on Biomedical Engineering, Kuala Lumpur.

Kalaiarasi, S., G. Sainarayanan, A. Chekima and J. Teo, 2006. Data Mining using Pruned Artificial Neural Network Tree (ANNT), in Proceedings of IEEE 2st International Conference on Communications Technologies: from Theory to Applications, Syria.

Karnin, E.D., 1990. A simple procedure for pruning back propagation trained neural networks, IEEE Trans. On Neural Networks, pp: 239-242.

LeCun, Y., B. Boser, J.S. Denker, D. Henderson, R.E. Horward, W. Hubbard and L.D. Jackel, 1990. Backpropagation applied to handwritten zip code recognition". Neural Computations, 1: 541-551.

Quilan, J., 1993. C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA.

Rolls, E.T., L. Franco, N.C. Aggelopoulos and S. Reece, 2003. An Information Theoretic Approach to the contributions of the Firing Rates and the Correlations Between the Firing of Neurons, Journal Neurophysiology. Vol 89,

Setiono, R., 1996. Extracting Rules from Pruned Neural Networks. Artificial Intelligence in Medicine, 8(1): 37-51.

UCI Repository of Machine Learning Databases (2011). http://www.ics.uci.edu/ ~mlearn/MLRespository.

Zhi-Hua, Z., Y. Jiang and S. Chen, 2000. A General neural Framework for Classification Rule Mining. International Journal of Computer Science and Security, 1(2).