**Available Online at www.ijcsmc.com**

**International Journal of Computer Science and Mobile Computing**

**A Monthly Journal of Computer Science and Information Technology**

**ISSN 2320–088X**

***IJCSMC, Vol. 3, Issue. 5, May 2014, pg.776 – 779***

**RESEARCH ARTICLE**

# Extraction of Classification Rules from Trained ANN for Multiclass Data

**Ankita Patel[1], Mahesh Panchal[2]**

Computer & Science Engineering, Kalol Institute of Technology and Research Canter, Kalol, Gujarat, India

[1] ankipatel.13@gmail.com; [2] mkhpanchal@gmail.com

***Abstract— In Data mining classification rules are generated using some classification methods. One of the most common methods for rule generation is Decision tree which refers supervised learning. After generating classification rules we can apply those rules on unknown data and reach to the results. Artificial Neural Network is one kind of network that has the ability to learn and thereby acquire knowledge and make it available for use. But ANN gives results of only trained data. Unseen samples can be classified by ANN but nobody can understand how that classification was made. All training is inside the ANN in the form of weights. So we require to extract that training (knowledge) in form of rules. Classification rules are generated from trained ANN for multi-class dataset by Rule Extraction methods. So during the phase of testing classification rules give results of testing data.***

***Keywords— Decision tree, Artificial Neural Network, Rule Extraction methods, Multiclass Data***

## I. INTRODUCTION

Artificial neural networks (ANNs) have been successfully applied to solve a variety of classification and function approximation problems. [1] Although ANNs can generally predict better than decision trees for pattern classification problems, ANNs are often regarded as black boxes since their predictions cannot be explained clearly like those of decision trees.[2] A standard three-layer feed forward ANN with four-phase training is the basis of the proposed algorithm. In the first phase, the number of hidden nodes in ANNs is determined automatically by a constructive algorithm. In the second phase, irrelevant connections and input nodes are removed from trained ANNs without sacrificing the predictive accuracy of ANNs. The continuous activation values of the hidden nodes are discretized by using an efficient heuristic clustering algorithm in the third phase. Finally, rules are extracted from compact ANNs by examining the discretized activation values of the hidden nodes. It can generate high quality rules from ANNs, which are comparable with other methods in terms of number of rules, average number of conditions for a rule, and predictive accuracy. [3] It is therefore desirable to have a set of rules to explain how ANNs solve a given problem. This is because the functionality of ANNs represented by a set of rules will be more comprehensible to human users than a set of connection weights of ANNs. [4]

 

## II. LITERATURE REVIEW

### *A. An Algorithm to Extract Rules from Artificial Neural Networks for Medical Diagnosis Problems.* [5]

In this paper rules are extracted from medical diagnosis problems such as breast cancer problem, diabetes problem and lenses problem. This paper presents a new algorithm, called rule extraction from ANNs (REANN), to extract rules from trained ANNs for medical diagnosis problems. A standard three-layer feed forward ANN with four-phase training is the basis of the proposed algorithm. breast cancer, diabetes and lenses, demonstrate that REANN can generate high quality rules from ANNs, which are comparable with other methods in terms of number of rules, average number of conditions for a rule, and predictive accuracy. The experimental results on three different problems show that REANN can able to explain the functionality of ANN by extracting simple and concise rules. The predication accuracy of rules generated by REANN for different problems is also encouraging in comparison with exiting works.

### *B. Extracting Rules from Neural Networks as Decision Diagrams.* [6][9]

Rule extraction from neural networks solves two fundamental problems: it gives insight into the logic behind the network and, in many cases, it improves the network's ability to generalize the acquired knowledge. This article presents a novel, eclectic approach to rule extraction from neural networks, named LORE, suited for multilayer perceptron networks with discrete (logical or categorical) inputs. The extracted rules mimic network behaviour on training set and relax this condition on the remaining input space. First, a multilayer perceptron network is trained under standard regime. It is then transformed into an equivalent form, returning the same numerical result as the original network, yet being able to produce rules generalizing the network output for cases similar to a given input. The partial rules extracted for every training set sample are then merged to form a decision diagram from which logic rules can be extracted. A rule format explicitly separating subsets of inputs for which an answer is known from those with an undetermined answer is presented. A special data structure, the decision diagram, allowing efficient partial rule merging is introduced. With regard to rules' complexity and generalization abilities, LORE gives results comparable with those reported previously. An algorithm transforming decision diagrams into interpretable Boolean expressions is described. Experimental running times of rule extraction are proportional to the network's training time.

### *C. Iris Classification Problem.* [7]

Dataset: 150 instances.
Input: Four continuous attributes- Sepal Length, Sepal Width, Petal Length, Petal Width.
Output: Three different Iris flowers- Setosa, Versicolor, Virginica.
The network has 2 hidden units and 10 connections after pruning.

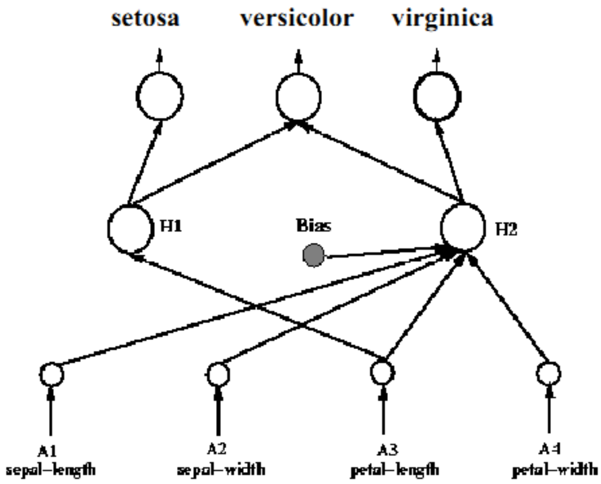


Fig. 1 Iris Network with 2 hidden units

Rules in terms of the Hidden unit activations:

If H1 > -0.7 : Iris Setosa.
Else if H2 < -0.55 : Iris Versicolor.
Else : Iris Virginica.

## III. PROPOSED APPROACH

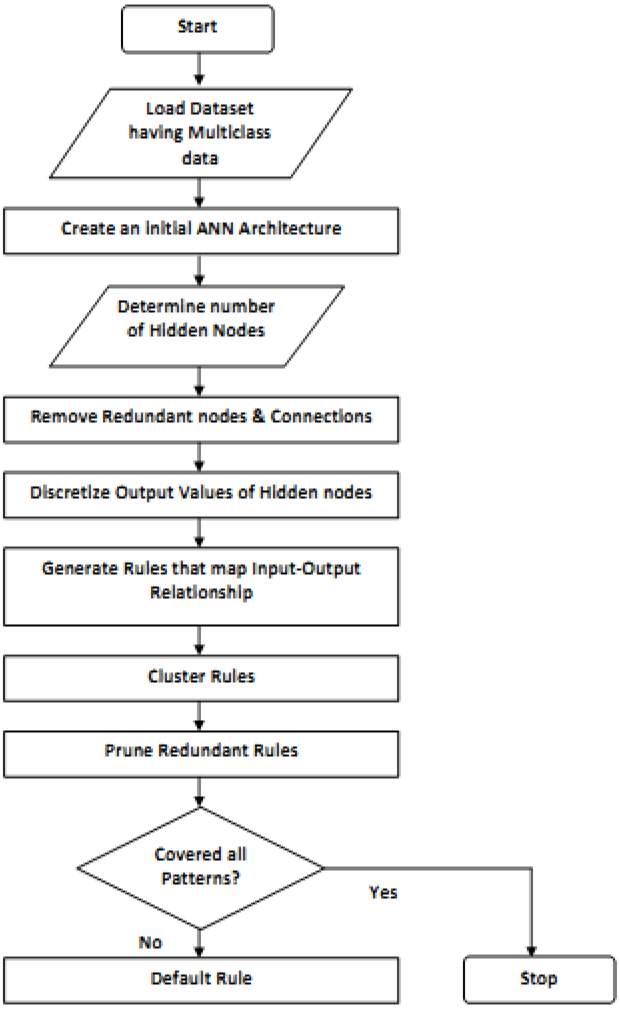


Fig. 2 Rule extraction Algorithm

1. Create an initial ANN architecture. The initial architecture has three layers i.e. an input, an output and a hidden layer. The number of nodes in the input and output layers is the same the number of inputs and outputs of the problem, respectively. Randomly initialize all connection weights of the ANN within a small range.
2. Determine the number of hidden nodes in the ANN by using a basic constructive algorithm.
3. Remove the redundant input nodes and connections by using a basic pruning algorithm. When pruning is completed, the ANN architecture contains only important nodes and connections.
4. Discretize the outputs of hidden nodes by using an efficient heuristic clustering algorithm. The reason for discretization is that the outputs of hidden nodes are continuous therefore rules cannot be easily extracted from the ANN.
5. Extract Rule:

   i=0; while (data is NOT empty/marked) {
   generate Ri to cover the current pattern and differentiate it from patterns in other categories;
   remove/mark all patterns covered by Ri ; i++}

6. Cluster Rule: Cluster rules according to their class levels. Rules generated in Step 5 are grouped in terms of their class levels. In each rule cluster, redundant rules are eliminated; specific rules are replaced by more general rules.
7. Prune Rule: replace specific rules with more general ones; remove noisy rules; eliminate redundant rules.
8. Check whether all patterns are covered by any rules. If yes then stop, otherwise continue.
9. Determine a default rule: A default rule is chosen when no rule can be applied to a pattern.

## IV. CONCLUSION AND FUTURE WORK

Although ANNs have been widely used to solve many problems, they are often viewed as black boxes. This work is an attempted to open up these black boxes by extracting rules from trained ANNs by the proposed rule extraction algorithm. The experimental result on sales forecasting problem show that Rule Extraction algorithm can able to explain the functionality of ANN by extracting simple and concise rules. The features of rules generated by Rule Extraction algorithm for the problems are also encouraging in comparison with exiting works. After rule generation, apply those rules on some unseen data for testing, which will measure accuracy of rules generated by this method.

**REFERENCES**


1. Han Jiawei and Kamber Micheline, "Classification of Data Mining Systems", *Data Mining: Concepts and Techniques*. Second edition, 29-31.
2. Han Jiawei and Kamber Micheline, "Architecture of typical Data Mining system", *Data Mining: Concepts and Techniques*. Second edition, 5-9.
3. Han Jiawei and Kamber Micheline, "Classification and Prediction", *Data Mining: Concepts and Techniques*. Second edition, 285-296.
4. Han Jiawei, Department of computer science, University of Illinois at Urbana-Champaign, 2006 *chapter-6,* www.cs.uiuc.edu/~hanj .
5. S. M. Kamruzzaman and Md. Monirul Islam, "*An Algorithm to Extract Rules from Artificial Neural Networks for Medical Diagnosis Problems,"* International Journal of Information Technology, 2006, Vol. 12, 8, 41-59.
6. Jan Chorowski, Jacek M. Zurada, "*Extracting Rules from Neural Networks as Decision Diagrams",* IEEE Transactions On Neural Networks, 1-12.
7. Rudy Setiono,*Rule Extraction from Artificial Neural Networks*, School of Computing, National University of Singapore.
8. Atiya Amir, "Learning algorithms for Neural Networks", Ph.D. dissertation, California Institute of Technology.
9. R. Setiono and H. Liu, "Symbolic presentation of neural networks," IEEE Computer, March 1996, 71-77.
10. R. Setiono, Generating concise and accurate classification rules for breast cancer diagnosis, Artificial Intelligence in Medicine, 2000, Vol 18, No. 3, 204-219.


 