Journal of Computer Applications ISSN: 0974 – 1925,
Volume-5, Issue EICA2012-1, February 10, 2012

Credit Card Fraud Detection using Rough Sets and Artificial Neural Network

Nikita Gupta,
Dept of CSE,
Army Institute of Technology,
Alandi Road, Dighi Hills,
Pune -411015
E-mail: gupta_n ikita 12@ yahoo.co.in

ArchanaDhankar,
Dept of CSE,
Army Institute of Technology, Alandi Road, Dighi Hills,
Pune-411015
E.mail:archana19dhankar @gmail.com

KaranSinghRanawat,
Dept of CSE,
Army Institute of Technology, Alandi Road, Dighi Hills,Pune-411015
E.mail:karanranawat@gmail .com

ShivyaNautiyal,
Dept of CSE,
Army Institute of Technology, Alandi Road, Dighi Hills,
Pune-411015
E.mail: shivya91 @gmail.com

Abstract— We propose an approach towards detecting Credit Card Frauds using Rough Set Theory and Artificial Neural Networks. We aim at finding an optimal attribute Reduct using rough set theory and then using these attributes to train an ANN system which would then carry out the credit card fraud detection. This hybrid model will maximize the correct diagnosis by minimizing both the number of false alarms and the number of fraud transactions not recognized. The foremost step in our proposed model is the reduction of the transaction data set to ensure a manifold decrease in space and time requirement for fraud detection. This pre-processed data will then be used for learning and validating the system.

Keywords: *Artificial neural network, Back Propagation Algorithm, Credit card frauds, Feed Forward Network, Lower Approximation, Rough settheory, Upper Approximation.*

I. INTRODUCTION

Credit card fraud is emerging as one of the greatest financial challenges faced by humanity in this era. In the US alone, losses from all types of credit card fraud are projected to exceed $850 million, representing a 10% increase in fraud losses over 1991[1]. Since only one financial transaction in a thousand is fraudulent, an accuracy rate less than 99.9% is unacceptable[2] .

The proposed approach uses a combination of Rough Set Theory and ANN which exhibits various advantages over other conventional techniques. An advantage of the Rough Sets theory is that it does not need any assumptions about the independence of the attributes nor does it require any background knowledge about the data. A neural network also has some advantages over a linear program as it can perform tasks that the latter cannot. When an element of the neural network fails, it can continue without any problem because of their parallel nature. A neural network learns and does not need to be reprogrammed.

II. ROUGH SET THEORY

Rough set theory was introduced by ZdzislawPawlak [3] in 1982. It is a new mathematical tool to deal with partial information. It deals with the classificatory analysis of data

tables. A rough set is a formal approximation of a crisp set (i.e., conventional set $[x]_P$) in terms of a pair of sets which give the lower and the upper approximation of the original set of attributes P.

- **Lower Approximation and Positive Region**

The P-lower approximation, or positive region, is the union of all equivalence classes in $[x]_P$ which are contained by (i.e., are subsets of) the target .The lower approximation is the complete set of objects in $\mathbb{U}/P$that can be positively (i.e., unambiguously) classified as belonging to target set X [7].

- **Upper Approximation and Negative Region**

The P-upper approximation is the union of all equivalence classes in $[x]_P$ which have non-empty intersection with the target set .The upper approximation is the complete set of objects that in $\mathbb{U}/P$that cannot be positively classified as belonging to the complement ($\overline{X}$) of the target set X. In other words, the upper approximation is the complete set of objects that are possibly members of the target set X.

The set $\mathbb{U} - \overline{P}X$therefore represents the negative region, containing the set of objects that can be definitely ruled out as members of the target set [7].

The tuple $\langle \underline{P}X, \overline{P}X \rangle$composed of the lower and upper approximation is called a Rough Set.

The accuracy of the Rough-Set representation of the set X can be given (Pawlak 1991) by the following:

$$\alpha_P(X) = \frac{|\underline{P}X|}{|\overline{P}X|} \quad (1)$$

This provides a measure of how closely the rough set is approximating the target set[6].

- **Reduct and Core**

Reduct can be thought of as a sufficient set of features – sufficient, that is, to represent the category structure.The set of attributes which is common to all Reducts is called the Core i.e. it is the set of attributes which is possessed by every legitimate Reduct [10]

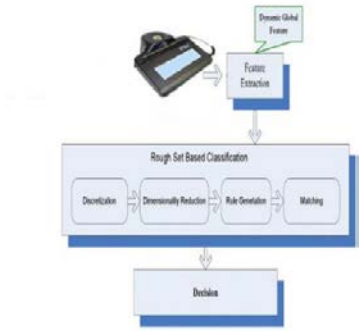

Figure 1: Framework of the proposed Rough Set system

III. ATTRIBUTE SET FOR CREDIT CARDS

Credit card faud detection system is a data driven processing system. And the timely detection of abnormal transaction requests need to be identified as fast as possible.Thus appropriate selection of attributes is a critical step for increasing the efficiency of the model. This ensures elimination of redundant features and hence decreases the time complexity of fraud detection. The use of Rough set theory has been proposed for the same to reduce the space dimension as well as the training time.

A. Feature Selection using ROSE

ROSE2 (Rough Sets Data Explorer) is a software implementing basic elements of the Rough Set theory and rule generation techniques [4],[5]. It has been created at the Laboratory of Intelligent Decision Support Systems of the Institute of Computing Science in Poznan [8]
ROSE is a micro computer software designed to analyze data by means of the Rough Set theory. Ithas a modular architecture consisting of an integrated environment and external executable modules.

The system contains several tools for rough set based knowledge discovery, e.g.

- data preprocessing, including discretization of numerical attributes
- performing a standard and an extended rough set based analysis of data
- search of Reducts and a Core of attributes permitting data reduction
- inducing sets of decision rules from rough approximations of decision classes
- evaluating sets of rules in classification experiments,

B. The Rough Set Methodology

Step 1: Identification of the type of an attribute:

- The conditional attributes: These are represented by the 24 fields in our data set corresponding to the various user credit card details given during an online purchase.
- The decision attributes: representing whether the given transaction details are fraudulent or not.

Step 2: Calculation of equivalence class using the Decision attribute.
Step 3: Calculation of equivalence class using the conditional attributes.
Step 4: Generation of rules based on the comparison of the equivalence classes obtained in steps 1 and 2.
Step 5: Simplification of the dataset

- Calculation of attribute Reduct-The dispensable attributes can be dropped without Affecting the decision rules
- Calculation of the value Reduct for each rule[14]

IV. ARTIFICIAL NEURAL NETWORKS

It is a computing paradigm designed to mimic the human brain and functioning of the nervous system.

A. Neuron : The Basic Building Block

The black box section of the neuron consists of an activation function F(X), in our case its F(wSum – T) where wSum is the weighted sum of the inputs and T is a threshold or bias value[9].

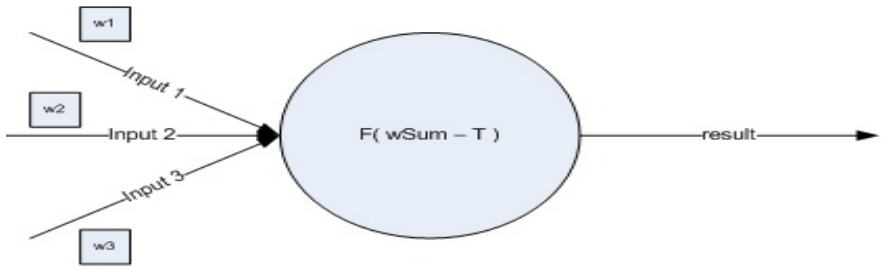

Figure 2: The Neuron as a Black Box

$$wSum = \sum_{i=1}^{n} weight_i \times input_i \quad (2)$$

Where, The most common activation functions include the step function and the sigmoid function. We will be using the sigmoid function in our proposed system.

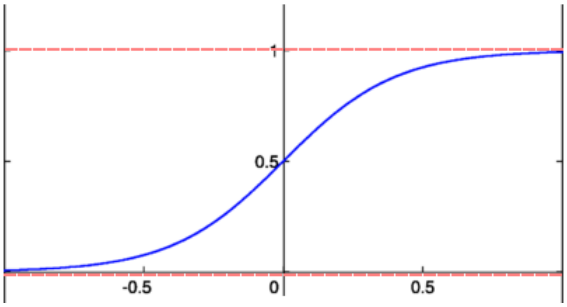

Figure 3: Graph of the sigmoid Function

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

The sigmoid function has an asymptotic nature and is easily differentiable.

B. The Learning Procedure

Machine Learning is primarily of three types : Supervised, Unsupervised and Reinforced learning. In supervised learning, the training data consist of a set of training examples where each sample instance consists of an input vector with a class label ,i.e output of the data is known a-priori. Back Propagation Method is an example of the gradient descent type of Supervised Leaning.

We will be using a Feed forward network where the value of the output (the value of the activation function for the weighted sum of inputs) at a neuron is calculated and then used as the input for the next layer.

The basic procedure is as follows:

- Processing an input pattern through the function
- Calculating the error (Mean Square Error)
- Updating the weights according to learning rate and error
- Repeating the process with the next pattern

C. Back Propagation Method

It is the method we will be using for adjustment of weights. The input is fed in and the errors are calculated and filtered back though the network making changes to the weights to try and reduce the error.

Back Propagation (generalized gradient descent) is a generalization of the LMS (least mean square) algorithm..

The Mean squared error is used as the performance index. This is achieved by adjusting the weights. The generalized delta rule does this by calculating the error for the current input example and then back propagating this error from layer to layer.

$$MSE = \frac{\sum_{i=0}^{n}(desired\ value\ i - actual\ value\ i)^2}{n}$$

$$where\ n = number\ of\ patterns\ in\ set \qquad (4)$$

The Back Propagation Network learns during a training epoch described as follows:

For each input entry in the training data set:

- The input data is fed in (feed forward).
- The output is checked against the desired value and the error is fed back (back-propagate).

Where back-propagation consists of :

- Calculation of the error gradients.
- Updating of weights.

These are some commonly used stopping conditions used for neural networks: setting the desired accuracy , setting the desired mean square error and specifying an upper limit on the elapsed epochs[9].

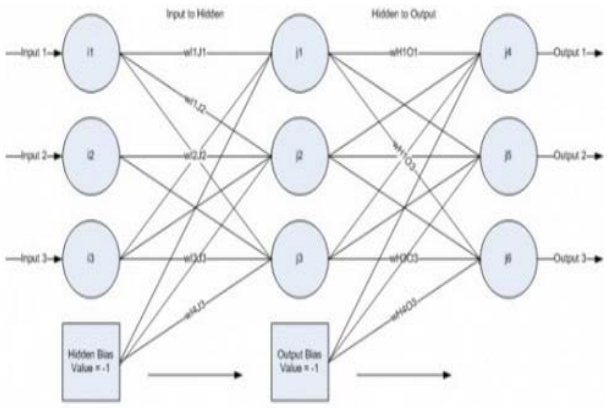

Figure 4: Architecture of a basic multilayer neural network

I. The Neuron Error Gradients

The weight changes are calculated by using the gradient descent method. This means we follow the steepest path on the error function to try and minimize it.

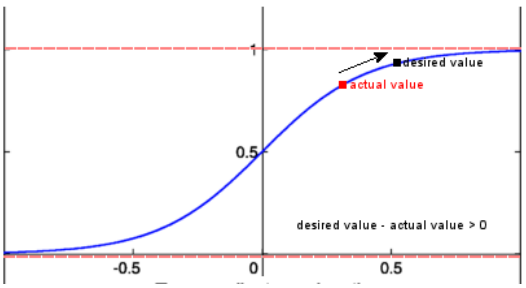

Figure 5: Error Gradient

This is the formula to calculate the basic error gradient for each output neuron k:

$$\delta_k = y_k(1 - y_k)(d_k - y_k) \quad (5)$$

Where, y_k is the value at output neuron k and d_k is the desired value at output neuron k.

II. The Weight Update

The final step in the algorithm is to update the weights, this occurs as follows:
The alpha value is the learning rate , this is usually a value between 0 and 1.

$$w_{ij} = w_{ij} + \Delta w_{ij} \text{ and } w_{jk} = w_{jk} + \Delta w_{jk}$$

$$\text{where } \Delta w_{ij}(t) = \alpha . inputNeuron_i . \delta_j$$
$$\text{and } \Delta w_{jk}(t) = \alpha . hiddenNeuron_j . \delta_k$$

$$\alpha - learning\ rate$$
$$\delta - error\ gradient \quad (6)$$

It affects how large the weight adjustments are thus also alters the learning speed of the network. This value needs to be carefully selected to provide the best results.

Heuristic modifications of backpropagation

Following are the two heuristic modifications that can be made to the Back Propoagation Method for enhanced performance and improved efficiency :

- Momentum
- Learning rate

Back propogation is also called generalized gradient descent and generalized delta rule. The aim is to find the global minimum in the Back Propagation error space.

The use of momentum will smooth out the oscillations and produce a stable trajectory by averaging theupdates to the parameters that helps in faster converge.

The only change to the implementation is in regards to the weight updates[9]

$$w_{ij} = w_{ij} + \Delta w_{ij} \text{ and } w_{jk} = w_{jk} + \Delta w_{jk}$$

$$\text{where } \Delta w_{ij}(t) = \alpha . inputNeuron_i . \delta_j \text{ and } \Delta w_{jk}(t) = \alpha . hiddenNeuron_j . \delta_k$$

$$\alpha - learning\ rate$$

$$\delta - error\ gradient$$

Those weight updates now become:

$$\Delta w_{ij}(t) = \alpha . inputNeuron_i . \delta_j + \beta . \Delta w_{ij}(t-1)$$

$$\Delta w_{jk}(t) = \alpha . hiddenNeuron_j . \delta_k + \beta . \Delta w_{jk}(t-1)$$

$$where\ \beta - momentum\ constant \quad (7)$$

The learning rate is constant throughout training (standard steepest descent). The performance is very sensitive to the proper setting of the learning rate

Very small value will cause slow convergence while too big will have oscillation, overshooting of the minimum

It is not possible to determine the optimum learning rate before training as it changes during training and depends on the error surface. The goal of variable learning rate is to keep the learning rate as large as possible while keeping learning stable.

Typically the number of hidden neurons and hidden layer is determined by trial and error.

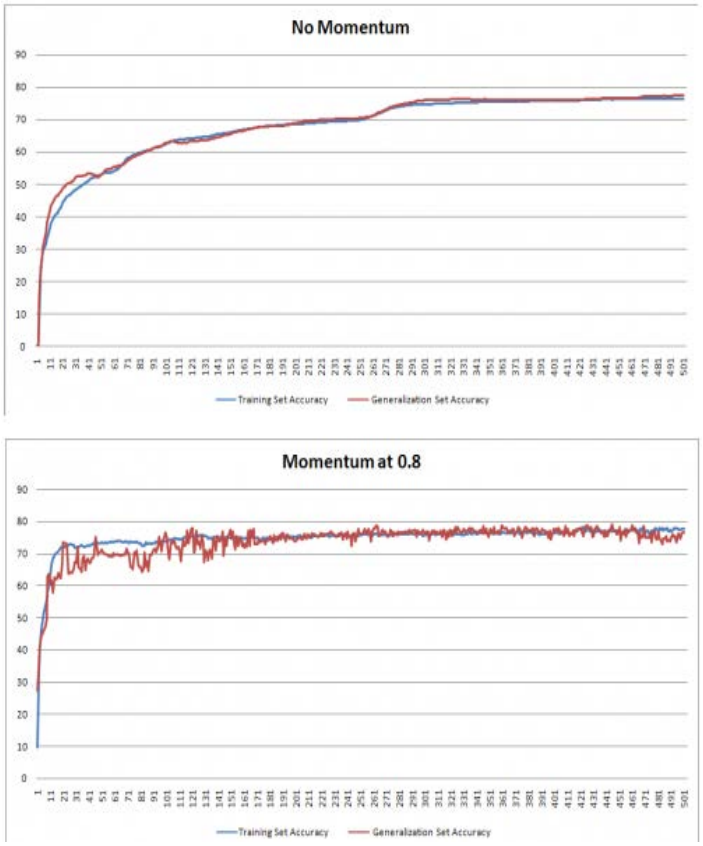

Figure 6: Comparison of learning with and without momentum

The task constrains the number of inputs and output units but not the number of hidden layers and neurons in them[15].

- Too many free parameters (weights) – overtraining
- Too few – the network is not able to learn the input-output mapping
- A heuristic approach would be to start with: 1 hidden layer with n hidden neurons
  n = (inputs + output_neurons)/2

III. EXPERIMENTAL RESULTS

A. Attribute Reduct

The Reduct is calculated using ROSE2 software

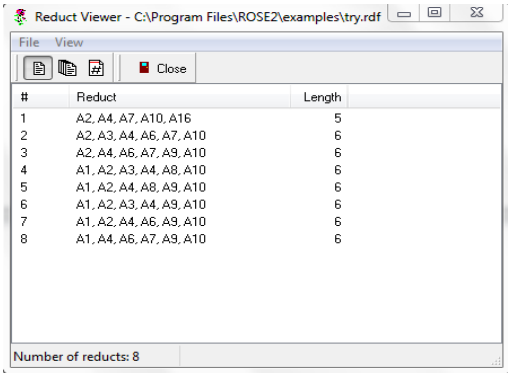

Figure 7: Attribute reduct

B. ANN learning

The Back Propagation Algorithm is implemented and is trained for a given dataset. The Input is first divided into training set, generalization and validation set. The mean square error and accuracy is calculated..Our priliminary results have yielded the following results:

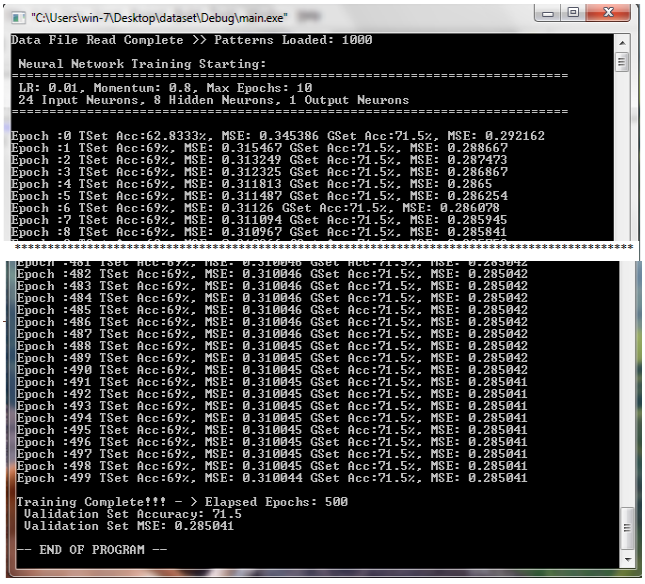

Figure 8: Efficiency of Back propagation

- Validation Set Accuracy = 71.5%
- Validation Set Mean Square Error = 0.285041

IV. CONCLUSION

In this paper we have proposed a model for detecting credit card frauds using Roughset theory for data reduction and Artificial Neural Network for learning the input data pattern. We propose to increase the accuracy futher in the subsequent phases of the Software Development Life Cycle.

In future we will then extend this work to develop a fully deployable system with a user friendly GUI for easier and faster detection of online frauds pertaining to credit cards for use by various banking and financial organisations to safeguard themselves and their customers. A comparative study of our proposed algorithm will also be made to determine its relative efficiency vis-à-vis other algorithms developed.

V. ACKNOWLEDGEMENT

We are thankful to Director, Joint Director, Principal and HoD, Dept. of Computer Engineering of AIT for allowing us to work on such an important topic. Authors would also like to thank the reviewers of this paper for their valuable comments and the conference organizers.

REFERENCES

1. TheNilson Report, Issue 540, January 1993
2. R. Brause,T. Langsdorf, M. Hepp “Neural Data Mining for Credit Card Fraud Detection”
3. Pawlak, “Rough sets,” Journal of Computer and Information Sciences
4. Predki, R.Slowinski, J.Stefanowski, R.Susmaga, Sz.Wilk: ROSE Software Implementation of the Rough Set Theory. In: L.Polkowski, A.Skowron, eds. Rough Sets and Current Trends in Computing, LectureNotes in Artificial Intelligence, vol. 1424. Springer-Verlag, Berlin (1998), 605-608.
5. B.Predki, Sz.Wilk: Rough Set Based Data Exploration Using ROSE System. In: Z.W.Ras, A.Skowron, eds.Foundations of Intelligent Systems, Lecture Notes in Artificial Intelligence, vol. 1609. Springer-Verlag, Berlin (1999), 172-180.
6. “Rough set,” http://en.wikipedia.org/wiki set.
7. Nikita gupta, Shankar lal“ rough set based behavior cluster modeling in network data”
8. http://idss.cs.put.poznan.pl
9. http://takinginitiative.net/
10. Rough Sets and Current Trends in Computing: 7th International Conference, RSCTC 2010, Warsaw, Poland, June 28-30, 2010 Proceedings
11. R. Agrawal, T. Imielinski A. Swami,” Database Mining: A Performance Perspective”. IEEE Transaction on Knowledge and Data Engineering
12. Chan, P. K., “Distributed Data Mining in Credit Card Fraud Detection,” IEEE Transaction 1094-7167/99, 1999.
13. B.R. Yang, Z.Y. Xu, W. Song, “An Efficient Algorithm for Computing Core Based on Positive Region”. ICAI’07, Las Vegas, Nevada, USA, 2007:124-132.
14. T. Y. Lin,”Rough Set Theory in Very Large Databases”
15. http://www.coursehero.com /759680-COMP4302

BIOGRAPHY

Ms. Nikita Gupta is a Lecturer in Department of Computer Engineering in Army Institute of Technology, Pune. She has National and International Conferences publications in the field of Network Security,Rough Set Theory and AI. She completed her M.Tech in Computer Science & Engineering from Defence Institute of Advanced Technology, Pune and completed B.E in Information Technology with Silver Medal from R.G.P.V Bhopal.

Archana Dhankar is a Final year student of Department of Computer Engineering in Army Institute of Technology, Pune

Karan Singh Ranawat is a Final year student of Department of Computer Engineering in Army Institute of Technology, Pune.

Shivya Nautiyal is a Final year student of Department of Computer Engineering in Army Institute of Technology, Pune.