PERFORMANCE EVALUATION OF GENETIC ALGORITHM AND COUNTER PROPAGATION NEURAL NETWORK FOR ANOMALY DETECTION IN ONLINE TRANSACTION

Olabode, Anthony Onaolapo

Department of Computer Science and Engineering,
Ladoke Akintola University of Technology, Ogbomoso, Nigeria

Corresponding Author: anthony2olabode@gmail.com

ABSTRACT: In e-commerce, credit card fraud is an evolving challenge. The increase in the number of credit card transactions provides more opportunity for fraudsters to steal credit card numbers and execute fraud. Fraud detection is a continuously evolving discipline to tackle ever changing tactics to commit fraud. Existing techniques of genetic algorithm (GA) and counter propagation neural network (CPNN) have been applied to take credit card fraud detection using different dataset and features. This paper evaluates the performance of GA and CPNN using the same dataset and features. The results show that CPNN outperform GA in terms of accuracy, sensitivity, miss rate, hit rate and prediction time.
KEYWORDS: Genetic algorithm, Counter propagation neural network, Credit cards.

1. INTRODUCTION

The increase in the popularity of e-commerce in our daily lives, credit card usages have dramatically increased over the years. Credit card frauds have also been observed to surge as the number of online transactions have increased [LHJ14]. Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. Anomalies in credit card transaction data could indicate credit card fraud or identity theft [VAV09; NLE12]. As the scale of electronic commerce transaction has grown, it has become very attractive to criminals, and the volume of fraudulent e-commerce transactions is growing rapidly. Therefore, there has been an increase in the amount of attention given to the security of the payment systems used to process online transactions [MP17]. Counter Propagation Neural Network (CPNN) is a multilayer feed forward Artificial Neural Network (ANN) based on the combination of the input, competitive, and output layers. Model of CPNN is instar-outstar. It is three-layer neural network that performs input-output data mapping, that is, producing output in the response to an input vector on the basis of Competitive Learning [V+15]. Genetic Algorithms (GA) are computer-based search techniques patterned after the genetic mechanisms of biological organisms that have adapted and flourished in changing highly competitive environment. GA is the solution for optimization of hard problems quickly, reliably and accurately [MSY11].

Fraud detection is a continuously evolving discipline to tackle ever changing tactics to commit fraud and there is need for special methods of intelligent data analysis to detect and prevent it [RA14].

This research used genetic algorithm and counter propagation neural network to detect anomalies in an online transaction. The performance of the GA and CPNN was evaluated using evaluation metrics to know which technique will perform better than the other in credit card fraud detection in any online transaction. Subsequently, the rest of this paper is organized in the following sections: some reviews on related anomaly detection, methodology of a proposed system, followed by results and discussion. The final section concludes the paper along with some recommendations for future research.

2. REVIEW OF RELATED WORKS

Credit card fraud detection has drawn lot of research interest and number of techniques, with special emphasis on neural networks; data mining and distributed data mining have been suggested [KPA14]. The detection of fraud is a complex computational task and still there was no system that surely predicts any transaction as fraudulent. The existing results predicted the likelihood of the transaction to be a fraudulent [P+14]. In 2000 [ZSA12] designed a system based on genetic programming. A Genetic algorithm is used to establish logic rules capable of classifying credit card transactions into suspicious and non-suspicious classes. The result has scalability issue. [B+00] designed the hidden Markov model (HMM) to detect the credit card fraud. A HMM is initially trained with the normal behaviour of the cardholder. If the current transaction is not accepted by the trained HMM with high probability, it is considered to be

fraudulent. [TG08] applied the neural data mining method. This system is based on customer's behaviour pattern. Deviation from the usual behaviour pattern is taken as an important task to create this system. The neural network is trained with the data and the confidence value is calculated. The credit card transaction with low confidence value is not accepted by the trained neural network and it is considered as fraudulent. If the confidence value is abnormal, then again it is checked for additional confirmation. The detection performance is based on the setting of fixed threshold, which was not efficient. [P+09] suggested a fusion approach, consisting of four components namely, rule based filter, Dempster-Shafer Adder, transaction history database and Bayesian learner. Rule based filter is used to find the suspicion level of the transaction. Dempster-Shafer Theory is used to compute the initial belief, which is based on the evidences given by the rule based filter. The transactions are classified as normal, abnormal or suspicious depending on this initial belief. Once a transaction is found to be suspicious, belief is further strengthened or weakened according to its similarity with fraudulent or genuine transactions history using Bayesian learning. Extensive simulation with stochastic models shows that fusion of different evidences has a very high positive impact on the performance of a credit card fraud detection system as compared to other methods. [A+12] investigated the effects of threshold in credit card fraud detection system. The study implemented another method of selecting threshold values (dynamic/adaptive) based on individual cardholder spending profile. The threshold value was obtained using the average of initial threshold (0.5) and ratio of acceptance probabilities of old and new transactions estimated from HMM algorithms. The performance of the system was tested with different cardholder profiles cum non-optimization and optimization of HMM parameters using some selected performance metrics. Thus the adaptive thresholds gave a better performance than fixed threshold though system reported at an outrageous prediction time. [DKG14] designed a credit card fraud detection using time series analysis. The fraud detection is done with data mining approaches. The parameters considered are transaction amount and transaction time. They used the periodic pattern in the spending behaviour of a cardholder to detect the anomalies in the transaction with respect to the analyses of the past history of transactions belonging to an individual cardholder. The approach decreased the false positive situation and hence it is ensured that few genuine transactions were not rejected.

However, the performance of the GA and CPNN have not been tested under similar condition of dataset and features in building credit card fraud detection system model. Therefore, this paper evaluates the performance of the GA and CPNN.

3. METHODOLOGY

Dataset of one thousand and three hundred (1300) transactions were acquired from thirteen (13) cardholders. Seven hundred and eighty (780) transactions were used for training while five hundred and twenty (520) transactions were used for testing. The accumulated data were prepared and presented in the form acceptable to the designed system CPNN and GA with respect to its parameters as illustrated in Fig 1.

A. CPNN-GA Algorithm

CPNN, a variant of ANN was used for classification due to its capacity for generalization because of its refined network and experimentally proven better learning rate. GA's optimization was integrated into this system in order to optimize the CPNN training parameters so that the best chromosome having optimal parameter setting can be obtained, and used by CPNN for classification purposes. The system operated in two stages; in the first phase, GA formed clusters. Clustering was done by dot product, while in second phase, the weights between the cluster units and the output units were adjusted. Minimizing error function; error function being the average error incurred when CPNN classifies large input data was considered. Initial weights were randomly selected between 0 and 1, with an assumed initial population size. Genetic algorithm performed optimization with respect to determination of the network topology, determination of the set of input attributes and determination of the neuron weights. GA tried to optimize the network topology as it evaluates the genomes in its population for candidate network topologies, and tries to optimize that specific topology for set of input features. For each of these input feature combinations, a CPNN test was constructed and trained. The construction took place for each candidate solution, given the fixed topology as determined by GA. In addition, GA optimized the weights for the constructed CPNN. The input factors, topology, and weights were encoded into a single genome for optimization. One-dimensional array of real numbers was used for encoding. The number of input factors, the number of layers, and the number of nodes in each layer determine the length of the genome. The total length of the genome L was calculated as;

$$L = n_{input} * n_1 + \sum_{i=1}^{k-1}(n_j * n_{i+1}) + n_k \qquad (1)$$

The flowchart of the design of CPNN and GA is depicted in Fig 2, where n input is the number of input attributes for the CPNN, k is the number of internal layers, and n_i is the number of nodes in layer i. The last term in the equation (1) is for the weights between the last internal layer and the output layer, which consists of a single node. The assumption is that each node in one layer is connected with every node in the subsequent layer. The encoding of the genome representing input features is done via simple binary encoding. . A zero in a specific bit in the genome means an input attribute is not chosen for the CPNN design, whereas a one in that bit means that the input attribute is chosen for the CPNN design.

The classification is done by using CPNN. The input vector was fed into the network with adjusted weights to obtain desired output vector as training mode. The cluster unit does not assume any topology, but the winning unit was allowed to learn. The steps for the classification using CPNN are as follows:

i. Normalize the input vector.
ii. The highest Kohonen layer neuron is declared the winner and its weight is adjusted to yield unity output.
iii. Then the weight vector of the winning Kohonen neuron is equsl to the input vector with the best approximation value. Kohonen neuron is unsupervised.
iv. The output of the Grossberg layer is calculated using dot product method.

$$g_i = \sum_j v_{ij} k_j = v_{ih} k_h = v_{ih} \quad (2)$$

v. Weights from non-zero kohonen neurons (non-zero Grossberg layer inputs adjusted. Weight adjustment follows the relation in equation 3.3

$$v_{ij}(n+1) = v_{ij}(n) + \beta[T_i - v_{ij}(n) k_j] \quad (3)$$

vi. The weights converged to the average value of the desired outputs, that is, best match an input-output (x-T) pair.

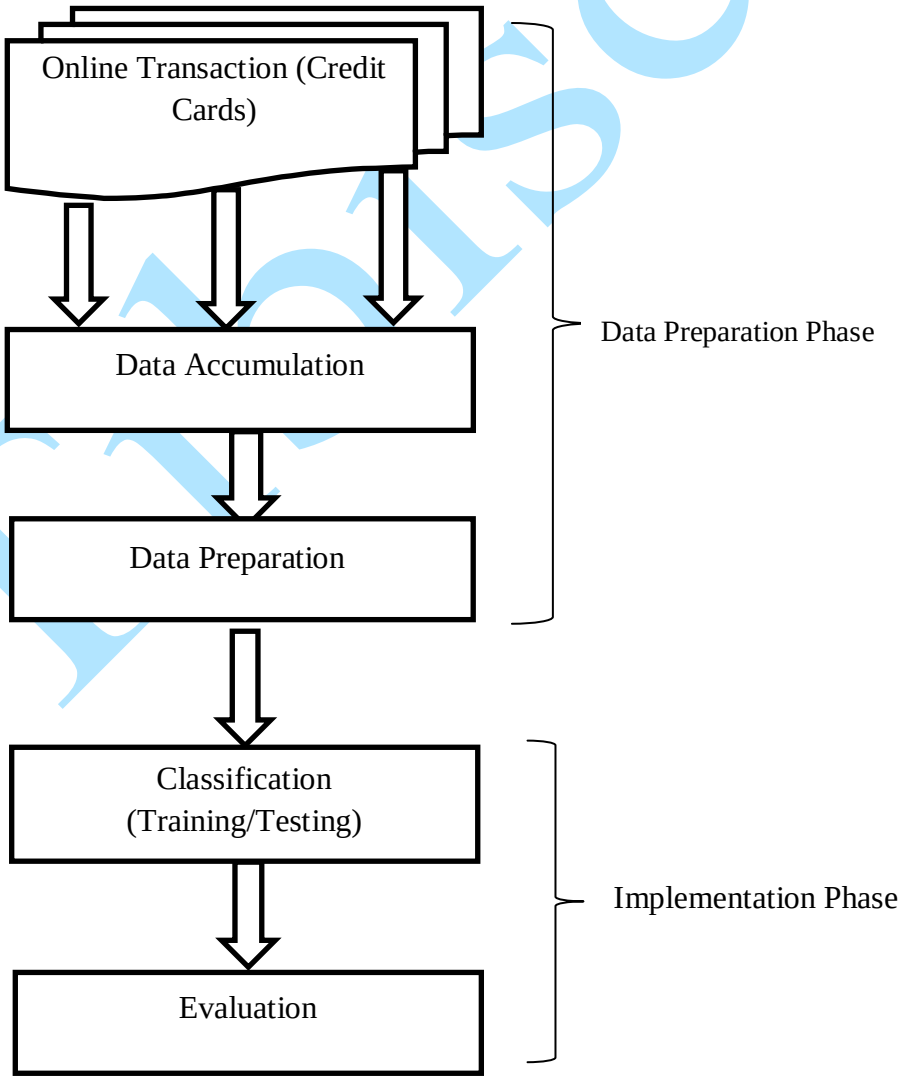

Fig 1: Architecture of the designed system

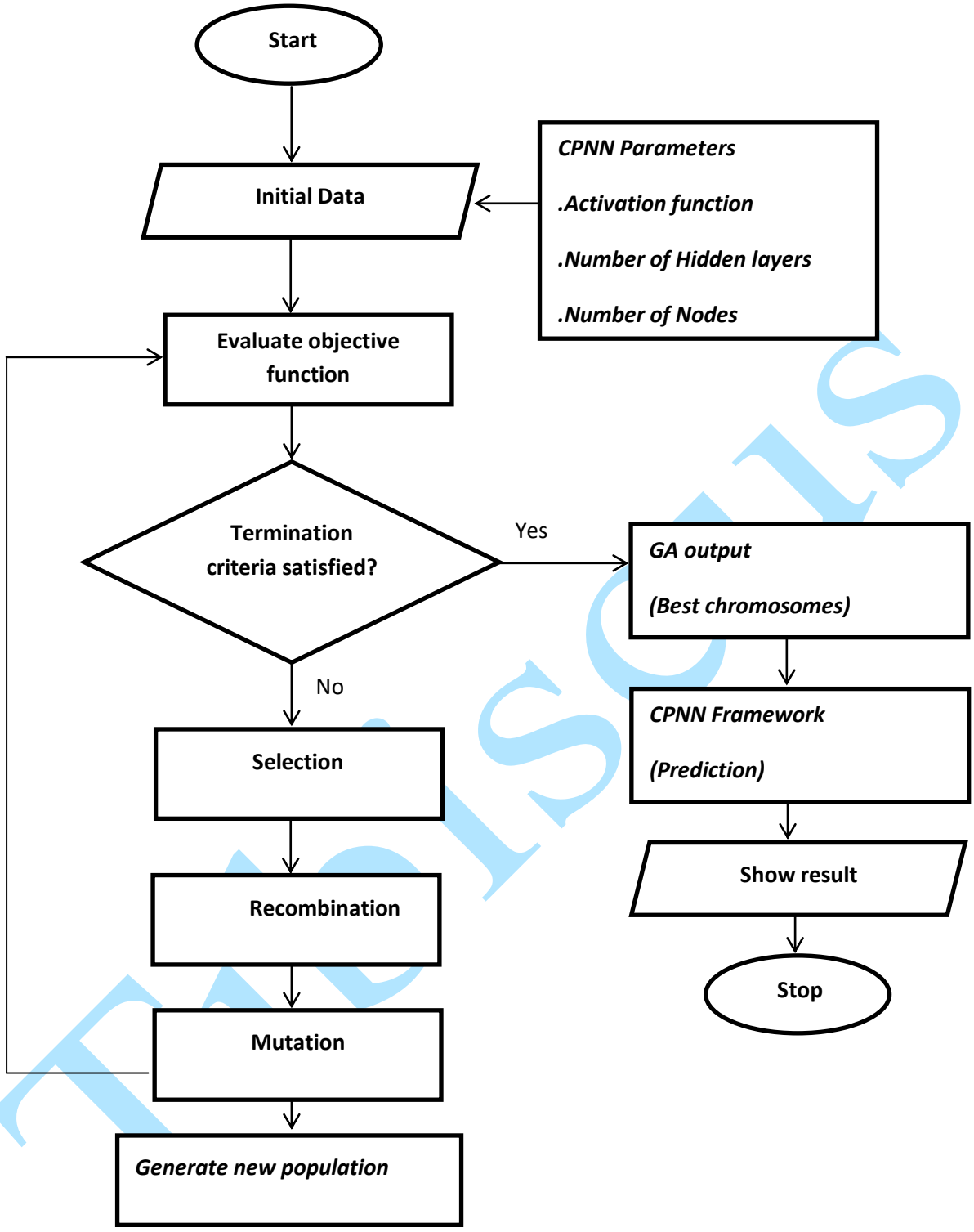

Fig 2: Flowchart showing CPNN-GA for Anomaly Detection System

4. RESULTS AND DISCUSSION

The implementation tool used was MATLAB R2012a version on Windows 7 Ultimate 32-bit operating system, Intel®Pentium® B960@2.20GHZ, 4GB Random Access Memory and 500GB hard disk drive. In a fraud detection domain, the metrics deemed best for evaluation of the designed system include False Acceptance Rate (FAR), False Rejection Rate (FRR), Prediction Accuracy, Hit rate, Miss rate, Negative Predictive Value (NPV) and prediction time.

False Acceptance Rate (FAR) denotes the rate at which the designed system incorrectly accepts a fraudulent transaction as genuine.

$$False\ Acceptance\ Rate\ (FAR) = \frac{\text{FP}}{(\text{TP+TN+FP+FN})}\ x\ 100 \quad (4)$$

False Rejection Rate (FRR) denotes the rate at which the designed system erroneously flags a genuine transaction as fraud.

$$False\ Rejection\ Rate\ (FRR) = \frac{\text{FN}}{\text{TN+FP+TP+FN}}\ x\ 100 \quad (5)$$

Prediction accuracy (ACC) represents the percentage ratio of the total number of transactions that were correctly identified.

$$Prediction\ Accuracy\ (ACC) = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

Hit rate denotes the exactness of the designed system at spotting genuine transactions in a pool of genuine transactions.

$$Hit = \frac{TP}{TP+FN} \quad (7)$$

Miss rate denotes the ratio at which the system erroneously rejects genuine transaction in a midst of genuine transactions.

$$Miss = \frac{FN}{TP+FN} \quad (8)$$

Negative predictive value (NPV) is the rate at which a fraudulent transaction is correctly identified in ratio to all negatively assigned instances.

$$Negative\ predictive\ value\ (NPV) = \frac{TN}{TN+FN} \quad (9)$$

The overall simulation results of the GA and CPNN based system were considered as illustrated in Table 1. In terms of false rejection rate, CPNN had the least FRR of 6.35% as compared with GA with FRR of 8.85%. It implies that the CPNN is tolerant in falsely accepting impostor that could have access to cardholder's account. In terms of false acceptance rate, CPNN has FAR of 6.35% as compared GA with FAR of 8.85% as illustrated in Fig 3. In terms of prediction accuracy, the CPNN has highest predictive ability to correctly identify transaction types. CPNN has 89.42% prediction accuracy of 95.58%, while GA has prediction accuracy of 84.42%. In terms of hit rate, CPNN has 93.20% while GA has hit rate of 90.08%. Also, CPNN has highest negative predictive value of 42.18%, while GA has negative predictive value of 34.87%.

Table 1: Table showing performance evaluation comparison

	FAR (%)	FRR (%)	ACC (%)	HIT RATE (%)	MISS RATE (%)	NPV (%)	PREDICTION TIME (s)	TRAINGING TIME (s)
GA	6.73	8.85	84.42	90.08	10.07	34.87	8.94	9.09
CPNN	3.85	6.35	89.42	93.20	6.88	31.54	3.42	5.67

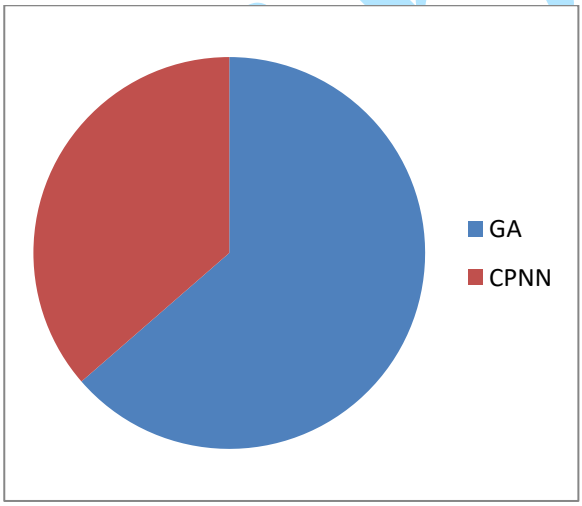

Fig 3: Pie chart showing false acceptance rate between GA and CPNN

5. CONCLUSION AND FUTURE WORK

The simulation result of CPNN and GA busing MATLAB ascertain the effectiveness of the des CPNN over GA technique. The performance evaluation result from the research showed that CPNN outperformed the GA algorithm, as it had the least false acceptance rate, least false alarm rate, highest prediction accuracy, highest hit rate, lowest miss rate and highest negative predictive value. Future work can be carried out by comparing the effect of other artificial neural network algorithms with another optimization algorithm.

REFERENCES

[A+12] **Alese B. K., Adewale O. S., Aderounmu G. A., Ismaila W. O., Omidiora E. O.** – *Investigating the Effects of Threshold in Credit Card Fraud Detection System.* International Journal of Engineering and Technology, 2 (7):1328-1332, 2012.

[B+00] **Bentley P. J., Kim J., Gil-Ho J., Choi J. U.** – *Fuzzy Darwinian Detection of Credit Card Fraud*. Proceedings of 14th Annual Fall Symposium of the Korean Information Processing Society, 1-4, 2000.

[DKG14] **Devaki R., Kathiresan V., Gunasekaran S.** – *Credit Card Fraud Detection using Time Series Analysis*. International Journal of Computer Applications (IJCA), 8 -10, 2014.

[KPA14] **Khan M. Z., Pathan J. D., Ahmed A. H. E.** – *Credit Card Fraud Detection System Using Hidden Markov Model and K-Clustering*. International Journal of Advanced Research in Computer and Communication Engineering, 3 (2):5458-5461, 2014.

[LHJ14] **Lee M., Ham S., Jiang Q.** – *E-commerce Transaction Anomaly Classification*. Statistics Department Stanford University, pp 1-5, 2014.

[MP17] **Malini N., Pushpa M.** – *Analysis on Credit Card Fraud Detection Techiniques By Data Mining and Big Data Approach*. International Journal of Research in Computer Applications And Robotics, 5 **(**5):38-45, 2017.

[MSY11] **Malhotra R., Singh N., Yaduvir S.** – *Genetic Algorithms: Concepts, Design for Optimization of Process Controllers*. Computer and Information Science, 4 (2):39-54, 2011.

[NLE12] **Neda N., Leila B., Ebrahim N.** – *Surveying Different Aspects of Anomaly Detection and Its Applications*. The Journal of Mathematics and Computer Science, 4 **(**2):129-138, 2012.

[P+09] **Panigrahi S., Kundu A., Sural S., Majumdar A. K.** – *Credit card fraud detection: A fusion approach using Dempster-Shafer theory and Bayesian learning*. Special Issue on Information Fusion in Computer Security science direct, pp: 354-363, 2009.

[P+14] **Priya B., Malvika D., Shweta P., Nivedita S., Dhake B. G.** – *Survey on Credit Card Fraud Detection Using Hidden Markov Model*. International Journal of Advanced Research in Computer and Communication Engineering, 3 (5):6445-6448, 2014.

[RA14] **Razak T. A., Ahmed G. N.** – *A Comparative Analysis on Credit Card Fraud Techniques Using Data Mining*. International Journal of Data Mining Techniques and Applications, Integrated Intelligent Research (IIR), 3 (2): 398-400, 2014.

[TG08] **Tao G., Gui-Yang L.** – *Neural Data Mining for Credit Card Fraud Detection*. International conference on Machine Learning and Cybernetics, 7, 3630-3634, 2008.

[VAV09] **Varun C., Arindam B., Vipin K.** – *Anomaly Detection: A Survey*. ACM Computing Surveys, 41 (3):1-58, 2009.

[V+15] **Vandana S., Sanjeev J., Vilas S., Dev A.** – *Fuzzy Counter Propagation Neural Network Control for a Class of Nonlinear Dynamical Systems*. Computational Intelligence and Neuroscience, pp 1-12, 2015.

[ZSA12] **Zareapoor M., Seeja. K. R., Alam M. A.** – *Analysis of Credit Card Fraud Detection Techniques: based on Certain Design Criteria*. International Journal of Computer Applications, 52 (3):35-42, 2012.