Journal of Engineering Science and Technology
Vol. 6, No. 3 (2011) 311 - 322
© School of Engineering, Taylor's University

DATA MINING APPLICATION IN CREDIT CARD FRAUD DETECTION SYSTEM

FRANCISCA NONYELUM OGWUELEKA

Department of Computer Science, University of Abuja-Nigeria
Email: nonnyraymond@yahoo.co.uk

Abstract

Data mining is popularly used to combat frauds because of its effectiveness. It is a well-defined procedure that takes data as input and produces models or patterns as output. Neural network, a data mining technique was used in this study. The design of the neural network (NN) architecture for the credit card detection system was based on unsupervised method, which was applied to the transactions data to generate four clusters of low, high, risky and high-risk clusters. The self-organizing map neural network (SOMNN) technique was used for solving the problem of carrying out optimal classification of each transaction into its associated group, since a prior output is unknown. The receiver-operating curve (ROC) for credit card fraud (CCF) detection watch detected over 95% of fraud cases without causing false alarms unlike other statistical models and the two-stage clusters. This shows that the performance of CCF detection watch is in agreement with other detection software, but performs better.

Keywords: Neural network, Data mining, SOMNN, ROC curve, CCF, Clusters.

1. Introduction

Timely information on fraudulent activities is strategic to the banking industry. Banks have many and huge databases. Valuable business information can be extracted from these data stores [1]. Credit card fraud detection is the process of identifying those transactions that are fraudulent into two classes of legitimate (genuine) and fraudulent transactions [2]. Credit card frauds can be broadly classified into three categories, that is, traditional card related frauds (application, stolen, account takeover, fake and counterfeit), merchant related frauds (merchant collusion and triangulation) and Internet frauds (site cloning, credit card generators and false merchant sites) [3].

Nomenclatures

$1 - P_D$	Probability of false negative
H_0	Normal
H_1	Fraudulent
P_D	Probability of detection
P_F	Probability of false positive
LOGIT	Logistic regression

Abbreviations

ANN	Artificial neural networks
CCF	Credit card fraud
CCFW	Credit card fraud watch
DM	Data mining
NN	Neural network
QDA	Quadratic discriminant analysis
RB	Rule Based
ROC	Receiver-operating curve
SOMNN	Self-organizing map neural network

Data mining is a process that uses a variety of data analysis tools to discover patterns and relationships in data that may be used to make a valid prediction [4]. The six basic steps of data mining process are defining the problem, preparing data, exploring data, building models, exploring and validating models, deploying and updating models. Neural network is the data mining technique used in this study and it utilized these steps for accurate and reliable result. Neural network was used because of its ability to adapt and generalize.

Artificial neural networks are massively parallel-distributed processor that has the natural propensity for storing experiential knowledge and making it available for use [5]. The processes of ANN comprise three stages such as training, testing, and deployment. There are two types of NN training methods - supervised and unsupervised methods. The unsupervised type was used in this study. Neural networks are an extension of risk scoring techniques and are based on the statistical knowledge contained in extensive databases of historical transactions, and fraudulent ones in particular. These neural network models are basically trained by using examples of both legitimate and fraudulent transactions and are able to correlate and weigh various fraud indicators (e.g., unusual transaction amount, card history, etc.) to the occurrence of fraud.

The self-organizing map is an unsupervised learning model that was introduced by Kohonen [6]. This network contains two layers of nodes - an input layer and a mapping (output) layer in the shape of a two-dimensional grid [7]. SOMNN component learning is a learning procedure that divides a set of input patterns into clusters that are inherent to the input data.

The concept of fraud detection has been founded on data mining techniques such as association rules and classification. Research on fraud detection has been focused on pattern matching in which abnormal patterns are identified from the normality. Some of these are the Detector Constructor Framework called DC-1 proposed by Fawcett and Provost [8] for telephone call fraud detection, Instruction Detection Framework and algorithms for pattern comparison proposed by Lee et al. [9]. Doronsoro et al. [10] described the operational system for fraud detection of credit card operations based on neural classifier; Aleskerov et al. [11] presented a neural network based data mining system for credit card detection and tested it on synthetically generated data; and Chan and Stolfo [12] addressed the question of non-uniform class distributions in credit card fraud detection. Haimowitz and Schwarz [13] presented a framework for credit customer optimization based on clustering and prediction. Hanagandi et al. [14] used radial basis function networks to create credit card scores from historical credit card transactions; while Stolfo et al. [15] presented a meta-learning approach in credit card fraud detection to combine results from multiple classifiers. Ibrahim [16] has introduced a hierarchical off-line anomaly network intrusion detection system based on Distributed Time-Delay Artificial Neural Network. The results indicated that dynamic neural nets (Distributed Time-Delay Artificial Neural Network) can achieve a high detection rate, where the overall accuracy classification rate average is equal to 97.24%.

This study presented an application of artificial neural networks with built-in learning capabilities, which can be used to determine fraudulent and legitimate models from the huge transaction data. A technique of self-organizing artificial neural networks and transaction rules were used to develop a decision aid known as Credit Card Fraud Watch (CCFW), which could run at the background of existing banking software to detect breaches of transaction policy, which cannot be easily detected using other methods. The credit card fraud detection algorithm was presented in a way that it can easily be implemented and tested. Real banking transaction data was used for testing to ensure that the results obtained are reliable and not biased by some unconscious assumptions.

2. Methodology

The credit card fraud detection system developed used four clusters of low, high, risky and high risk [1] as shown in Fig.1. Once the transaction is legitimate, it was processed but if any transaction falls into any of these clusters; it was labeled as suspicious/fraudulent. The alert goes off and the reason is given. The fraudulent transaction will not be processed but will be committed to the database.

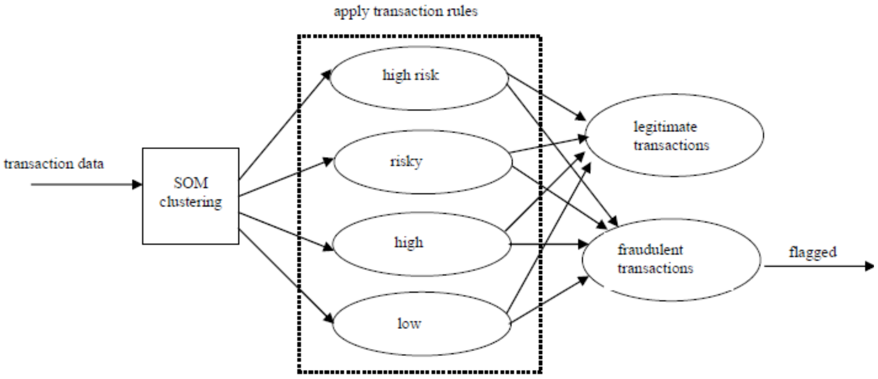

Fig. 1. A Four-Stage Credit Card Fraud Detection Model [17].

The approach involves the following steps, which is also illustrated in Fig. 2. The steps are select an appropriate algorithm; implement the algorithm in software; test the algorithm with known data set; evaluate and refine the algorithm as it is being tested with other known data sets; and show the results.

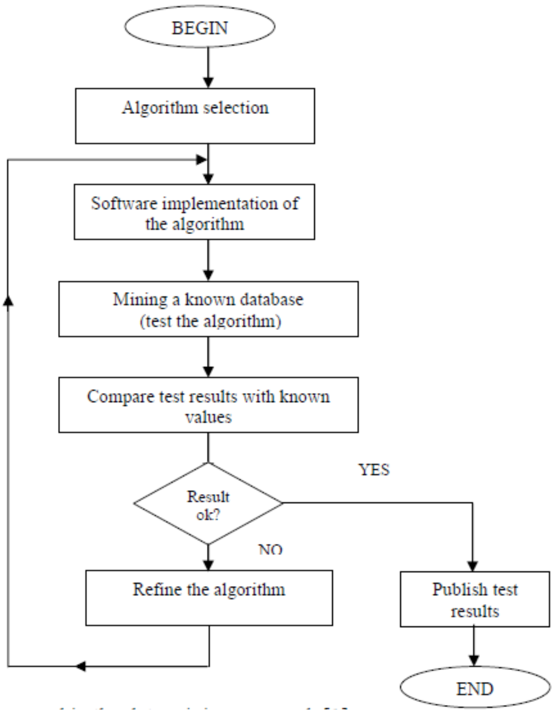
BEGIN
Algorithm selection
Software implementation of the algorithm
Mining a known database (test the algorithm)
Compare test results with known values
Result ok?
YES
NO
Refine the algorithm
Publish test results
END

Fig. 2. Steps Used in the Data Mining Approach [1].

The major functionalities of the artificial neural network (ANN) based credit card detection system designed are as follows: to facilitate real-time transaction entry, and react to a suspicious transaction that may lead to fraud. The design of the architecture is based on a neural network unsupervised method, which was applied to the transactions data to generate four clusters: the low, high, risky and high-risk clusters [1]. The system runs secretly beneath the banking software within banks offering credit card services where fraudulent transactions are observed. Business rules relevant to the enlisted CCF types are further applied to the four clusters to detect transactions that deviate from the norm. Deviation from the usual pattern of an entity implies the existence of a fraud. Each transaction entering the database such as withdrawal, deposit, and any card transaction is treated as a signature, suspected and prone for verification. The similarity between a customer’s present transaction and a known fraud scenario indicates the same fraud may occur again. Suspected transactions are flagged within seconds for further investigations and subsequent decision-making. Visualization is provided using appropriate graphical user interface (GUI).

The architecture of the artificial neural network based credit card fraud watch is shown in Fig. 3 [17].

The implemented architecture consists of two subsystems: database interface and credit card fraud (CCF) detection engine. The database interface subsystem is the entry point through which the transactions are read into the system. It is the system's interface with the banking software. Visual Basic.Net was used for the design of CCF detection, that is, as a front-end while Microsoft Access was used for the design of training and test database, as back-end. In the CCF detection subsystem, each transaction entering into the system was passed to the host server where the corresponding transaction profile is further checked using neural networks and transactions business rules.

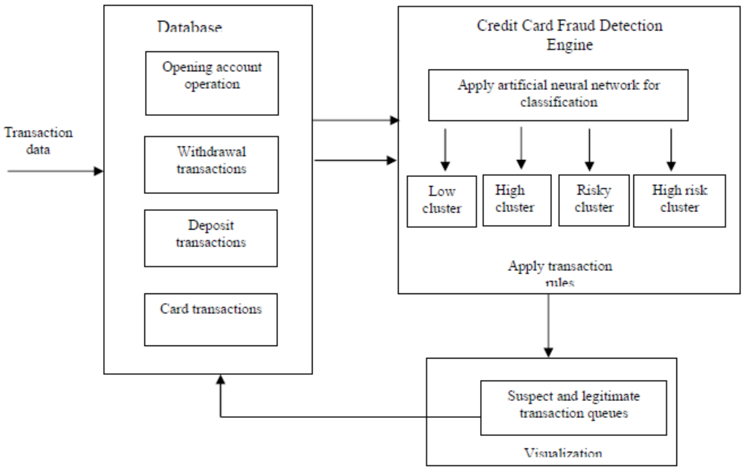

Fig. 3. Architecture of the Credit Card Fraud (CCF) Watch [17].

3. Results and Discussion

In this study, the fraud detection system watch consists of two units namely, the withdrawal and deposit unit. Each of the two units is in turn made up of the following subunits: the database interface, the neural network classification, and the visualization. The database interface subunit was tested to ensure that the necessary transaction data was imported and used. The neural network classification was done using the self-organizing neural network algorithm where the available data set was randomly partitioned into a training set and a test set, and the training set was further partitioned into subsets: used for elimination of the model (i.e., training the algorithm) and a subset used for evaluation of the performance of the model (i.e., validation). The GUI visualization subunit is also tested to facilitate event driven. If any of the above test fails, the subunit was redesigned or the program statements rewritten, followed by a retesting, until all the subunits pass the test.

Test data was designed and run on the system with the tested program. The result of the process was compared with a manually prepared result to determine the efficiency and effectiveness of the new system. The test data for two credit card banking transactions under study, which is withdrawal and deposit, was presented. The software-programs were experimented module by module to give an expected result. For actual and expected results; a 0 in the corresponding cell refers to legitimate transaction, a 1 refers to suspected transaction, while a 2 refers to fraudulent transaction.

The principles underlying detection software are grounded in classical statistical decision theory. There are two sources that generate inputs to the detection software: normal (H_0) and fraudulent (H_1). The normal source generates legal or authorized transactions. The fraudulent source generates illegal or fraudulent transactions. In a typical real-life detection scenario, a large percentage of transactions are legal. The skewed nature of the frequency distribution makes detection of illegal transactions difficult. The detection software observes the transaction but does not know whether it came from a normal or fraudulent source. The goal of detection software is to classify each transaction as legal or fraudulent. The types of errors that can occur in this classification are classification of a fraudulent transaction as legal (false negative); and classification of a legal transaction as fraudulent (false positive).

Probability of detection = $P_D = P_r$ (classify into $H_1 \mid H_1$ is true) or

Probability of false negative = $1 - P_D$

Probability of false positive = $P_F = P_r$ (classify into $H_1 \mid H_0$ is true)

Let the numerical values for the normal and fraudulent transactions follow exponential distributions with parameters λ_N and λ_F, $\lambda_N > \lambda_F$ respectively.

The probability of detection P_D and probability of false positive P_F as

$$P_D = \int_t^\infty \lambda_F e^{-\lambda_F x} dx = e^{-\lambda_F t}$$

$$P_F = \int_t^\infty \lambda_N e^{-\lambda_N x} dx = e^{-\lambda_N t}$$

Thus P_D can be expressed as a function of P_F as

$P_D = P_F^r$ where $r = \lambda_F/\lambda_N$ is between 0 and 1.

Trees [18] stated that the quality profile of most detection software is characterized by a curve that relates its P_D and P_F known as the receiver operating characteristic curve (ROC). ROC is a function that summarizes the possible performances of a detector. It visualizes the trade-off between false alarm rates and detection rates, thus facilitating the choice of a decision functions. This was therefore used in the performance analysis of CCF and other detection watch systems.

The effectiveness of this detection software is measured in terms of the classification errors, which consist of system detection rate and false alarm rate. The data used in the application were collected from a Nigerian bank, which consist of transaction data made per day during the observed period, that is, in a month. The collection was done according to the two types of bank operations investigated. The aim is to identify frauds within each of the categories by identifying flagged transactions using the CCF detection watch approach. Details of the test datasets are listed in Table 1.

Table 1. Summary of the Two Data Subsets Used to Test each of the Operations of CCF Detection Model.

Operation	Transaction	Fraudulent	Proportion of fraudulent
Withdrawal	10,650	5	0.47%
Deposit	8,102	2	0.24%
Total	18,752	7	0.37%

The performance analyses of the respective detection algorithms are carried out using MATLAB software package and the results compared with the collected data are as shown in Figs. 4 and 5.

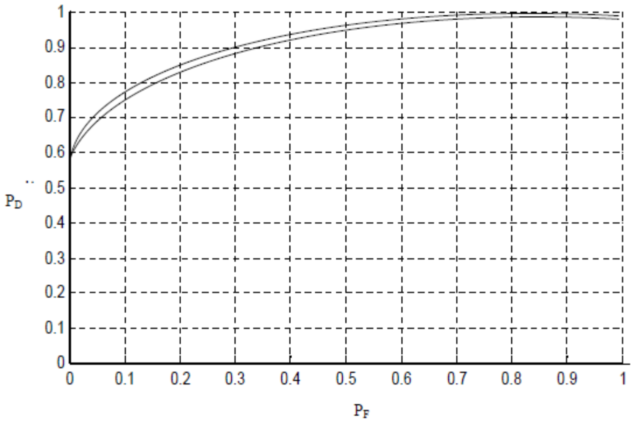

Fig. 4. ROC for Withdrawal Fraud Detection.

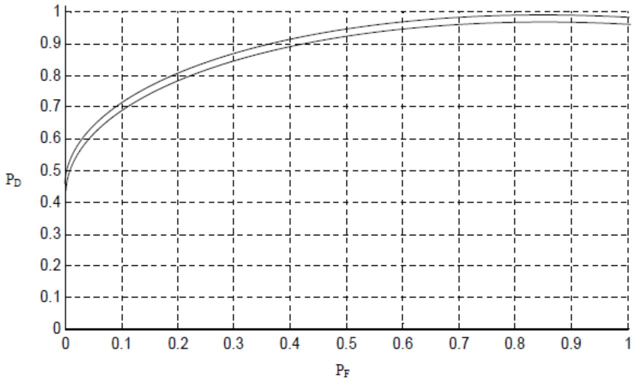

Fig. 5. ROC for Deposit Fraud Detection.

Figures 4 and 5 show that the model results compared satisfactorily well with the collected data results for the two transactions examined in this paper.

In the comparison of CCF detection watch performance with two other fraud detection models using the ROC curve, it was noted that the results from CCF detection using neural network was accurate and reliable. The two different commercial products, quadratic discriminates analysis (QDA) and logistic regression (LOGIT) were selected to test the feasibility of using neural network tools for the purpose of CCF detection watch. The performance analysis of the CCF detection watch model (deposit transaction) is compared with these two commercial packages and the result is shown in Fig. 5.

From Fig. 6, the ROC curve for CCF detection watch detects over 95% of fraud cases without causing false alarms. This is followed by the ROC curve for logistic regression with 75% detection with no false alarms, the quadratic discriminant analysis with 60% detection. This shows that the performance of CCF detection watch is in agreement with other detection software, but performs better.

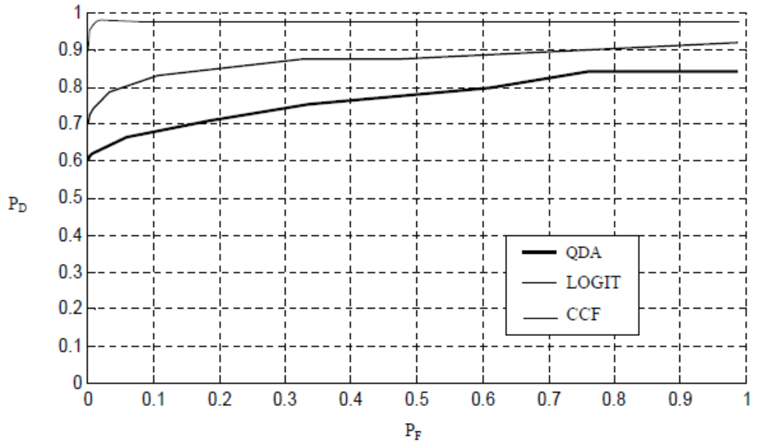

Fig. 6. Comparison of CCF Detection Watch with other Fraud Detection System ROC for Deposit Fraud Detection.

Using two software products enabled this work to illustrate different user interfaces available, alternative to neural networks models, design for decision-making, and the performance metrics of different models. Comparison of performance of neural network model with traditional statistical models increased the confidence in the ability of CCF detection watch in successful modelling of credit card frauds in the banking industry.

In the comparison of the performance evaluation of the two- stage cluster model and the four-stage cluster model, the two-stage credit card fraud detection model works as a binary classifier that has two choices, “fraud” or “legal”. It works by producing a score. The score is a measure of the confidence of the classifier that a particular transaction is legal or fraud. To decide which it is, the score is compared to a threshold then it is deemed to be legal. If it is greater, then it is fraud. This type of model consists of a component and five user interfaces and tends to classify most legal transactions as fraudulent. The four-stage credit card fraud detection model designed consists of two components with seven user interfaces. The two components are Artificial Neural Network (ANN) component and the Rule Based (RB) component. It was developed using four classes of cluster - low, high, risky and high risk.

In the SOMNN engine, the data set description, number of data points in the dataset, and the number of clusters are entered. The data point’s entry was created and filled. The entries are sorted in ascending order under the cluster list. When the train/generate clusters are selected, the cluster label becomes ready for filling depending on the number of clusters entered and then stored on the database. The database was stored in Microsoft Access table and was also used to determine when a card transaction was to be processed, blocked, unblocked, or the alert set off. After each transaction, the data point entry and clusters made are processed by the SOMNN engine and sent into the database. This helps the detection engine to know when any data entry is legitimate or fraudulent, and the reason is given immediately after the alert. The transaction will not be processed, but will be stored in the database. The SOMNN setup report displays the number of clusters, the names of the clusters and the list of cluster transactions.

The database was meant to run at the background of the existing banking software and be getting its data from real-time banking transaction, checking whether the transaction is legitimate (and so will be processed) or fraudulent (transaction will not be processed and the alert will be let off with reasons displayed). The detection software is interfaced with the source data. Microsoft Access was used to create and manage the database. Each of the data tables created in this study such as transaction tables, account open tables, cluster tables, legitimate and fraudulent tables, is stored in a separate file. In the cluster name table, four cluster names were used. Since the binary classifiers used in most fraud detection software sometimes take legitimate transactions as fraudulent, the decision to use four cluster names is to eliminate the problem. The cluster names are low, high, risky and high risk. The cluster tables to be used can be from four to any number but not more than ten.

The input database interface provides a graphical user-friendly interface with possibility of importing necessary bank transaction data for credit card fraud from the database. In the output database interface, when a transaction data is checked, the software provides two ways of presenting detection results namely: with all details and without details. Detection with all details provides step-by-step visualization of the checked transactions such as the cluster it belongs, the tables either suspect or legitimate. Detection without details option does not show certain details, e.g., clusters. It merely presents the result of the detection.

After each transaction, the data point entry and clusters made are processed by the SOMNN engine and sent into the database. This helps the detection engine to know when any data entry is legitimate or fraudulent, and the reason is given immediately after the alert. The transaction will not be processed, but will be committed to the database. This is illustrated in the SOMNN engine setup form [17] in Fig. 7.

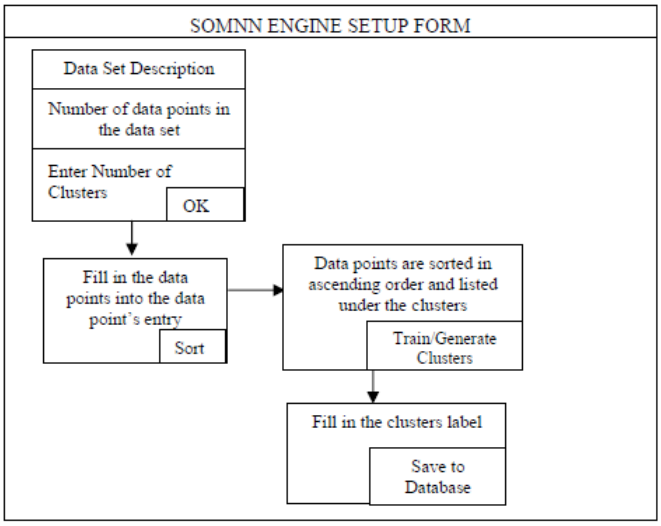
SOMNN ENGINE SETUP FORM
Data Set Description
Number of data points in the data set
Enter Number of Clusters
OK
Fill in the data points into the data point's entry
Sort
Data points are sorted in ascending order and listed under the clusters
Train/Generate Clusters
Fill in the clusters label
Save to Database

Fig. 7. SOMNN Engine Setup Form.

The artificial neural network (ANN) module is a VB standard module that takes care of the artificial neural network algorithm for classifying data based on their attributes. Given a dataset containing any number of data points, the algorithm separates the dataset into groups of similar data points. It takes as input the number of groups or clusters to separate a dataset into and a dataset containing the items to be clustered or separated into groups. Each data point in the dataset is assigned to the cluster and stored in the database.

The card fraud alert was designed in a way that alerts are generated when the selected risk exceeds a threshold. By adjusting the threshold, the number of alerts was controlled. As the threshold is lowered, the number of alerts increases. Consideration was made so as to remove any resulting deviation from accurate and reliable result because at the extreme, if the threshold is reduced to zero then every transaction will be alerted and although a 100% detection rate could be claimed there will also be an overwhelming number of false positives.

The designed system supports several different types of alert and each has different priority levels of the four clusters used but two of the four clusters were adjusted to one. The levels used are low, medium (high and risky) and high-risk. By setting the thresholds for medium level alerts to the optimal value, it can then set thresholds for the high-risk level alerts to catch high-risk incidents and the low-level alerts to sweep up the low-risk ones. By differentiating alerts levels in this way, users can target resources and opt to be informed by email or SMS of particular alert types and levels. The available alerts introduced are system, pattern and customer alerts.

The risk engine generates system alerts and the computation of the risk measures was based on the systems built in algorithms, where each measure generates alerts so that the bank can see immediately when the expected loss on an account exceeds a threshold. With the pattern alerts, banks can set up patterns that experience has taught them are good detectors of fraud or to capture short term situations such as transactions from a particular merchant. Patterns allow banks to use their specialist knowledge of their particular client-base, and further, variables derived from the raw transaction data such as rate of spend are also exposed as it was included in patterns. A separate risk threshold, that is, the customer risk threshold was included so that customers can be sent an SMS message alerting them that their card has been used and detailing certain aspects of the transaction such as amount, time, merchant, etc. The customer only needs to reply to this message if they wish to confirm that the transaction is fraudulent. A window in which a customer must respond is configurable. After this time the transaction is assumed to have been confirmed. The advantages to this designed system mechanism are for better detection, immediate feedback and improved system performance.

4. Conclusions

The study resulted in a model, which was used to detect abrupt changes in established patterns and recognize typical usage patterns of fraud. The CCF detection system was designed to run at the background of existing banking software and attempt to discover illegitimate transactions entering on real-time basis. This proved to be very effective and efficient method of discovering fraudulent transactions.

The principles of neural networking are motivated by the functions of the brain especially pattern recognition and associative memory. The neural network recognizes similar patterns, predicting future values or events based upon the associative memory of the patterns it has learned. The advantages neural networks offer over other techniques is that these models are able to learn from the past and thus, improve results as time passes. They can also extract rules and predict future activity based on the current situation. By employing neural networks effectively, banks can detect fraudulent use of a card, faster and more efficiently.

The data used was a mixture of normal (legitimate) and fraudulent with an unknown mixing mechanism. CCF detection system detected most of the fraudulent transactions, but more importantly, the probability of detection, probability of false positive was below 2 or 3 percent. The method was noted to be effective in detecting system with data from real banking environment.

The research presented the following contributions to knowledge – the design of neural networks credit card fraud (CCF) detection model using four clusters; the design of the CCF detection watch architecture and the design of the CCF detection alert. An approach was used to developed the credit card fraud detection system that utilizes both conventional data mining and neural network approaches to achieve a synergy that better handles the Nigerian credit card fraud situation using the four clusters instead of two-stage model/clusters normally used in fraud detection algorithms. This reduced the classification of legitimate transactions as fraudulent, ensured accurate and reliable result. The study reinforces the validity and efficiency of ANNs as a research tool and laid a solid groundwork for intelligent detection methodologies to be used in an operational fraud detection system.

References

1. Ogwueleka, F. N. (2008). *Credit card fraud detection using data mining techniques*. Ph.D. Dissertation. Department of Computer Science. Nnamdi Azikiwe University, Awka, Nigeria.
2. Maes, S.; Tuyls, K.; Vanschoenwinkel, B.; and Manderick, B. (2002). Credit card fraud detection using Bayesian and neural networks. *Proceeding International NAISO Congress on Neuro Fuzzy Technologies.*
3. Bhatla, T.P.; Prabhu, V.; and Dua, A. (2003). *Understanding credit card frauds*. Crads Business Review# 2003-1, Tata Consultancy Services.
4. Edelstien, H.A. (1999). *Introduction to data mining and knowledge discovery.* (2nd Ed.), Two Crows Corporation.
5. Haykin, S. (1999). *Neural networks: A comprehensive foundation.* Prentice Hall. New York.
6. Kohonen, T. (1997). The self-organizing map. *Proceedings of the IEEE* 78(9), 1464-1480.
7. Hiotis, A. (1993). Inside a self-organizing map. *AI Expert*, 8(4), 38-43.

8. Fawcett, T; and Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 1(3), 291-316.

9. Lee, W.; Stolfo, S.J.; and Mok, K.W. (1999). Algorithms for mining system audit data. In Lin, T.Y.; Yao, Y.Y.; and Zadeh, L.A. (Eds.): Data mining, rough sets and granular computing, 95, *Studies in Fuzziness and Soft Computing*, 166-189.

10. Dorronsoro, J.R.; Ginel, F.; Sanchez, C.; and Cruz, C. (1997). Neural Fraud Detection in Credit Card Operations. *IEEE Transactions on Neural Networks* 8(4), 827-834.

11. Aleskerov, E.; Freisleben, B.; and Rao, B. (1997). CARDWATCH: A neural network-based database mining system for credit card fraud detection. *Proceeding of the IEEE/IAFE on Computational Intelligence for Financial Engineering*, 220-226.

12. Chan, P.K.; and Stolfo, S.J. (1998). Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, 164-168.

13. Haimowitz, I.J.; and Schwarz, H. (1997). Clustering and prediction for credit line optimization. *Proceedings of AAAI-97 Workshop on AI Approaches to Fraud Detection and Risk Management*, 29-33.

14. Hanagandi, V.; Dhar, A.; and Buescher, K. (1996). Density-based clustering and radial basis function modeling to generate credit card fraud scores. *Proceedings of the IEEE/IAFE 1996 Conference on Computational Intelligence for Financial Engineering (CIFEr)*, 247-251.

15. Stolfo, S.J.; Fan D.W.; Lee, W.; Prodromidis, A.; and Chan, P.K. (1997). Credit card fraud detection using meta-learning: Issues and initial results. *Proceedings of AAAI-97 Workshop on AI Approaches to Fraud Detection and Risk Management*, 83-90.

16. Ibrahim, L.M. (2010). Anomaly network intrusion detection system based on distributed time-delay neural network (DTDNN). *Journal of Engineering Science and Technology* (*JESTEC*), 5(4), 457-471.

17. Ogwueleka, F.N.; and Inyiama H.C. (2009). Credit card fraud detection using artificial neural networks with a rule-based component. *The IUP Journal of Science and Technology*, 5(1), 40-47.

18. Trees, H.L.V. (2001). *Detection, Estimation and Modulation Theory-Part I.* John Wiley, New York.