

A Novel Machine Learning Approach to Credit Card Fraud Detection

L.U. Oghenekaro
Department of Computer Science,
University of Port Harcourt
Choba, Rivers State, Nigeria.

C. Ugwu
Department of Computer Science
University of Port Harcourt,
Choba, Rivers State, Nigeria

ABSTRACT

The use of credit cards is of paramount importance in improving the economic strength of any nation, however, fraudulent activities associated with it is of great concern. When fraud occurs on credit cards, the negative impact is huge as the financial loss experienced cuts across all the parties involved. This paper provides a proactive measure at detecting fraudulent activities regarding the credit card. A novel approach in machine learning known as the cortical learning algorithm was adopted to build the credit card fraud detection model. The algorithm worked on the credit card data obtained from the UCI Repository, it converted the highly populated data to a sparse representation, and then used its learning columns to learn spatial and temporal patterns. The object oriented analysis and design methodology was used in the design of the system which was implemented with JAVA programming language. The simulation was carried out with Matlab. The resulting model performed online learning and recorded higher percentage accuracy of 91% and beyond in detecting fraudulent transactions as compared to the Neural Network model that recorded 89.6%, hence, cases of misclassification was reduced to the barest minimum and efficiency of fraud detection was increased.

Keywords

Cortical Learning Algorithm, Merchant, Sparse Distributed Representation, Classification, True Positive and False Positive.

1. INTRODUCTION

The economic growth of any nation lies greatly in the need of running a cashless economy, the benefits of running a cashless economy cuts across board both for the government and the people, with a cashless economy, cost of manufacturing and transporting currency notes is reduced, economic growth is increased as payments are made efficiently, corruption is curbed, cash handling charges are eliminated, risk of robbery is reduced and economic activities will increase. With a population of over 180 million, the country can barely move forward without effective consumer lending. The use of credit card by an average citizen is a major move to boost a cashless economy that is highly desired [12]. While financial and government institutions are working hand

in hand to promote and encourage the use of credit card by setting up the right platforms and creating awareness, it is only wise for us as researchers and scientists to swing into action in ensuring that applications and checks are put in place in ensuring the safety of the use of the credit card. The credit card has been in use in many developed countries but was only introduced into the Nigerian economy in the year 2006 [3], before then, the use of credit card was perceived as impossible and unworkable in the Nigeria's business environment but today, it is a reality as all banks in the

country are driving its use. Recently, the Central bank of Nigeria introduced the nation's first non-bank credit card [12] and this goes to show that the nation is improving on the use on credit card, hence the need for this type of research.

Credit card fraud (CCF) is a recurrent problem all over the world, some countries, despite having a high use of credit card, have a low percentage of credit card fraud recorded while other countries with an average use of credit card have recorded a very high percentage [4], and the difference between these countries lies in the measures they have put in place to checkmate fraud, hence it is expected that proactive measures are put in place to curb CCF before the credit card becomes widely used in our nation. CCF can be defined as an unauthorized account activity when an individual uses the credit card of another individual without his/her permission or the permission of the card issuer [2]. It is also a form of theft and it poses a great threat to business establishments of today.

The cortical learning algorithm (CLA) is the learning algorithm of the hierarchical temporary memory (HTM) and it is inspired by the neocortex of the brain. The HTM theory has created a new approach of machine learning for prediction and anomaly detection [11]. The strength of the HTM technology lies in its principles of hierarchical organization, learning time-based transitions and sparse distributed representations. It was first introduced by Jeff Hawkins and Sandra Blakeslee in 2005 and has since been very successful in a variety of time-series prediction and anomaly detection problems.

Terms such as cell states, synapses, potential dendrites, columns and dendrite segments were used throughout the life of the algorithm. More light is thrown on them to facilitate the understanding of the algorithm.

1. *Cell States:* The HTM cells have three output states which are; the active state from feed-forward inputs, the active state from lateral input (representing a prediction) and the inactive state.
2. *Dendrite Segments:* Each HTM cell has one proximal dendrite segment and dozens of distal dendrite segments. The proximal dendrite segment receives feed-forward inputs while the distal dendrite segments receive lateral inputs from nearby cells. The spatial pooler function operates on the proximal dendrite segment while the temporal pooler function operates on distal dendrite segments.
3. *Synapses:* HTM synapses have binary weights and to model the forming and un-forming of these synapses, the concept of potential synapses and permanence comes in. Potential synapses represent all the axons that pass close enough to a dendrite segment that they could potentially form a synapse while permanence is a scalar value ranging from 0.0 to 1.0 assigned to each potential synapse. The

permanence of a synapse refers to a range of connectedness between an axon and a dendrite. During learning the synapse's permanence is being increased and decreased. The synapse is connected with the weight of 1 if its permanence is above a threshold, otherwise, it is disconnected with the weight of 0.

2. RELATED WORKS

The literatures have proposed different techniques ranging from artificial intelligence techniques to statistical data analysis techniques. Some of these related literatures were reviewed to give a guide on the current situation of CCF detection. Ghosh and Reilly (1994), proposed a fraud detection system for credit card using a three-layer approach of feed forward neural network. The system was trained on a sample of labelled data that included all the activities on the account over a period of two months. Various kinds of fraud data were used to train the neural network. After the training, it recorded significant success in detecting fraudulent transactions than the earlier rule-based fraud detection systems used. However, the system required long training time [5]. Aleskerov et al. (1997), proposed a database mining technique to design a fraud detection system for credit cards using approach called CARDWATCH, the system was based on a neural network learning section and provided interface to a variety of commercial databases. The test results obtained from the detection model indicated high fraud detection rate. CARDWATCH exhibited high processing speed and great accuracy in fraud detection but has a shortcoming of requiring one network to one customer.

Maes et al. (2002) applied two machine learning techniques; artificial neural networks (ANN) and Bayesian belief networks (BNN) in detecting credit card fraud. The work looked at the features of a new transaction to classify if the transaction is genuine or fraudulent. The STAGE approach (which is an instance of global optimization) was used in learning the topology of the BNN. Primary data was obtained by Serge Waterschoot at Europay International (EPI). The Receiver Operating Curve (ROC) was used to measure the performance of both the ANN and BNN systems. The ROC combined the information of true positive and false positive in a single graph, the true positive is plotted on the y-axis and false positive on the x-axis. The resulted graph showed that accuracy of predictions made depends on how much the system had learned. The ROC graph showed that BNN yields better results than ANN when applied to CCF detection, and the learning time in BNN is very short as compared to ANN, however, the CCF detection process is faster with ANN.

Neda and Akbari (2014), put forward a new model for CCF detection using Artificial Immune Systems (AIS) named AIS-based Fraud Detection Model (AFDM) and the model was implemented on a cloud-base file system (cloud computing). The work extended the previous work done by Watkins et. al, 2004 by introducing distributed storage to the artificial immune recognition system, thereby allowing training phase to be done in a parallel manner hence reducing training time.

An increase of fraud detection by 25%, reduced cost by 85% and decrease system response time up to 40% as compared to the base algorithm AIRS (Artificial Immune Recognition System) was recorded.

Khan et al. (2014), extended the research [6] on CCF detection with K-Clustering and Hidden Markov Model (HMM), clusters generated by K-means clustering algorithm were used, the system architecture had training stage and the detection stage. After the development of the model, the spending profile of every incoming online transaction was matched to the existing database to determine if the transaction was genuine or not, if it is, transaction was carried out successfully, else, the HMM traced the IP address of the organization from where the unauthorized user carried out the transaction and sent notification to authorized user's phone number and also raises alarm to the admin system [8]. However, this appears to have a reduced false positive rate but it did not take the aspect of time into consideration.

Having reviewed previous machine learning approach on credit card fraud, this work seeks to address the lingering problem of misclassification and also to improve on efficiency. This paper will use a novel approach of the hierarchical temporal memory cortical learning algorithm

to develop a more efficient model, taking into consideration more attributes such as the credit history,

employment status, owned properties, job status, and number of existing credits at the bank etc., rather than the spending profile that was mainly considered in most previous works.

3. MATERIAL AND METHODS

The methodology employed in this paper is the object oriented methodology. This software development methodology uses a recursive object-oriented development method for which the entire system is broken down into subsystems and modules. Due to the high sensitive nature of the credit card data, banks and credit card companies in general do not easily disclose this information [Bahnsen, 2013]. This reason serves as one of the limitations for researches in this area. The dataset used in this system was obtained from the UCI repository. The dataset consists of a thousand instances and each instance has twenty attributes that gave detailed information of the transaction, but these attributes were coded in numerical and symbolic form for confidential reasons. The HTM cortical learning algorithm carried out three major steps in analyzing and making prediction from the streaming data inputs that comes into every region of an HTM [11]. These steps are outlined below:

- A. Forming a sparse distributed representation (SDR) of the input.
- B. Forming a representation of the input in the context of previous inputs:
- C. Forming a prediction based on the input in the context of previous inputs.

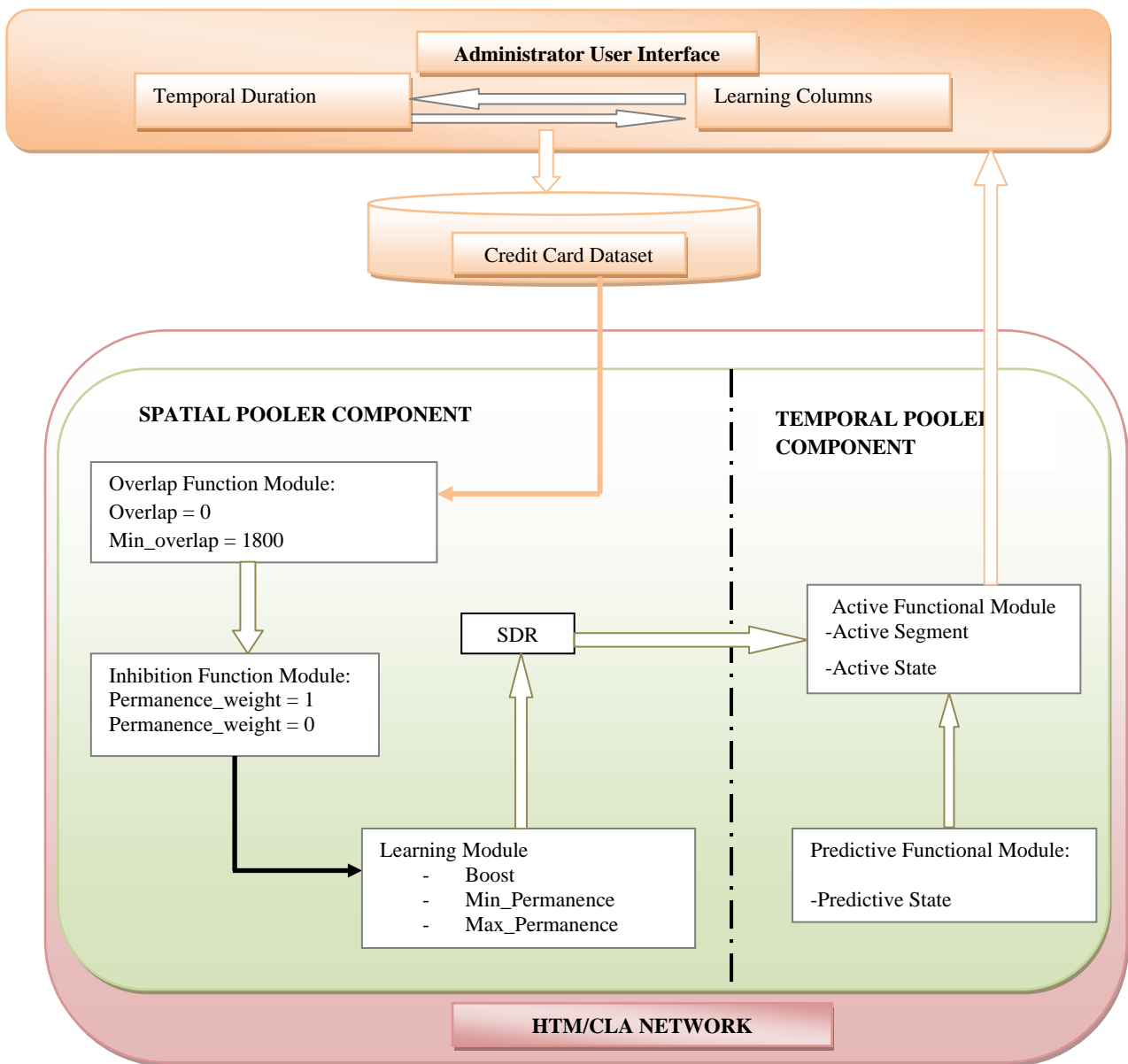


Figure 1: System Architecture

From the architecture of the system as depicted in figure 1, the CLA is divided into two major modules; the Spatial Pooler (SP) and the Temporal Pooler (TP).

As streaming data from the credit card dataset comes into an HTM region, the overlap function computes the number of connected synapses with active inputs; it does this by ensuring that each column in a region is connected to a unique subset of the input bits, hence, different input patterns results in different levels of activation of the columns. The outcome of the overlap function goes to the inhibition function that make columns with the strongest activation deactivate the columns with weaker activations and the product of this phase goes into the learning module where permanence values for all synapses are being updated. For all active columns, their synapses are increased, otherwise, they are decreased. The output of the learning module is the sparse distributed representation which goes into the temporal pooler block. The first module of the temporal pooler, active functional module, calculates the active state of each cell in the winning column.

In this function, cells that are active activates the cells that in their predictive state else, all the cells in the column get activated showing that a new input have come in. However, learning takes place in this function by the adjustment of the synapses' permanence values, if the synapse is connected to active cells; its permanence is increased, else decreased. The output of the cortical learning algorithm is the set of active columns.

The mathematical model that illustrated the computations carried out in the HTM system as used in this thesis can be described using the following notations (Bryne, 2015):

- n^{FF} represents the width of the input.
- x^{FF} represents the input vector.
- n_s represents the number of synapses on the dendrite, where each synapse act as a binary gate for a single input bit.
- i represents a single synapse.
- j represents a neuron

p_i represents the permanence of each synapse
 θ_i represents the threshold which is often a global constant of 0.2
 x_j represents the input vector of each dendrite.
 N represents the number of neuron in each layer.
 s represents its sparsity
 Y_{SDR} represents the estimate of input by an SDR of neurons.

The first step which is the processing of the feedforward sensory inputs on the proximal dendrites can be modeled as a bit vector;

$$x_{FF} \in \{0,1\}n^{FF} \dots\dots\dots (1)$$

The binary connection vector for the dendrites is thus:

$$c_i = \frac{1+sgn(p_i - \theta_i)}{2} \dots\dots\dots (2)$$

The transformation of an input vector into a sparse distributed representation is thus:

$$\hat{x}_{SDR} = \sum_{j \in Y_{SDR}} \hat{x}_j \dots\dots\dots (3)$$

The Hebbian update model is used to adjust the parameters of the model during learning:

$$p_i^{(t+1)} = \begin{cases} (1 + \delta_{inc})p_i^{(t)} & \text{if } j \in Y_{SDR}, (x_j)_i = 1 \text{ and } p_i^{(t)} \geq \theta_i \\ (1 - \delta_{dec})p_i^{(t)} & \text{if } j \in Y_{SDR} \text{ and } ((x_j)_i = 0 \text{ or } p_i^{(t)} < \theta_i) \\ p_i^{(t)} & \text{otherwise} \end{cases} \dots\dots\dots (4)$$

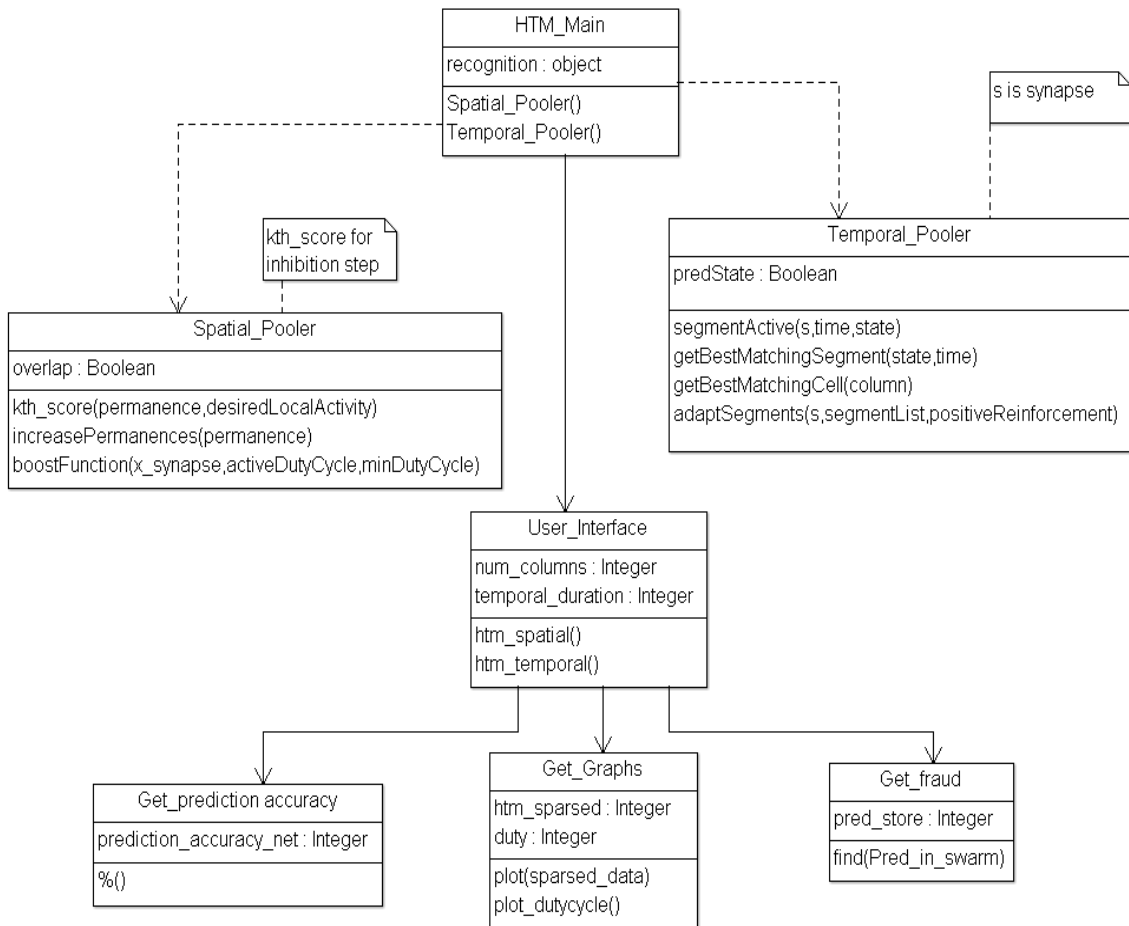


Figure 2: Computational Class Diagram of the System

Figure 2 is the computational class diagram of the system which shows the classes of the system, their interrelationships (including inheritance, aggregation, composition and association), and the operations and attributes of these classes.

4. EXPERIMENTATION AND RESULTS DISCUSSION

The system -CCF Smart Detector- is an online learning system that does its prediction while reviewing the streaming data. The user interface (UI) has been developed for ease of simulations. It is interactive and permits data loading, adjustment of HTM Network parameters and Cortical learning simulations. The CCF Smart Detector user interface structure is shown in Figure 3.

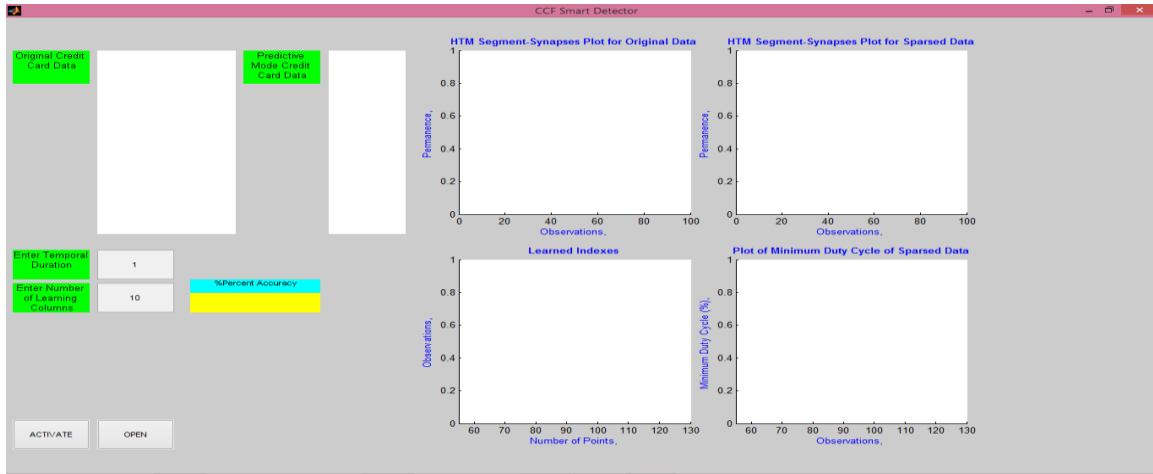


Figure 3: CCF Smart Detector User Interface

The experiment for the system was carried out thus:

1. Run the Main Application
2. Enter Network parameters e.g. number of learning columns, temporal duration.
3. Load the CCF data into memory
4. Click the Activate button
5. Read and record the prediction values
6. Save generated figure plots

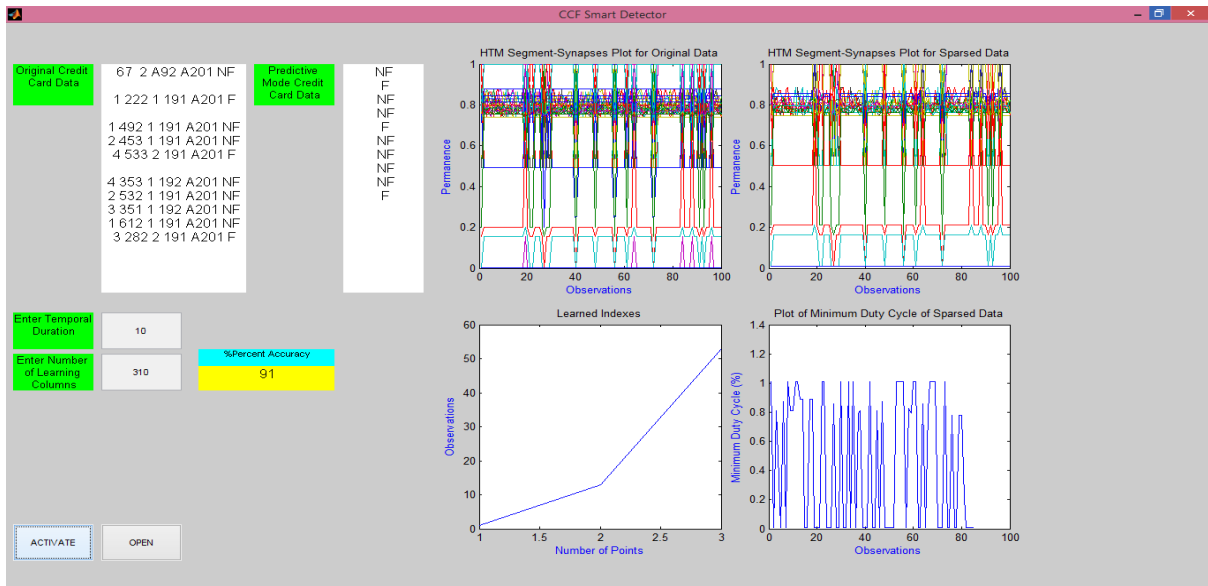


Figure 4: CCF Smart Detector User Interface with Cortical learning Simulation

The administrator selects the number of learning columns and temporal duration to use and then clicks the activate button. Once the system is activated, the original data comes in through the original credit card data panel, the CLA works on the streaming data as described in figure 1 and produces its prediction results on the predictive mode credit card panel. Figure 4 shows the interface after simulation and the visualization of how the CLA works in the conversion of the original data into SDR; it shows that the conversion does not change its learning activity of the system, as learning is also at its peak within same permanence range of 0.75 and 0.9 as it was in the original data. 310 learning columns was used in one of the simulations carried out and a percentage of 91 was recorded for the percentage accuracy of the system which already surpasses that of the existing system. The graphical representation of the learned indexes shows that the system learns better with more observations as in the case of the human brain. The number of learning columns influences the

percentage accuracy, experiments carried out has shown that with more learning columns, percentage accuracy is increased. The various simulations have been tabulated in Table 1 using the specified network parameters, from the tabulated results, an increase in the number of learning columns parameter, leads to increase in the percentage accuracy of the system, however, it would be at the expense of computational power. Figure 5 shows a graphical representation of the various simulations carried out.

Table 1 Simulation tests with prediction accuracies for each HTM/CLA run.

Run Attributes		HTM-CLA Prediction Accuracy
Cell size (in columns)	Number of Runs	% Accuracy
300	2	88

320	1	92
350	3	92
370	2	91
380	1	91
400	1	93
410	3	93
450	3	90
500	3	91
550	2	94
650	2	94
670	2	95
700	3	92
Run Attributes		HTM-CLA Prediction Accuracy
Cell size (in columns)	Number of Runs	% Accuracy
750	2	94
800	1	94
850	1	93
900	1	95
950	1	93
1000	1	93
1200	1	95
1500	1	95

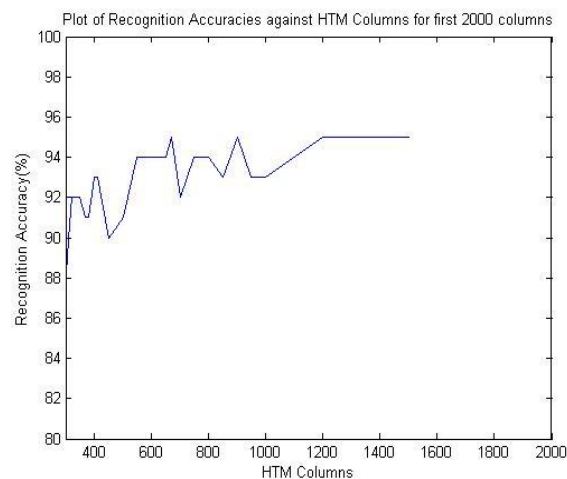


Figure 5: Graphical Representation of simulation results for the HTM system.

5. CONCLUSION

The paper has succeeded in developing a novel approach for credit card fraud detection and reduction of cases of misclassification to the barest minimum and efficiency of fraud detection was increased. The model experimentation was done in real time and the average percentage accuracy obtained was 91% in detecting fraud transactions in the use of credit card. This is an improvement to the neural network model in previous works that recorded a detection percentage of 89.6%. This implies the misclassification was reduced and the efficiency of fraud detection was increased in the new model. However, introduction of threshold activators to the cortical learning algorithm can be considered in future works on credit card fraud detection using cortical learning algorithm, since the permanence bias for the synapses of HTM cells is not always a successful solution to enhancing potential activation capability of HTM cells.

6. REFERENCES

- [1] Aleskerov, E., Freisleben, B. and Rao, B. (1997). Computational Intelligence for Financial Engineering (CIFER). CARDWATCH: A Neural Network based database mining system for Credit Card Fraud Detection. *In Proceedings of the Institute of Electrical and Electronics Engineers*. 220-226.
- [2] Azeem, S., Nadeem, A. and Mohammad N. (2014). Real-Time Credit-Card Fraud Detection using Artificial Neural Network Tuned by Simulated Annealing Algorithm. *In Proceedings of International Conference on Recent Trends in Information, Telecommunication and Computing*, 113-121.
- [3] Bamojo, O. (2006). *Pioneering Nigeria's Naira Credit Card*. TheNation, 26 November, 2006. Business Category.
- [4] Bhatla, T., Vikram, P. and Amit, Dua. (2003). Understanding Credit Card Frauds. *Tata Consultancy Services*, 1-14.
- [5] Chaudhary, K., Yadav, J. and Mallick, B. (2012). A review of Fraud Detection Techniques: Credit Card. *International Journal of Computer Applications*. Vol. 45. 975-988.
- [6] Dhok, S. (2012). Credit Card Fraud Detection Using Hidden Markov Model. *International Journal of Soft Computing and Engineering (IJSCE)*. Vol. 2, Issue 1. 231-237.
- [7] Ghosh, S. and Reilly, D. (1994). Credit Card Fraud Detection with a Neural-Network. *In Proceedings 27th Hawaii International Conference on System Sciences: Information Systems: Decision Support and Knowledge-Based Systems*, Vol. 3. 621-630.
- [8] Khan, Z., Jabir, D., and Ali, H. (2014). Credit Card Fraud Detection System Using Hidden Markov Model and K-Clustering. *International Journal of Advanced Research in Computer and Communication Engineering*, Vol. 3, Iss 2, 5458-5461.
- [9] Maes, S., Karl, T., Bram, V., Bernard, M. (2002). Credit Card Fraud Detection using Bayesian and Neural Networks. *Computational Modeling Lab (COMO)*, Belgium.
- [10] Neda, S. and Akbari, K. (2014). A novel model for Credit Card Fraud Detection using *Artificial Immune Systems*. *Applied Soft Computing (ASC)*, Elsevier. 40-49.
- [11] Numenta (2011). Hierarchical Temporal Memory. URL: <http://numenta.org/cla-whitepaper.html>
- [12] Pinheiro, A. (2014). *Nigeria's First Non-Bank Credit Card Launched*. ThisDay Live, (29 August, 2014). Business Category.
- [13] Sahin, Y. and Duman, E. (2011). Detecting Credit Card Fraud by Decision Trees and Support Vector Machines. *In Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS)*. Vol. 1, 1-6.