DESIGN AND SIMULATION OF AN EFFICIENT MODEL FOR CREDIT CARDS FRAUD DETECTION

Ibrahim K. Ogundoyin, Kudirat O. Jimoh, Lawrence O. Omotosho and Dimple T. Ogunbiyi

Department of Information and Communication Technology
Osun State University, Osogbo, Nigeria.
Ibraheem.ogundoyin@uniosun.edu.ng

ABSTRACT

In this study a model which can improve the accuracy and reliability of credit card fraud detection was proposed. This is with a few to mitigating contentious issues regarding online transaction of credit card, such as amount of transactions that have resulted in payment default and the number of credit card fraud cases that have been recorded, all of which have put the economy in jeopardy. To address this challenge,sample dataset was sourced from online repository database of Kaggle. The feature extraction on the data was performed using Principal Component Analysis (PCA). The credit card fraud detection model was designed using Neuro-fuzzy logic technique, clustering was done using Hierarchical Density Based Spatial Clustering of Application with Noise (HDBSCAN) .The simulation of the proposed model was done in Python programming environment.The performance evaluation of the model was carried out by comparing the proposed model with Neuro-Fuzzy (NF) technique using performance metrics such as precision, recall, F1-score and accuracy. The simulation result showed that the proposed model (NF + HDBSCAN) had precision of 98.75%, recall of 98.70%, F1-Score of 97.65% and accuracy 99.75% . NF had Precision of 94.60%, recall of 94.50%, F1-Score of 95.50% and accuracy 95.70% using training dataset. Likewise, when test dataset were used, the proposed (NF + HDBSCAN) had precision of 93.50%, recall of 95.50%, F1-Score of 94.50% and accuracy 95.50%. NF had Precision of 92.50%, recall of 93.00%, F1-Score of 94.00% and accuracy 93.50%. The simulation results of the proposed model was viable, reliable and showed possibility of being designed as module which could be integrated into the existing credit card design for lowering fraud rate and assisting fraud investigators.

Keywords: Fraud detection, Credit card, Simulation, Model, Neuro-fuzzy

1. Introduction

The advancement in network technologies has caused online transactions to grow fast over the last decade, making online transactions which include credit cards and other online payment systemsthe most popular mode of payment. However, there are some contentious issues regarding online transaction of credit card, such as the amount of transactions that have resulted in payment default and the number of credit card fraud cases that have been recorded, all of which have put the economy in jeopardy. Many studies have been reported on credit card fraud detection, but most these works face challenges of detection accuracy and reliability(Leo et al (2019); Varmedja et al (2019); Warghade et al (2020); Rahul and Amit (2020);).

Businesses, corporations, financial institutions, and other organizations are now offering online services such as e-commerce to provide clients with greater efficiency and accessibility. Because the card or cardholder does not need to be present for a transaction to be completed, it is difficult for businesses to identify if the customer is the true cardholder (Seera et al., 2021). The scam generally happens whenever anyone obtains credit or debit card numbers through unprotected websites or through an identity theft scheme in order to get money or property fraudulently. Because of the frequency with which it occurs and the financial institutions involved, it is crucial to take preventative steps as well as recognise when a transaction is fraudulent. It is possible to take the necessary preventative actions to stop this exploitation of such fraudulent acts, as well as to study how to limit it and protect against similar occurrences in the future.

Many machine learning techniques have been tested for their applicability in credit card fraud detection, including genetic algorithm, support vector machine, frequent itemset mining, decision tree, migrating bird's optimization algorithm, and naive bayes, among others (Maniraj et al (2019);Rahul and Amit (2020)). However, there are a number of challenges associated with credit card detection, such as fraudulent behavior profiles that are highly imbalanced (or skewed); optimal feature (variables) selection for credit card detection,

accuracy and reliability of fraud detection(Seeja and Zareapoor (2014);Venkata et al (2018)). Several studies have been reported, over time to improve credit card fraud detection accuracy and reliability utilizing machine learning approaches, still no work has reported absolute accuracy and reliability(Seeja and Zareapoor (2014);Venkata et al.(2018);Maniraj et al (2019);Rahul and Amit, (2020)).In order to better detect credit card fraud as well as address the detection problem. This study therefore, proposes an improved model in terms of detection accuracy and reliability.

2. Related Works

There are quite a good number of works regarding the state- of- the- art, credit card and other online transactions fraud detections. Seeja and Zareapoor (2014), presented FraudMiner: A Novel Credit Card Fraud Detection Algorithm Based on Frequent Itemset. Data Miningtechniquie was used to detect fraud from highly imbalanced and anonymized credit card transaction datasets using an intelligent credit card fraud detection model . The problem of class imbalance was addressed by employing frequent itemset mining to identify legitimate and fraudulent transaction patterns for each consumer. The proposed model was found to have a very high fraud detection rate, balanced classification rate, Matthews correlation coefficient, and very low false alarm rate when compared to other state-of-the-art classifiers.

Venkata et al (2018)conducted a study on Machine Learning approaches to Credit Card Fraud detection. In the research, a machine learning strategy based on logistic regression was used to detect credit card fraud. The findings suggest that the methodology based on logistic regression outperform MLP with the highest accuracy, and that it may be employed effectively by fraud investigators.

Maniraj et al (2019) proposed the use of Machine Learning and Data Science to detect credit card fraud. The authors noted that it is critical for credit card firms to be able to recognize fraudulent credit card transactions so that customers are not charged for products they did not purchase. The authors proposed that such issues may be solved with Data Science, and its relevance, along with that of Machine Learning. Themodel proposed in the study was utilized to determine whether or not a new transaction is fraudulent. The result of the proposed model in the study showed appreciable fraud detection while reducing the number of inaccurate fraud categories.
Varmedja et al (2019), proposed Machine Learning approaches for detecting credit card fraud. The study demonstrated a number of algorithms that can be used to determine if a transaction is fraudulent or real. The Credit Card Fraud Detection dataset was designed using SMOTE technique for oversampling because the dataset was significantly unbalanced. In addition, feature selection was carried out, and the dataset was divided into two parts: training data and test data. Logistic Regression, Random Forest, Naive Bayes, and Multilayer Perceptron were the algorithms employed in the experiment. The results suggested that each algorithm may be used to detect credit card fraud with high accuracy.

Oğuz and Layth (2020) performed a Comparative Analysis of Different Distributions Dataset by Using Data Mining Techniques On actual credit card transactions from European cardholders. The authors used four data mining techniques, namely naive Bayesian (NB), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Random Forest (RF). The authors further made the following four key contributions in this study. Because of the high imbalance class, which implies a skewed distribution, the authors utilized under-sampling to balance the dataset. Second, NB, SVM, KNN, and RF were used to classify under-sampled transactions into fraudulent and real, then tested and compared the performance metrics using a confusion matrix. Finally, cross-validation (CV) with ten folds to assess the accuracy of the four models with a standard deviation done, then compared the results for all the models used. Then, using the confusion matrix and AUC (Area Under the ROC Curve) ranking measure, the authors compared these models to the complete dataset (skewed) to determine which model would be the best for to be utilized with a certain sort of fraud. The accuracy of the NB, SVM, KNN, and MLP classifiers is 97.80%, 97.46 percent, 98.16 percent, and 98.23 percent, respectively.

Rahul and Amit (2020)presented an indication of different commonly available Data Mining (DM) and Machine Learning Technique (MLT) for detecting credit card fraud in a survey. The survey presented a review on Credit Card Fraud Detection Using Data Mining Classification Techniques and Machine Learning Algorithms. Data mining was presented to becoming an increasingly important aspect of the knowledge discovery process. The authors noted that fake transactions are unlikely to increase due to the rapid growth of cashless transactions. Proposing solution to credit card fraud, the authors proposed studying credit cards of diverse behaviors past transaction history dataset to identify a fraudulent transaction. The authors therefore defined a fraudulent transaction as one that deviates from the available cost pattern in any way. DM and MLT were identified as

reliable and commonly used techniques in the designed and implementation of credit card fraud detection.

Warghade et al (2020), proposed Credit Card Fraud Detection Using Machine Learning Algorithm from Imbalanced Dataset.The paper examined various machine learning algorithms utilizing a variety of measures to evaluate different classifiers. Rather than misclassifying a legitimate transaction as fraud, the strategy improved fraud detection.

Francis et al (2021) proposed a hybrid technique for detecting fraudulent transactions based on deep learning machine technique. The methodology of the study created a fraud detection system which successfully discovered fraudulent transactions in the given Kaggle dataset. The Kaggle dataset contains data that is unbalanced, with 99.83 percent normal data and 0.17 percent fraud data. The proposed hybrid technique of deep learning and machine learning addressed the imbalance in the dataset. Fraudulent transactions were detected with an accuracy of 87 percent using the proposed approach. The proposed model outperformed existing models (Isolation Forest, Local Outlier, and LSTM-Autoencoder), which had detection rates of 79 percent, 3 percent, and 82 percent, respectively.

El Naby et al (2021) proposed a Deep Learning Approach for Detecting Credit Card Fraud. In the paper, the author employed deep learning algorithms to effectively detect fraudsters in credit card transactions using Kaggle's credit card dataset. The model was designed using Over Sampling with Convolution Neural Network.The dataset was also subjected to the MLP (Multi-layer perceptron) algorithm. A comparative result showed that the proposed model outperformed other model used in the study.

Alam et al (2021)carried outcomparative performance analysis of random forest (RF), AdaBoost, and CatBoost classifiers for classification of credit card fraudulent activities. The classifiers were also used to choose the most significant attributes in in the dataset used in the study. The results demonstrate that MLP and CatBoost had the best performance among the classifiers tested, with 99.92 percent accuracy in detecting credit card frauds.

Several studies as reviewed above in literature have contributed immensely to credit card fraud detection and other unusual behaviors in electronic payment transactions systems. However,most existing works suffer setbacks in terms of reliability and detection accuracy. Therefore, in this study, a methodological approach which considered more details and new dynamics of credit card fraud detection through which a reliable detection accuracy and reliability are enhancedis proposed.

3. Materials and Methods

The methodology of this research includes various methods proposed to achieve the goal of the paper. The methods consist of: data collection and description, model formulation, simulation and evaluation of the proposed model performance. Figure 1 is the system architecture of the proposed credit card fraud detection model. From Figure 1, dataset would be loaded, preprocessing and feature extraction would take place. Thereafter, the preprocessed dataset would be classified as either fraud or normal. After classification, if the result turns to be fraudulent, then the card owner would be prompted to authenticate the transaction. If the result is normal,further processing would take place, the entire transaction log of the customer, which is the dataset would be clustered based on the current transaction, the density of clustered group would be calculated to know if such transaction or owner's behaviour is common in the previous transactions. If the density is above a set threshold such transaction would be deemed to be normal, but if it is below a set threshold, the card owner would be requested to authenticate the transaction before such transaction could be acceptable and added to the owner's log.

In other to achieve the aim and objectives of this paper, the sample dataset used in the study was obtained from kaggle. The feature extraction on the data was performed using Entropy Based Mutual Information Gain. The credit card fraud detection model was design using Neuro-fuzzy logic technique, clustering was done using Hierarchical Density Based Spatial Clustering of Application with Noise (HDBSCAN).The simulation of the proposed model was done in Python programming environment.The performance of the model was evaluated by comparing the proposed model with NF technique using performance metrics such as precision, recall, F1-score, and accuracy.

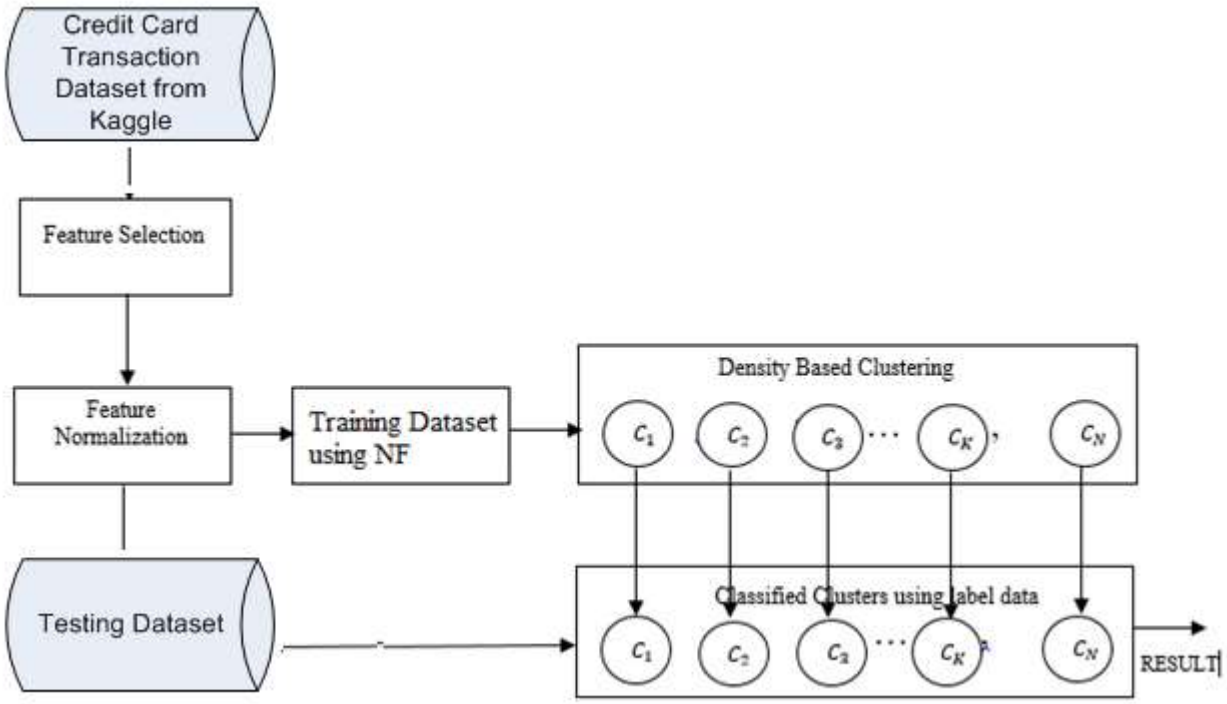

Figure 1:Thesystem flowchart of the proposed credit card fraud detection model

3.1 Description of Dataset

The Dataset used in this research was sourced from online repository database of Kaggle. The dataset is a simulated credit card transaction dataset that included both genuine and fraudulent transactions. It covers 1000 clients' credit cards for transactions with a pool of 800 merchants. Each row has its own unique identifier,Table 1 shows attribute description of dataset used in the study

Table 1: Attribute Description of Dataset used

S/N	Name of Attribute	Definition
1.	Id_trans	unique identifier
2.	Date	transaction date
3.	Time	Transaction time
4.	cc num -	Customer Credit Card Number
5.	merchant –	Merchant,
6.	Name category -	Category of Merchant
7.	amt	Amount of Transaction
8.	first	First Name of Credit Card Holder
9.	last	Last Name of Credit Card Holder,
10.	gender	Gender of Credit Card Holder,
11.	street	Credit Card Holder Street
12.	Address city	City of Credit Card Holder,
13.	state	State of Credit Card Holder,
14.	zip	Zip Code of Credit Card Holder
15.	lat	Credit Card Holder's Latitude,

16.	long	Credit Card Holder's Longitude,
17.	loc	Credit Card Holder's Location,
18	C_popl	city population,
19.	C_job	Job of Credit Card Holder
20.	dob	Credit Card Holder's Date of Birth,
21	trans num	Transaction Number,
22	unix time	- UNIX Transaction Time
23	Target Class	Fraud Flag

3.2 The Detail Algorithm of the Credit Card Fraud Detection Model Formulation

In this paper, the approaches of NF and HDBSCAN were used in the formulation of the proposed credit card fraud detection. The detailed algorithm of the proposed model formulation is as presented in Figure 3.

Algorithm 1: Model Formulation

1: Load transaction dataset from kaggle as data source
2: Extract necessary features from kaggletransaction dataset that follow standard (Detail in step 7) format using MS-Excel (Microsoft Excel) Sheet, targeting suspicious and non-suspicious transaction dataset
3: Load the dataset into dataframe in python using pandas
4: Analyse the dataset in dataframe using pandas and numpy to know the true state of the dataset
5: Normalized each transaction data record using Min-Max normalization:
6: $$Q = \left(\frac{P - Min(p)}{Max(P) - Min(P)}\right) * (N - M) + M \quad (1)$$
where the value of P feature needs to be normalized into value Q. *Min*(*P*) and *Max*(*P*) is the minimum and maximum values of feature P respectively. *M* and *N* indicates Lower and Upper Values respectively in the new range. (0,1) is used to normalized the features of *P*, this make *Q* to be in the range 0 and 1
7: Calculate the relevant of each feature to the label feature using Principal Component Analysis (PCA)
8: Select Best Feature with Highest information gain
9: Divide the dataset into training and testing in ratio 7.5:2.5, 75% for training and 25% for testing the model.
10: Scale the train dataset and test dataset for all the features values to be in the same scale.
11: Call function nf(), i.e. NF for training and classification of the dataset as illustrated in (section3.3)
- i. NF display the classification result, which could either be true or false, subject to further verification.
- ii. The current data record searches a clustered group of similar characteristics, this is performed in line 12.

12: Cluster the training dataset applying HDBSCAN clustering algorithm as illustrated in algorithm 2
- (a) Use Euclidean distance as metric parameter for the HDBSCAN as shown in Formula

$$Dist(p,q) = \sqrt{\sum_{i=1}^{n}(P_i - q_i)^2} \quad (2)$$

n is the number of the features for point object *p* and *q*, p_i and q_i are the first of the feature of point object *p* and *q*
- (b) The resulting cluster label is store in cluster. label, labeling the suspicious and Non-suspicious clusters as 0 and 1 and Noise as −1

13: Use the testing dataset as shown in Algorithm 3

14: After the HDBSCAN clustering, the density of the cluster of the current data record is calculated to determine if the transaction should be further subjected to verification or not.

Algorithm 2: HDBSCAN Main Steps

Input: B(Selected Best Features), Parameter m_{pts}
Output: HDBSCAN Hierarchy
1: **begin**
2: Compute the core distance w.r.t. m_{pts} for all objects in B
3: Compute an MST of G_{mpts}, the Mutual Reachability Graph
4: Extend the MST to obtain MST_{ext}, by adding for each vertex a "self edge" with the core distance of the corresponding object as weight.
5: Extract the HDBSCAN hierarchy as a dendrogram from MST_{ext}:
6: For the root of the tree assign all objects the same label (single "cluster")
7: Iteratively remove all edges from MST_{ext} in decreasing order of weights (in case of ties, edges must be removed simultaneously):
8: Before each removal, set the dendrogram scale value of the current hierarchical level as the weight of the edge(s) to be removed.
9: After each removal, assign labels to the connected component(s) that contain(s) the end vertex(-ices) of the removed edge(s), to obtain the next hierarchical level: assign a new cluster label to a component if it has at least one edge, else assign it a null label ("noise")
10: Return the clusters result in Clusterer and label others as Noise
11: **end**

Algorithm 3: Classifying Test Dataset
Input: Testpoints(M_i,*i*= 1,...,*n), NF, Clusterer*
Output: Classified Testpoints

1: **begin**
2: Input Testpoint into HDBSCAN approximate predict
3: **for** each TestpointsM_i **do**
4: Classify and predict Testpoints class
5: **end for**
6: Return classified Testpoints from the HDBSCAN approximate predict
7: **end**

Figure 2: Model Formulation

3.3 Principal Component Analysis

Principal Component Analysis is considered as a dimensionality reduction method. Principal Component Analysis is a simple, non-parametric method of extracting relevant information from large datasets. Principal Component Analysis helps in finding a mapping from inputs in original d-dimensional space to a new k (k<d) dimensional space, with minimum loss of information and data. Principal Component Analysis is concerned with explaining the variance-covariance of a set of variables (multi-variant).

$x_1, x_2, x_3, \dots, x_p$ (3)

Through a few linear combinations of these variables to capture the variability of original dataset (Covariance Matrix)

$a_1x_1 + a_2x_2 + a_3x_3 \dots + a_px_p$ (4)

The variance of each variable is the average squared deviation of its n values around the mean of that variable.

$V_i = \frac{1}{n-1}\sum_{m=1}^{n}(x_{im} - \bar{x}_i)^2$ (5)

The degree to which the variables are linearly correlated is represented by their covariance's.

$C_{ij} = \frac{1}{n-1}\sum_{m=1}^{n}(X_{im} - \bar{X}_i)(X_{jm} - \bar{X}_j)$ (6)

Where

C_{ij} = Covariance Of variables i and j

$\sum_{m=1}^{n}$ = Sum over All n objects

X_{im} = Value of variable i in object m

$\bar{X}_i$ = Mean of variable i

X_{jm} = Value of variable j in object m

$\bar{X}_j$ = Mean of variable j

3.4 The Neuro-fuzzy Approach

The NF uses a feed forward network to search for fuzzy decision rules that perform excellently on a given task using the input-output data set. The NF model is a network framework consisting of a number of nodes that are connected through direct links. Each node represents a process unit, and the links between nodes specify the causal rapport between the linked nodes. It is a five layer network that can accept different input variables from ranges 1 to 10. The architecture of a NF model for two inputs x and y is shown in Figure 3. All or parts of the nodes are adaptive, which makes the output of the nodes to depend on modifiable parameters pertaining to these nodes. The learning rules specify how the parameters should be updated depending on a stopping criterion. The NF systems are multilayer feed forward adaptive networks that realize the basic elements and functions of traditional fuzzy logic. Since fuzzy logic systems are universal approximators, the NF systems can also be put to use as universal approximators. The basic operations of the layers can be seen in the nodes for each layer. At the first layer, each input parameter is clustered into several class values to build up fuzzy rules, and for each input, the membership grades in the corresponding fuzzy sets are estimated as shown in equation(3) and (4) as follows:

$$Q_{1,i} = \mu_{Ai}(x) \quad 7)$$

$$Q_{1,i} = \mu_{Bi-2(y)} \quad (8)$$

A_i, B_{i-2} are a fuzzy set associated with this node. At the second layer, each fuzzy rule would be constructed through several parameters of membership function, fuzzy intersection is used to calculate the firing strength of each rule

$Q_{2,I} = W_i$ (9)

$$W_i = \mu_{Ai(x)} * \mu_{Bi(y),} i = 1,2 \quad (10)$$

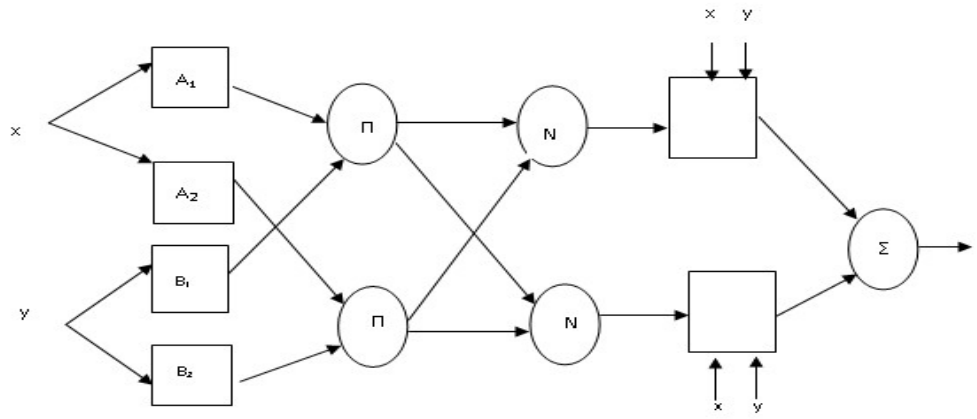

Figure 3: Neuro-Fuzzy Architecture (Hikmet , 2006).

The third layer is used for the calculation of the ratio of its rule's firing strength to the sum of all rule's firing strengths.

$$Q_{3,1} = \varpi = \frac{w_i}{w_1 + w_2}, i = 1,2 \tag{11}$$

The fourth layer multiplies the normalized firing strength with the linear result

$$Q_{4,1} = \varpi_i f_i = \varpi_i (p_{ix} + q_{ix} + r_i) \tag{12}$$

Where ϖ_i is the output of layer 3 and p_i, q_i and r_i are the parameter set.
These parameters are known as consequent parameters.
The fifth layer calculates the overall output as the sum of all incoming signals from layer 4

$$\text{Overall output} = Q_{5,i} = \sum \varpi_i f_i \tag{13}$$

$$Q_{5,i} = \frac{\sum_i \varpi_i f_i}{\sum_i \varpi_i} \tag{14}$$

The second and third layer contains nodes that form the antecedent parts in each rule. The fourth layer performs the parameter estimation for the model.

3.4 Experimental Design and Simulation Environment

Simulation was setup for the proposed credit card fraud detection model in Pythonenvironment.
To meet the goal of the study, dataset obtained from kaggle was used. This is because utilizing an existing dataset that requires credit card transactions from bank clients in reality are difficult to get and are rarely being disclosed, owing to market competitiveness, as well as legal reasons and user data privacy. The feature extraction on the data was performed using Entropy Based (Mutual information gain). The credit card fraud detection model was design using Neuro-fuzzy logic technique, clustering was done using Hierarchical Density Based Spatial Clustering of Application with Noise (HDBSCAN). The "!pip install'' keyword was used to import libraries that are not in collab by default. Different libraries were imported such as; Keras, Tensorflow, NumPy, glob, shuttle, sklearn, imutils, matplotlib, argparse. After the preparation of the simulation environment, the first step was loading of the collected dataset. The second step was the preprocessing procedure which involved cleaning, standardizing of the dataset and feature extraction. Cardholders spending behavior were inputted into the NF, which was used to train the system. The trained model was then tested in two ways: first, the training dataset were used to test the trained NF model. Second, the test datasets were as well used to test the NF model. Results in both cases were obtained. Another level of testing carried out on the model was a complete model testing. In this case, records of the dataset in test dataset were tested for fraud. If the result turns to be fraudulent, then the card owner would be prompted to authenticate the transaction. If the result is normal,

further processing would take place, the entire transaction log of the customer, which is the dataset, would be clustered based on the current transaction being tested using HDBSCAN. Thereafter, the density of clustered group was calculated to know if such transaction or owner's behaviour is common in the previous transactions. If the density is above a set threshold, such transaction was deemed to be normal, but if it is below a set threshold, the card owner was requested to authenticate the transaction before such transaction could be acceptable and added to the owner's log if normal, else it would be labelled as fraud. Figure 4 is a system flowchart of the simulation scenario. The clustering and density of transaction were the features added to the NF model to ensure improved reliability and accuracy of the proposed model. The clustering performed in this simulation using HDBSCAN was major, and was done on three attributes:location, amount of transaction and frequency of transaction. The entire location in the dataset was group into six, i.e. the card holder can operate within the six locations. Each location comprises of many regions grouped together. On the amount used for transaction, it was classified as low, medium and high. Frequency of transaction was also classified as low, medium and high. Table 2 Shows the cluster group based on the attributes, location, amount of transaction and frequency.

Table 2: Cluster showing the card owner behaviors

Location	Transaction Amount (TA) /Transaction Frequency (TF)		
	Low (L_{TA}, L_{TF})	Medium (M_{TA}, M_{TF})	High (H_{TA}, H_{TF})
1	1, L_{TA}, L_{TF}	1, M_{TA}, M_{TF}	1, H_{TA}, H_{TF}
2	2, L_{TA}, L_{TF}	2, M_{TA}, M_{TF}	2, H_{TA}, H_{TF}
3	3, L_{TA}, L_{TF}	3, M_{TA}, M_{TF}	3, H_{TA}, H_{TF}
4	4, L_{TA}, L_{TF}	4, M_{TA}, M_{TF}	4, H_{TA}, H_{TF}
5	5, L_{TA}, L_{TF}	5, M_{TA}, M_{TF}	5, H_{TA}, H_{TF}
6	6, L_{TA}, L_{TF}	6, M_{TA}, M_{TF}	6, H_{TA}, H_{TF}

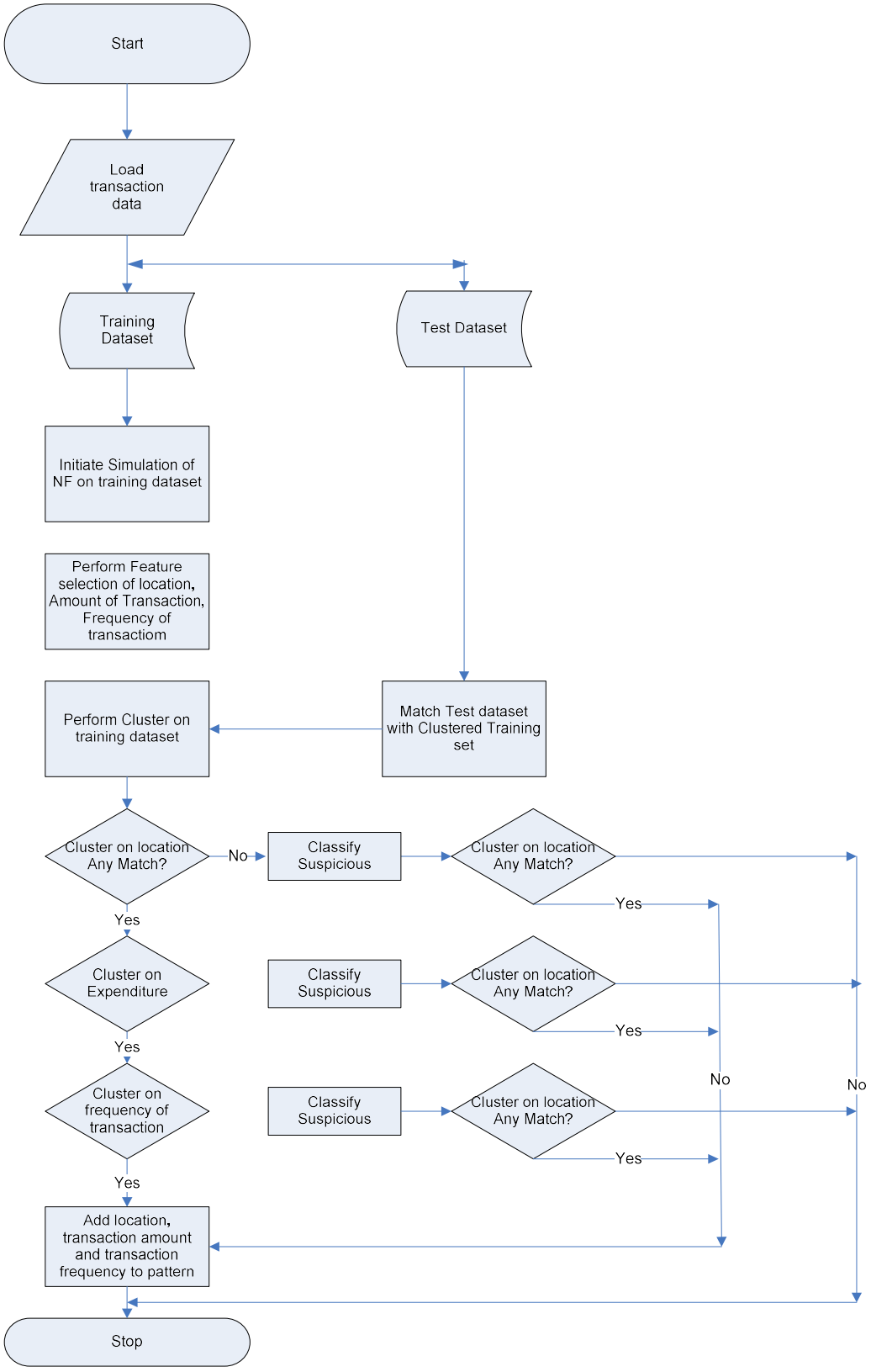

Figure 4: System Flowchart of the Simulation Scenario

3.5. Model Performance Evaluation.

In order to evaluate the performance of the proposed credit card fraud detection model, some metrics such as Accuracy, Precision, F1-Score and Recall were used:

i. Accuracy

Accuracy in classification problems is the number of correct predictions made by the model over all kinds predictions made. The formula for Accuracy is

$$\frac{TP+TN}{TP+TN+FP+FN} \quad (15)$$

ii. Precision

Precision is a measure that shows what proportion of actual predictions from the dataset. The formula for this is

$$\frac{TP}{TP+FP} \quad (16)$$

iii. Recall

Recall is the measure of our model correctly identifying True Positives. The formula for this is

$$\frac{TP}{TP+FN} \quad (17)$$

iv. F1-Score

F1-score is the Harmonic mean of the Precision and Recall. The formula is

$$2 * \frac{Precision * Recall}{Precision + Recall} \quad (18)$$

4.0 Result and Discussion

In this section, the proposed credit card fraud detection model was evaluated by comparing its performance with based NF as proposed in this study. The proposed model was evaluated with training and test datasets. The comparative performance evaluations of the proposed model for training and test datasets were presented in Table 2 and Table 3. In Table 2, the performance of the proposed model (NF + HDBSCAN) was compared with NF model using training dataset. The proposed (NF + HDBSCAN) had Precision of 98.75%, Recall of 98.70%, F1-Score of 97.65% and Accuracy 99.75%. NF had Precision of 94.60%, Recall of 94.50%, F1-Score of 95.50% and Accuracy 95.70%. Likewise in Table 3, using test dataset, the proposed (NF + HDBSCAN) had Precision of 93.50%, Recall of 95.50%, F1-Score of 94.50% and Accuracy 95.50%. NF had Precision of 92.50%, Recall of 93.00%, F1-Score of 94.00% and Accuracy 93.50%.

Table 2: Comparative Performance Evaluation of the Proposed Model with NF During Training.

Performance Metrics +%	Proposed Model (NF + HDBSCAN)	NF
Precision	98.75	94.60
Recall	98.70	94.50
F1-score	97.65	95.50
Accuracy	99.75	95.70

Table 3:Comparative Performance Evaluation of the Proposed Model with NFDuringTest.

Performance Metrics in %	Proposed Model (NF + HDBSCAN)	NF
Precision	93.50	92.50
Recall	95.50	93.0
F1-score	94.50	92.50
Accuracy	95.50	93.50

Discussing the result, the proposed model (NF + HDBSCAN) produced the most reliable result in terms of the performance metrics used. The discrepancies in performance of the proposed NF + HDBSCAN and NF during test period as compared to training period was because the test dataset were not used for the model training, and so the proposed model was not conversant with the test dataset. From the results, both at training and test periods, the adoption of the NF + HDBSCAN in the formulation of the credit card fraud detection model wasadequate and reliable going by the simulation results. The integration of the fraud detection as a module in the credit card design will enhance the security features of the credit card, thereby safeguarding the users.

5.0 Conclusion

In this research, a model for credit card fraud detection was designed, formulated and simulated. The model was formulated based on observation, expert knowledge and a critical study of dynamics of credit card holder behaviours. The transaction dataset collected from kaggle was analyzed, preprocessed, and relevant features extracted and inputted into the proposed models(NF + HDBSCAN). The result generated was then analyzed and then used to classify sample dataset into suspicious and non-suspicious transaction. The simulation results of the proposed model was viable, reliable and showed possibility of being designed as module which could be integrated into the existing credit card design to for lowering losses and assisting fraud investigators.

Reference

Alam, M. N., Podder, P., Bharati, S., and Mondal, M. R. H. (2021). *Effective Machine Learning Approaches for Credit Card Fraud Detection,* International Conference on Innovations in Bio-Inspired Computing and Applications, 16th- 18th Dec 2021(Online), (pp. 154–163).

El Naby, A. A., El-Din Hemdan, E., and El-Sayed, A. (2021). Deep Learning Approach for Credit Card Fraud Detection. *2021 International Conference on Electronic Engineering (ICEEM)*, Menoufia University, Egypt,3 July 2021, Pg,1–5.

Francis, C., Dong, H. L., and Han, S. J. (2021). Deep learning based hybrid approach of detecting fraudulent transactions. *Journal of Theoretical and Applied Information Technology*, *99*(16), 4044–4054.

Leo, M., Sharma, S., and Maddulety, K. (2019). Machine Learning in Banking Risk Management: A Literature Review. *Risks*, *7*(1), 2 - 22.

Maniraj, S., Aditya, S., Shadab, A., and Swarna, D. S. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. *International Journal of Engineering Research And*, *08*(09): 110 - 115.

Oğuz, A., and Layth, H. (2020). Comparative Analysis of Different Distributions Dataset by Using Data Mining Techniques on Credit Card Fraud Detection. *Tehnicki Vjesnik - Technical Gazette*, *27*(2): 618-626.

Rahul, G., and Amit, Kumar, M. (2020). Review on Credit Card Fraud Detection using Data Mining Classification Techniques and Machine Learning Algorithms. *International Journal of Research and Analytical Reviews*, *7*(1), 972 - 976.

Seera, M., Lim, C. P., Kumar, A., Dhamotharan, L., and Tan, K. H. (2021). An intelligent payment card fraud detection system. *Annals of Operations Research*, 2021, 1-23

Seeja, K. R., and Zareapoor, M. (2014). FraudMiner: A Novel Credit Card Fraud Detection Model Based on Frequent Itemset Mining. *The Scientific World Journal*, *2014*, 1–10.

Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., and Anderla, A. (2019). Credit Card Fraud Detection - Machine Learning methods. *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, 1–5.

Venkata Suryanarayana, S., N. Balaji, G., and Venkateswara Rao, G. (2018). Machine Learning Approaches for Credit Card Fraud Detection. *International Journal of Engineering and Technology*, *7*(2): 917 - 920.

Warghade, S., Desai, S., and Patil, V. (2020). Credit Card Fraud Detection from Imbalanced Dataset Using Machine Learning Algorithm. *International Journal of Computer Trends and Technology*, *68*(3), 22–28.