Performance Evaluation of Machine Learning Models for Cyber Threat Detection and Prevention in Mobile Money Services

Bodunde Odunola Akinyemi[1*], Dauda Akinwuyi Olalere[2], Mistura Laide Sanni[1], Emmanuel Ajayi Olajubu[1], Ganiyu Adesola Aderounmu[1], Isa Ali Ibrahim[3]
[1]Obafemi Awolowo University, Ile-Ife, Nigeria
[2]MTN, Nigeria
[3]Research and Development Department, Federal Ministry of Communications and Digital Economy
E-mail: bakinyemi@oauife.edu.ng, DaudaO@mtnnigeria.net, msanni@oauife.edu.ng, emmolajubu@oauife.edu.ng, gaderoun@oauife.edu.ng, isaaliibrahim@hotmail.com

Keywords: machine learning, SMOTE, mobile money, cyber threats, evaluation, predictive models

Received: February 20, 2023

In this paper, an investigation was made to evaluate the effectiveness of the different classifiers suitable to predict the probability of a cyber-threat or fraudulent intent applicant during the Mobile Money Service on-boarding or service activation process, with the goal of determining the best machine learning model for the predictive model solution. Experimental work was carried out by formulating cyber threat predictive models using six supervised machine learning algorithms: Logistic regression(LR), Naïve Bayes, Shallow Neural Network (SNN), Deep Neural Network (DNN), Classification and Regression Trees (CART) and Random Forest (RF) of different configurations. Each model was simulated with both Synthetic Minority Operation Techniques (SMOTE) and without SMOTE (No-SMOTE) on 25,000 records of mobile money applicants. Twenty-four (24) different configurations of the formulated predictive models were simulated and evaluated using the Python programming language. Simulation results of the predictive models proved that the Random Forest model multiclass configurations with the SMOTE dataset outperformed all other configurations. The results also showed that the multiclass experiments with SMOTE had better performance than the binary configurations with NO-SMOTE in the predictive models. The study concluded that using the Random Forest-based predictive machine learning model will increase the security level of the Mobile Money solution by detecting and preventing anomalous customer registrations during the unbanked onboarding process.

Povzetek: Napovedovanje kibernetskih groženj mobilnega denarja z uporabo algoritmov strojnega učenja.

1 Introduction

Modern economies are today inspired mostly by digital currency, and the widespread usage of mobile devices has opened up a new market for digital financial services in emerging countries [1]. These developments have made it easier for the underprivileged people in these nations to access financial services [2–5]. In Africa, there is currently a great deal of demand for promoting financial inclusion, premised on the willingness of the nations to adopt financial inclusion action plans in order to eradicate poverty and boost their economies [6–8].

Unbanked financial services such as Mobile Money Services (MMS) typically function using smartphone applications that are backed by mobile operators or banking institutions. Despite the mobile money sector's expansion and its enormous prospects, research indicates that the adoption of Mobile Financial Services (MFS) is still low in sub-Saharan Africa [8]. The widespread use of mobile devices has significantly increased the number of people who have access to the Internet. As mobile money adoption continues to gain ground, fraudsters are now focusing on this new money transfer route [9–10]. As a result, this advancement has inadvertently ushered in a brand-new age of crime: cybercrime. Financial fraud has evolved and become more complex in recent years as a result of the widespread use of advanced technology. Consumers now accept mobile money as one of the latest means of getting access to financial services that offer quality, affordability, and ease of use. Meanwhile, criminals have discovered new ways to move their illicit funds or fund criminal activities covertly. Therefore, it is commonly acknowledged that the frequency of crime driven by the economy in many societies poses a serious danger to the growth and stability of the global economy.

Fraud is a global financial concern that endangers the viability of MMS. The likelihood of cybercrime in MMS is rising and becoming more pervasive [11]. If financial crime aimed at various stakeholders, mobile money agents, and Mobile Network Operator (MNO) systems is not properly addressed, it may deter people from using MMS, potentially undoing years of progress towards financial inclusion [12]. It was observed that due to the exclusion of inclusive development, inadequate security standards by both service providers, and the resulting restrictions and behaviour of mobile end users, Africa as a

continent lag behind all the other continents in financial inclusion [13]. With the increasing usage of MMS in these countries, it is critical to develop a comprehensive scheme for mobile money security that would alleviate security vulnerabilities and mitigate fraud, as several mobile money service providers have suffered huge losses in revenues due to this emerging threat.

It is impossible to overstate the importance of humans in successful cyberattacks against MFS transactions. They could be the attack's instigator, medium, or real perpetrator [13]. The administration of human stakeholders is highly essential to the mobile money security system. From platforms to platforms, internal and external users, and more especially the customer management strategy or practises employed to set up, update, and activate the users by the operators.

As a result, the risks posed by the human factor in the intensification of cybercrime on mobile money initiatives must be predicted and avoided through robust and intelligent countermeasures [14]. The existing methodology employs rule-based algorithms and manual eyeballing for the identification and blocking of fraudulent customer registrations [13]. This methodology is frequently time-consuming for the agents, uneconomical, and ineffective for detecting cyber-threats. Fraudsters are encouraged by the MMT services' quick proliferation, and MNOs that offer these services are required to identify ML activity. It is crucial that the tools for detection be effective at detecting threats and simple to use [15].

There have previously been a variety of methods used to address financial transaction fraud. These techniques included rule-based and related statistical techniques. The rule-based technique has a high proportion of false-positive outcomes and is time-consuming and expensive. However, these techniques are gradually losing their effectiveness as criminal behaviour patterns and operating procedures get more sophisticated [16]. The emphasis has shifted away from conventional, rule-based approaches to more advanced computational methods.

Applications of Artificial Intelligence (AI), data mining, and Machine Learning (ML) models have been discovered to reduce fraud in high-risk mobile payments and decrease false declines. Researchers have demonstrated the effectiveness of these methods in predicting the cyber threat to MMS and financial crimes [17]. ML addresses the problems associated with conventional approaches by allowing computers to adapt to data and generate predictions. When incorporated into MMS, ML is utilised to deliver automatic detection of potentially fraudulent activities. A collection of transactions that have been presumed to be fraudulent would be used to train an ML algorithm. The algorithm could be adjusted to identify impending fraudulent transactions based on learned experience by identifying patterns that match those in the training data. ML algorithms proactively detect suspicious transactions in real-time, swiftly identify and block transactions that may be fraudulent, minimise the number of fraudulent transactions, and consequently eliminate the need for significant human engagement.

Various pre-processing procedures or data transformation methods have been employed to enhance the data quality and, subsequently, the classification accuracy of the Financial Inclusion dataset [18]. ML algorithms are increasingly being used to predict fraudulent transactions. These algorithms, whether supervised or unsupervised, including logistic regression (LR), K-nearest neighbor (KNN), Support Vector Machines (SVM) and Naive Bayes, are trained with datasets and utilised to categorise and classify mobile financial transactions into valid and suspicious ones. Artificial Neural Networks (ANNs) and Convolutional Neural Networks (CNNs), among other deep learning techniques, have additionally been utilised to find anomalies in financial transactions. A significant degree of prediction accuracy has been exhibited by these deep learning and ML systems [19].

All indications point to the conclusion that ML models are useful for automating and modelling cyber-risk assessment in MMS. However, there is a need to investigate the performance of various machine learning classification models that can anticipate malicious customers having cyber-threat risks during the on-boarding procedures for MMS in the developing world.

The main objective of this investigation is to evaluate the effectiveness of several classifiers that can predict an applicant's likelihood of being a cyber-threat or having fraudulent intentions during the MMS onboarding or service activation process in order to discover the most accurate predictive ML model.

2 Related works

A significant chunk of past research in the field of mobile money focused on the best way to use MMS effectively while reducing fraud and financial concerns. These studies examine the variables that influence the successful application of mobile money. Most research on fraud prediction and detection using AI, data mining, and other statistical techniques that has been conducted in the domain of finance has focused on credit card fraud detection.

A state-of-the-art survey on the security issues of MMS identified some conventional methods that have been employed to improve the security of MMS [20–22]. These techniques include biometric methods [23], quantitative analysis of subject matter [24], two-factor authentication [25], structural equation modelling [26], case-based reasoning [27], and a variety of others. It was, however, noted that these conventional methods for combating fraud in MMS are ineffective due to the problems of cybercrime [28]. There are many difficulties with the procedures, rules, and measures for MMS to offer tools for curbing cybercrime threats because no practical solution has been offered, particularly in the context of developing countries, as demonstrated by the survey conducted in [29].

Investigation and research into various models and methodologies have become necessary due to the necessity of building a plan for managing the significant

risk of mobile money fraud detection. Any dataset on financial transactions, like MMS, has a relatively small fraction of transactions that are fraudulent (positive class) as opposed to valid (negative class). Because of this, the datasets are really imbalanced [30], and ML algorithms that use this data to make predictions are biased in favour of valid transactions, which has the long-term impact of making predictions based on this data potentially false.

Credit card transactions have a very imbalanced class distribution because fraud typically accounts for less than 1% of total transactions. Different computational sampling approaches have been used to address the issue of imbalanced data, such as K-means clustering and genetic algorithms [31], a hybrid model based on genetic algorithms [32], and kernel principal component analysis [33], which were used as feature selection methods with some chosen ML algorithms to detect fraud. Despite being a straightforward solution to the issue of data skewness, random undersampling or oversampling still introduces uninformative or unhelpful sub-structures in datasets.

The suitability of several machine learning models for fraud detection and classification techniques has been examined [34–35]. Methods of supervised learning are widely applied in the investigation of fraud. These models were used to predict the likelihood of credit card fraud based on a certain number of transactions. Some of the experimental works are SVM and Back Propagation Networks [36]; Weighted Support Vector Machine [37]; Naive Bayes, LR, SVM, and KNN [38]; KNN, Random Forest (RF), LR, Decision Tree, and Naive Bayes classifiers [39]; and comparison of various machine learning models for binary categorization of imbalanced credit card fraud data [34–35]. These applications of these approaches were assessed based on their accuracy, precision, specificity, and sensitivity. The results provide optimal accuracy for the classifiers supported by LR, SVM, Naive Bayes, and KNN, as shown in the summary in Table 1. Results from databases of credit card transactions demonstrate the effectiveness and efficiency of these ML algorithms in the fight against financial transaction fraud.

However, the majority of supervised learning techniques for fraud detection have typically been established with the presumption that the mobile money ecosystem is relatively harmless, i.e., that there are no enemies attempting to defeat MMS. Meanwhile, the MMS is now bedevilled by attacks. Given this situation, potential fraudster behaviours were taken into account in MMS using ML techniques [19, 28, 40–41]. Utilization of graph-theoretical methods to identify fraud schemes that result in long-term changes in the typical behaviour of MMS customers [42]. Additionally, ML models were employed to anticipate the adoption of mobile money [43–44].

Recently, a prediction model using a LR classifier was developed and assessed in order to identify and mitigate suspicious clients with the ability to commit cybercrime during the onboarding processes for MMS in emerging regions. Employing binary and multiclass setups, with or without Synthetic Minority Oversampling Technique (SMOTE or No-SMOTE), the model's performance in identifying and categorising fraudulent MMS application intentions was examined [13]. Among the different configurations of the experiments using LR, the results showed that the LR classifier with the SMOTE application achieved the highest classification accuracy.

Also, in order to categorise and forecast fraud in mobile money transactions, investigations were carried out on how well the LR classifier performed by experimenting with various undersampling, weighting, and oversampling strategies [19]. The findings demonstrated that manually adjusting the class weights for false positives and false negatives was the most effective model for these tests.

In most of the investigations conducted, the LR classifier and random forest model have been recognised as having the most exceptional performance among all measures, while other classifiers were highly beneficial in predicting suspicious transactions. Among all the classifiers, these two models were the most reliable and effective because they could be modified to reach high precision and successfully learn from data with multiple features. Despite the fact that LR and random forest classifiers are effective ML techniques for detecting fraud, more research is still required to examine how well other ML classification models perform in predicting suspicious customers with the potential for cyber threats during the on-boarding process for MMS in developing countries and produce a more conclusive result.

3 Methodology

The goal of this investigation is to develop and evaluate a reliable model for predicting fraudulent mobile money transactions. The work employed supervised learning algorithms to construct an effective prediction system for MMS, using known normal and fraud cases to train the models and uncover their properties.

In this study, machine learning models for cyber threat detection and prevention were developed. Analytical models were employed to ascertain the validity of incoming registration or activation record details from Mobile Money applicants. Supervised learning algorithms were used for the model building as follows:

Six (6) machine learning algorithm models were used for modelling the prediction of cyber-threat during MMS activation via customer on-boarding or SIM registration processes, namely LR, SNN, DNN, Naive Bayes, Decision Trees (Cart-Classification and Regression Trees), and RF. To avoid class imbalance, the length of the positive class was oversampled with synthetic data using the Synthetic Minority Oversampling Technique (SMOTE).

The algorithms were developed with a broad range of configurations determined by two variants: one based on balancing the dataset using SMOTE or not, and two based on binary and multiclass configurations of the algorithms, leveraging the experimental work done in [13]. This brought about the six (6) supervised learning algorithms having four different variants based on the configurations, finally resulting in a total of twenty-four (24) algorithms

as shown in Table 2. The historical SIM registration data set for new applications and current customers served as the training dataset for the models. To train the model, numerous iterations of this were done.

Table 1: Literature review summary table

Research work	Problems addressed and Techniques used	Dataset Distribution	Feature Classification	Results
[13]	Using logistic regression to create a prediction model to identify suspicious customers with potential cyber-threats	SMOTE	Binary and Multiclass	LR gives good results
[33]	Analysed the effectiveness of naive bayes, KNN, and LR on data from credit card fraud that is incredibly imbalanced.	oversampling and under-sampling	Binary	KNN performs better
[34]	Compare LR, RF, Naive Bayes and Multilayer Perceptron models for detection of fraud data	SMOTE	Binary	RF algorithm gives the best results
[35]	To investigate SVM-S and Back Propagation Networks (BPN) for building models representing normal and abnormal customer behavior	Random under-sampling	Binary	SVM-S have better prediction performance than Back Propagation Networks (BPN)
[36]	To judge the veracity of the LR, SVM, and RF algorithm in Credit Card Fraud Detection	random under-sampling	Binary	A weighted SVM model methodology perform best
[37]	To examine highly skewed data on credit card fraud using SVM, Naive Bayes, LR, and KNN	random under-sampling	Binary	LR was the most accurate
[38]	Exploring the use of KNN, Naive Bayes, Decision Trees, LR, and RF models to forecast the likelihood that a fraudulent credit card transaction would occur .	Imbalanced Dataset	Binary classification	Decision Tree Model is the best approach

Table 2: Rules for classifying records of mobile money applicants

	Algorithms	***Description***
A	***Logistics Regression***	
1	**LR Binary-No SMOTE**	Logistic Regression with *Binary* feature configuration and *NO*-SMOTE application to Dataset
2	**LR Binary-SMOTE**	Logistic Regression with *Binary* feature configuration and *with* SMOTE application to Dataset
3	**LR Multiclass-No SMOTE**	Logistic Regression with *Multiclass* feature configuration and *No*-SMOTE application to Dataset
4	**LR Multiclass-SMOTE**	Logistic Regression with *Multiclass* feature configuration and *with* SMOTE application to Dataset
B	***Shallow Neural Network***	
5	**SNN Binary-No SMOTE**	Shallow Neural Network with *Binary* feature configuration and *No*-SMOTE application to Dataset
6	**SNN Binary-SMOTE**	Shallow Neural Network with *Binary* feature configuration and *with* SMOTE application to Dataset
7	**SNN Multiclass-No SMOTE**	Shallow Neural Network with *Multiclass* feature configuration and *No*-SMOTE application to Dataset
8	**SNN Multiclass-SMOTE**	Shallow Neural Network with *Multiclass* feature configuration and *with* SMOTE application to Dataset
C	***Deep Neural Network***	
9	**DNN Binary-No SMOTE**	Deep Neural Network with *Binary* feature configuration and *No*-SMOTE application to Dataset
10	**DNN Binary-SMOTE**	Deep Neural Network with *Binary* feature configuration and *with* SMOTE application to Dataset
11	**DNN Multiclass-No SMOTE**	Deep Neural Network with *Multiclass* feature configuration and *No*- MOTE application to Dataset
12	**DNN Multiclass-SMOTE**	Deep Neural Network with *Multiclass* feature configuration and *with* SMOTE application to Dataset
D	***Naïve Bayes(NB)***	
13	**NB Binary-No SMOTE**	Naïve Bayes(NB) with *Binary* feature configuration and *No*-SMOTE application to Dataset
14	**NB Binary-SMOTE**	Naïve Bayes(NB) with *Binary* feature configuration and *with* SMOTE application to Dataset
15	**NB Multiclass-No SMOTE**	Naïve Bayes(NB) with *Multiclass* feature configuration and *No*-SMOTE application to Dataset
16	**NB Multiclass-SMOTE**	Naïve Bayes(NB) with *Multiclass* feature configuration and *with* SMOTE application to Dataset
E	***Decision Tree(CART)***	
17	**CART Binary-No SMOTE**	Decision Tree(CART) with *Binary* feature configuration and *No*-SMOTE application to Dataset
18	**CART Binary-SMOTE**	Decision Tree(CART) with *Binary* feature configuration and *with* SMOTE application to Dataset
19	**CART Multiclass-No SMOTE**	Decision Tree(CART)with *Multiclass* feature configuration and *No*-SMOTE application to Dataset
20	**CART Multiclass-SMOTE**	Decision Tree(CART) with *Multiclass* feature configuration and *with* SMOTE application to Dataset
F	***Random Forest(RF)***	
21	**RF Binary-No SMOTE**	Random Forest(RF) with *Binary* feature configuration and *No*-SMOTE application to Dataset

22	**RF Binary-SMOTE**	Random Forest(RF) with *Binary* feature configuration and *with* SMOTE application to Dataset
23	**RF Multiclass-No SMOTE**	Random Forest(RF) with *Multiclass* feature configuration and *No*-SMOTE application to Dataset
24	**RF Multiclass-SMOTE**	Random Forest(RF) with *Multiclass* feature configuration and *with* SMOTE application to Dataset

4 Results and discussions

The Python 3.7 programming language software was used for data analysis, which supports ML methods and data conversion and transformation capabilities. The simulation of the predictive model for detecting and preventing mobile money cyber-attacks was conducted at the time of registering the mobile money applicants' cyber threat intent prediction based on the applicant's biodata registration details to identify an applicant with malicious intentions while on-boarding in order to choose the ideal machine learning model for the outcome. Performance evaluation parameters and metrics were also defined for performance measures.

4.1 Result analysis by experiment grouping

The experiments were performed according to [13]. They were grouped into two for each algorithm, with two experiments per group, making a total of four experiments performed per algorithm. These experiments were done with or without rebalancing of the imbalanced dataset using SMOTE with the aim of seeking the best performing algorithm for the Mobile Money on-boarding process cyber threat predictions into multiple classes of applicants' details as compliant, class 0, low risks, class 1, and high risks, class 2.

For Group I experiments, the classifiers were tested with their default binary classification capability for classifying the classifiers' ability to categorise the applicants' records into compliant (zero) and bi-level illegitimate registration categories: low risk (1) and high risk (2). Before running the algorithms, the dataset was treated in two ways after the required preprocessing of string and categorical variables with bag of words and label and one-hot encoding, respectively. The dataset was unbalanced in the distribution of target classifications; each algorithm was run on the dataset, and after applying SMOTE and No-SMOTE, the dataset was rebalanced. The non-application of SMOTE constitutes experiment I, while the application of SMOTE before running the dataset constitutes experiment II of group I. The dataset was divided into a 70% training and 30% testing set. The results obtained are shown in Table 3 for Group I simulation experiments.

Overall, the outcome indicated that the dataset's balancing properties had a significant impact on the findings when the two scenarios were compared for all twelve experiments conducted in groups for different variants of the six algorithms simulated. Also, datasets with SMOTE performed better than when the SMOTE operation was not performed on the dataset before running the algorithm, except in the case of LR, where a binary feature No-SMOTE (accuracy = 0.72, MCC = 0.16) performed better than one with SMOTE (accuracy =0.42, MCC = 0.15), as in Table 3. However, a closer look at the confusion matrix for the classification showed that the LR just completed a binary classification and not into multiple classes of compliant (0), low risk (1), and high risk (2).

For Group II experiments, as shown in Table 4, these experiments test the multi-classification performance of the algorithms when the dataset was used with No-SMOTE and when SMOTE was applied to the unbalanced dataset for rebalancing. Again, the multiclass algorithm ran on a balanced dataset after the SMOTE application and performed better than those with No-SMOTE. In group II experiments, Random Forest has the highest performance indicator of mcc (0.88), accuracy (0.91), and misclassification rate of 0.05. Thus, the overall performance of the algorithms was better with the multiclass feature enabled and when SMOTE was applied to the dataset before algorithm training and testing. The ability of the classifiers to classify the dataset showed the reliability of the pre-processing processes used by the bag of words to string features in the dataset and SMOTE applications. Due to the underperformance of the binary algorithms, the multiclass feature of the algorithms also aids in better performance.

4.2 Result analysis by individual predictive machine learning algorithm models

Each classifier was trained with different configurations of the classifiers, such as binary or multiclass with the integration of SMOTE and No-SMOTE for MM applicant fraudulent intent detection and classification. For each algorithm, the classifier was trained to build the analytical model, and the results discussion for each was presented subsequently. Each classifier was run five times, and the average of the accuracy metrics was taken.

A. Logistics regression (LR) experiments

Evaluating the classification capability of LR in terms of its accuracy and the Mathews Correlation Coefficient (MCC) with unbalanced (No-SMOTE) and balanced (SMOTE) datasets shows marked differences. The results are described as follows:

(i.) **Accuracy and MCC:** With unbalanced datasets, a deceptively high prediction accuracy of 0.72 was observed with the default binary classification feature of the algorithm for the classified applicant's dataset (into compliant (class 0) and the two categories of cyber-threat risks: low risks (class 1) and high risks (class 2) in Experiment I of Group I, while with balanced datasets, the accuracy dropped to 0.42 in Experiment II of Group II, thus showing the true algorithm classification performance. This showed

that the default feature of LR was to do binary classification and not a good multi-class classifier as the confusion matrix revealed that the classification with an unbalanced dataset of 0.72 accuracy only classified the dataset into two classes: class 0 and class 2. However, when multi-class configuration was used with the LR classifier, the performance was better with SMOTE and NO-SMOTE as the datasets were classified into the three classes: class 0, class 1, and class 2. The classification accuracy was high for both No-SMOTE (0.71) and SMOTE (0.72). When including SMOTE, accuracy (0.72) was the same as when there was no SMOTE using the binary logistics feature, but with SMOTE, the classifier did classify

Table 3: Machine learning algorithm binary features for cyber threat prediction experiments

		MCC	*Accuracy*	*F1-Score*	*Precision*	*Mis-classification Rate*	*AUC*	TNR(Specificity)	FPR	TPR(Sensitivity)	*Runtime (min)*
Group I Experiment II	LR Binary-SMOTE	0.15	0.42	0.47	0.59	0.57	0.64	0.71	0.29	0.42	0.0256
	SNN Binary-SMOTE	0.53	0.67	0.69	0.77	0.33	0.87	0.83	0.17	0.67	0.0117
	DNN Binary-SMOTE	0.19	0.43	0.49	0.73	0.57	0.64	0.72	0.28	0.43	0.03
	NB Binary-SMOTE	0.31	0.54	0.53	0.55	0.46	0.72	0.77	0.23	0.54	0.01
	CART Binary-SMOTE	0.53	0.68	0.69	0.71	0.32	0.77	0.84	0.16	0.68	0.02
	RF Binary-SMOTE	0.86	0.90	0.90	0.92	0.10	0.98	0.95	0.05	0.90	0.18
Group I Experiment I	LR Binary-No SMOTE	0.16	0.72	0.79	0.92	0.29	0.62	0.76	0.24	0.48	0.0115
	SNN Binary-No SMOTE	0.25	0.62	0.63	0.7	0.39	0.69	0.76	0.24	0.49	0.2444
	DNN Binary-No SMOTE	0.2	0.71	0.81	0.96	0.29	0.56	0.69	0.31	0.39	0.1400
	NB Binary-No SMOTE	0.18	0.69	0.75	0.83	0.31	0.64	0.71	0.29	0.41	0.0100
	CART Binary-No SMOTE	0.34	0.74	0.78	0.85	0.26	0.64	0.75	0.25	0.51	0.3900
	RF Binary-No SMOTE	0.50	0.79	0.86	0.96	0.21	0.78	0.77	0.23	0.56	0.5300

Table 4: Machine learning algorithm multiclass features for cyber threat prediction experiments

		MCC	*Accuracy*	*F1-Score*	*Precision*	*Mis-classification Rate*	*AUC*	TNR (Specificity)	FPR	TPR (Sensitivity)	*Runtime (min)*
Group II Experiment II	LR Multiclass-SMOTE	0.58	0.72	0.72	0.72	0.28	0.84	0.86	0.14	0.72	0.0857
	SNN Multiclass-SMOTE	0.59	0.72	0.72	0.73	0.28	0.87	0.86	0.14	0.72	1.8200
	DNN Multiclass-SMOTE	0.43	0.61	0.65	0.76	0.39	0.87	0.80	0.20	0.61	1.3100
	NB Multiclass-SMOTE	0.34	0.56	0.56	0.58	0.44	0.74	0.78	0.22	0.56	0.0200
	CART Multiclass-SMOTE	0.82	0.88	0.88	0.88	0.12	0.89	0.94	0.06	0.88	0.3300
	RF Multiclass-SMOTE	0.88	0.91	0.91	0.93	0.09	0.99	0.95	0.05	0.90	0.400
Group II	LR Multiclass-No SMOTE	0.27	0.69	0.71	0.74	0.31	0.71	0.75	0.25	0.48	0.0156
	SNN Multiclass-No SMOTE	0.3	0.72	0.76	0.84	0.28	0.69	0.74	0.26	0.47	1.0100

	DNN Multiclass-No SMOTE	0.3	0.73	0.81	0.92	0.27	0.69	0.72	0.28	0.45	0.5400
	NB Multiclass-No SMOTE	0.24	0.72	0.78	0.88	0.28	0.66	0.72	0.28	0.42	0.0100
	CART Multiclass-No SMOTE	0.392	0.733	0.763	0.807	0.267	0.63	0.79	0.21	0.59	1.7400
	RF Multiclass-No SMOTE	0.51	0.79	0.86	0.96	0.21	0.78	0.77	0.23	0.56	1.7000

into distinct three classes, which made the performance better in the context of the multi-classification of cyber threat risks. For a clear performance evaluation, MCC was also used to substantiate the evaluation.

The MCC gave a very clear distinction and better performance measurements; hence, the LR experiment with the best classifier configuration was the configuration with Multiclass with SMOTE among the four LR experiments performed, which had the highest MCC of 0.58 when compared with other experiments MCCs of 0.27, 0.15, and 0.16, as shown in Table 5 and Figure 1.

(ii.) **Precision, recall and F1-score:** If the dataset was fairly balanced, accuracy as an evaluation metric would suffice for a sound conclusion; however, precision, recall, and F1 score are good for evaluating an imbalanced dataset. The fraud detection dataset was unbalanced; hence, to evaluate the effectiveness of such a model, examining the precision and recall is very important. As presented in Table 5, the precision or specificity for the LR binary with No-SMOTE classification experiment was 0.76, and the recall or sensitivity was 0.48 when compared with the multiclass classification logistic regression model specificity of 0.86 and sensitivity of 0.72.

(iii.)**ROC and predicted probabilities:** The Receiver Operating Characteristics (ROC) Area Under Curve (AUC) for multiclass LR of 0.84 was also higher than for all other LR experiments. This further buttresses the fact that the LR classifier (including the SMOTE application) provides the best classification performance among the various configurations of LR experiments performed. The ROC AUC value is presented in Table 5 and Figure 2. Thus, SMOTE improves the performance of the LR classifiers, although the multiclass with LR gave the best performance.

B. Shallow neural network (SNN) experiments

Evaluating the classification capability of SNN in terms of its accuracy and Mathews Correlation Coefficient (MCC) with the unbalanced (No-SMOTE) and balanced (SMOTE) dataset shows marked differences, as presented in Table 6, Figures 3 and 4. The results are described as follows:

(i.) Accuracy and MCC: The accuracy of both multiclass experiments remains the highest and equal 0.72 among the four experiments performed for SNN; however, the MCC value revealed the best algorithm with a value of 0.59 for multiclass configuration with SMOTE and 0.3 for multiclass configuration with No-SMOTE. The MCC thus revealed the algorithm configuration with the optimal efficiency among the different configurations of the SNN experiments for the classification into classes 0, 1, and 2.

(ii.) Precision, recall and F1-score: The performance parameters of the multi-class with SMOTE configuration of the SNN experiments were the highest, with a specificity of 0.86, a sensitivity of 0.72, and an F1-Score of 0.72, which implies it has the best performance among the other SNN experiment configurations.

(iii.)ROC and predicted probabilities: The Receiver Operating Characteristics (ROC) Area Under Curve (AUC) of 0.87 for multiclass SNN was the highest for the dataset with the SMOTE application, and this was the same for the binary configurations. This was the most successful of the experiments with No-SMOTE.

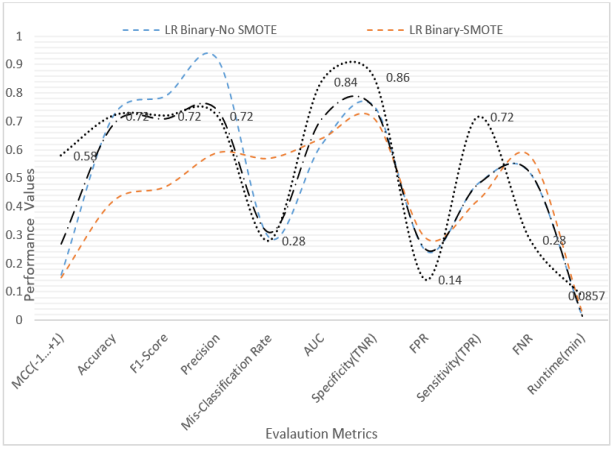

Figure 1: Different LR configuration results by performance parameters.

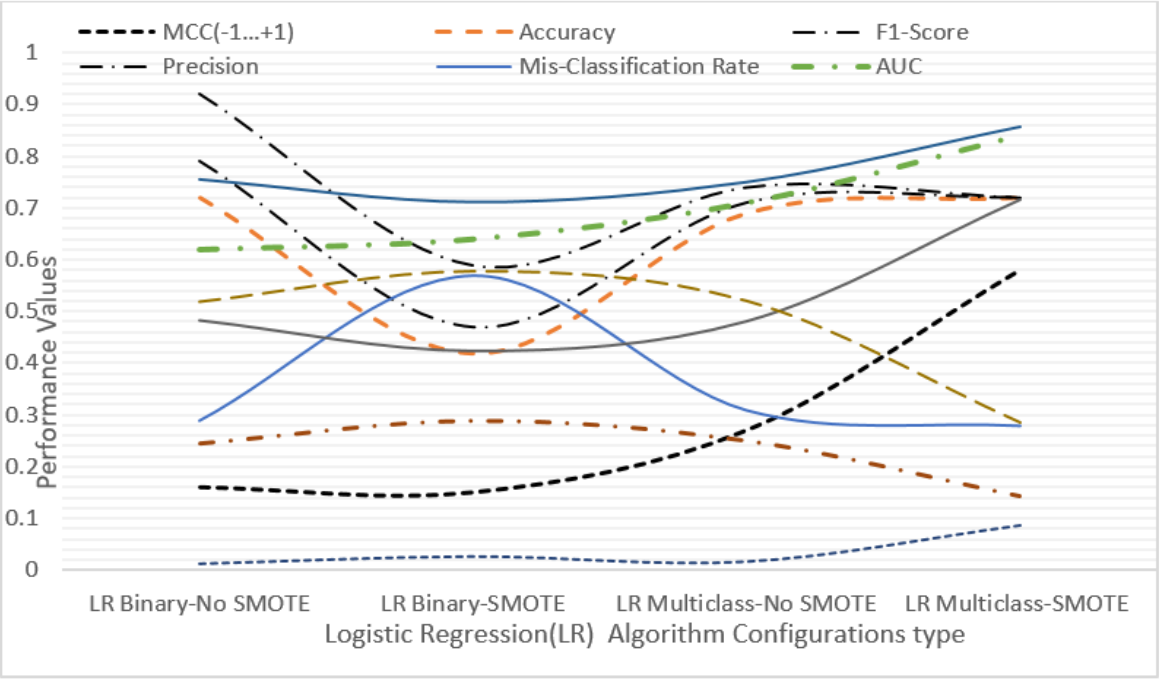

Figure 2: LR performance results for different configurations.

Table 5: Performance metrics for LR with (SMOTE)/without SMOTE (No-SMOTE)

Logistics Regression	*MCC (-1+1)*	*Accuracy*	*F1-Score*	*Precision*	*Mis-Classification Rate*	*ROC AUC*	Specificity (TNR)	FPR	Sensitivity (TPR)	FNR	*Runtime (min)*
LR Binary-No SMOTE	0.16	0.72	0.79	0.92	0.29	0.62	0.76	0.24	0.48	0.52	0.0115
LR Binary-SMOTE	0.15	0.42	0.47	0.59	0.57	0.64	0.71	0.29	0.42	0.58	0.0256
LR Multiclass-No SMOTE	0.27	0.69	0.71	0.74	0.31	0.71	0.75	0.25	0.48	0.52	0.0156
LR Multiclass-SMOTE	0.58	0.72	0.72	0.72	0.28	0.84	0.86	0.14	0.72	0.28	0.0857

Table 6: Performance metrics for SNN with (SMOTE)/without SMOTE (No-SMOTE)

Shallow Neural Network(SNN)	MCC	Accuracy	F1-Score	Precision	Mis-classification Rate	AUC	Specificity (TNR)	FPR	Sensitivity (TPR)	FNR	Runtime (min)
SNN Binary-No SMOTE	0.25	0.62	0.63	0.7	0.39	0.69	0.76	0.24	0.49	0.51	0.2444
SNN Binary-SMOTE	0.53	0.67	0.69	0.77	0.33	0.87	0.83	0.17	0.67	0.33	0.0117
SNN Multiclass-No SMOTE	0.3	0.72	0.76	0.84	0.28	0.69	0.74	0.26	0.47	0.53	1.0100
SNN Multiclass-SMOTE	0.59	0.72	0.72	0.73	0.28	0.87	0.86	0.14	0.72	0.28	1.8200

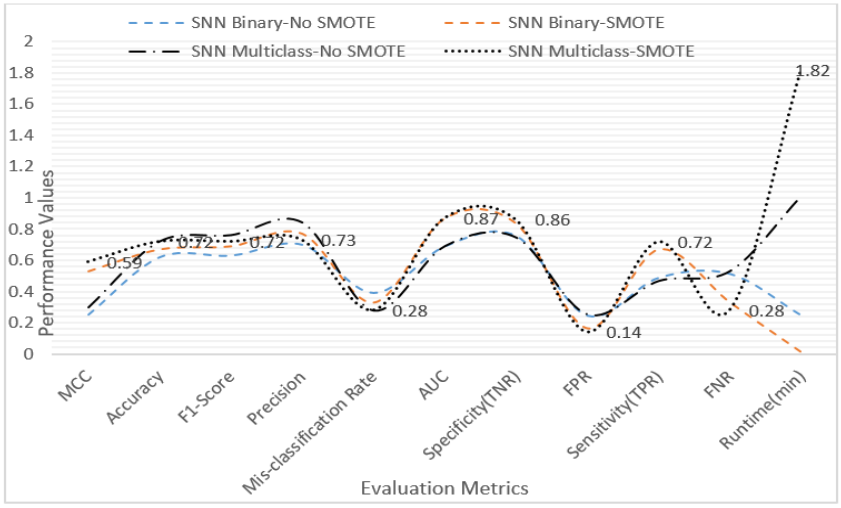

Figure 3: Different Shallow Neural Network (SNN) configuration results by performance parameters.

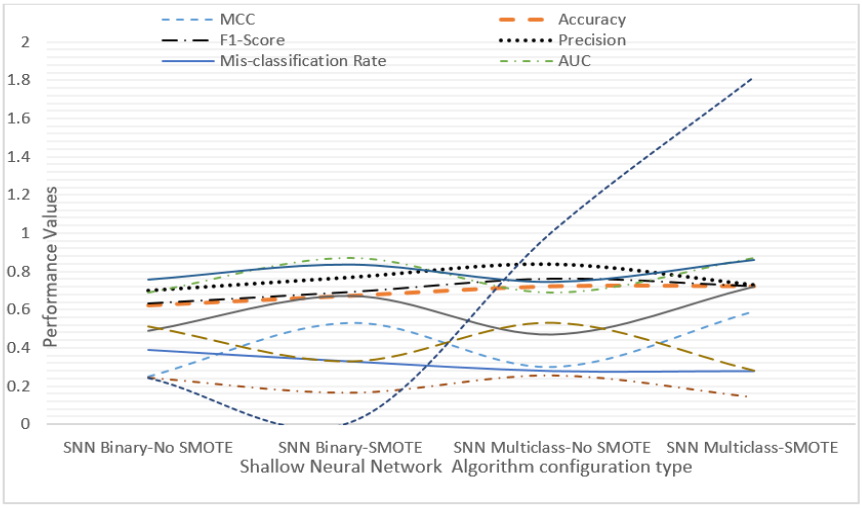

Figure 4: Shallow Neural Network (SNN) performance results for different configurations.

C. Deep neural network (DNN) experiments

Evaluating the classification capability of the DNN in terms of its accuracy and the Mathews Correlation Coefficient (MCC) with unbalanced (No-SMOTE) and balanced (SMOTE) datasets shows marked differences, as presented in Table 7 and Figures 5 and 6. The results are described as follows:

(i.) Accuracy and MCC: For the DNN, the classifier's performance was poorer than that of the SNN in all scenarios tested. This was evident from the accuracy metrics, with the highest accuracy of 0.71 for the binary and No-SMOTE configurations for the classifier, and the algorithm classified the dataset into two of the three classes. The multiclass with SMOTE configuration had the best performance among the DNN experiments performed, as revealed by the MCC value of 0.43 (but with an accuracy of 0.61), which is the highest among all the experiments. However, the overall performance is weak.

(ii.) Precision, recall and F1-score: The performance of the DNN was poorer than that of the SNN when compared with SNN performance parameters. The multi-class with SMOTE configuration of the DNN experiments had a specificity of 0.80, a sensitivity of 0.61, and an F1-Score of 0.65, which clearly showed a lower performance.

(iii.) ROC and predicted probabilities: The Receiver Operating Characteristics (ROC) Area Under Curve (AUC) of 0.87 for multiclass DNN with SMOTE was also the same as that of the SNN configurations in the multiclass experiment with SMOTE.

D. Naïve Bayes experiments

Evaluating the classification capability of Naïve Bayes in terms of its accuracy and the Mathews Correlation Coefficient (MCC) with unbalanced (No-SMOTE) and balanced (SMOTE) datasets shows marked differences, as presented in Table 8, Figures 7 and 8. The results are described as follows:

(i.) Accuracy and MCC: The No-SMOTE experiments for Naïve Bayes experiments performed better than with SMOTE application in terms of accuracy (0.69 and 0.71 for the binary No-SMOTE and multiclass with SMOTE, respectively); however, the MCC (0.31 and 0.34 for the binary with SMOTE and multiclass with SMOTE, respectively) and confusion matrix distributions showed that the multiclass performed better for the experiments. This still reinforces the fact that accuracy may not always be the best performance metric for evaluation. In general, Naïve Bayes performed poorly by performance metrics; however, it was the fastest algorithm in all the experiments performed, taking less than 2 seconds to run.

(ii.) Precision, recall and F1-score: The specificity of 0.78 and the sensitivity of 0.56 reinforce the fact that the multiclass configuration with the SMOTE application had the best performance out of all the Naïve Bayes experiments.

(iii.) ROC and predicted probabilities: The Receiver Operating Characteristics (ROC) Area Under Curve (AUC) of 0.74 for multiclass Naïve Bayes also highlights that multiclass with SMOTE application revealed that the multiclass performed the best for Naïve Bayes.

E. Decision tree (classification and regression tress- CART) experiments

The CART performed second best overall in the overall simulation for MMS cyber threat detection. Evaluating the classification capability of the CART algorithm in terms of its performance metrics with SMOTE and NO-SMOTE applied to the dataset for experimenting with the performance of CART to classify mobile money applicants showed marked differences, as presented in Table 9 and Figures 9 and 10. The results are described as follows:

(i.) **Accuracy and MCC**: Decision Tree algorithm performed very well with SMOTE application to the dataset, with an accuracy of 0.53 for binary and 0.88 for multiclass configurations within a very reasonable time. The multiclass configuration performed overall best among the four experiments performed for the algorithm, with an MCC of 0.82, which was far above any of the experiments for the algorithmic CART. The MCC thus further confirms the authority of the algorithm's performance.

(ii.) **Precision, recall and F1-score**: The specificity and sensitivity also gave interesting results for CART as a high-performing classifier in the research modelling scenario for a multiclass CART configuration with SMOTE. The specificity was 0.94 and the sensitivity was 0.88, which was the highest for the decision tree (CART) experiment performed.

(iii.) **ROC and predicted probabilities**: The ROC AUC of 0.89 for multiclass CART also confirms a relatively high performance for decision trees.

F. Random forest (RF) experiments

Random Forest performed the best overall for the predictive model. The ensemble classifier, Random Forest (RF), performance evaluation with SMOTE and

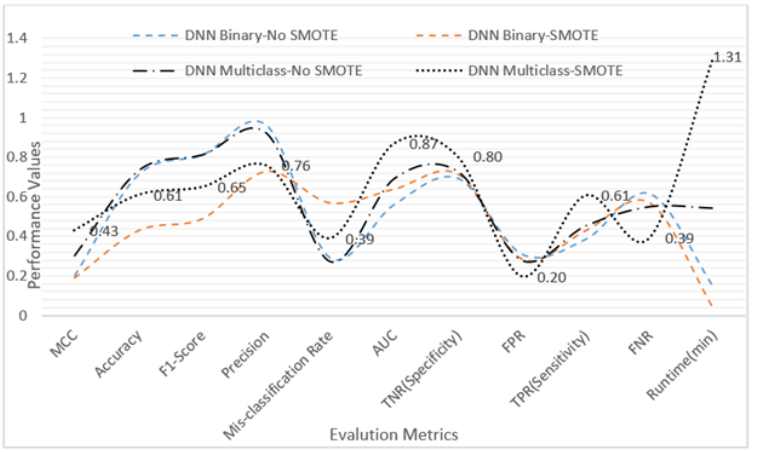

Figure 5: Different deep neural network (DNN) configuration results by performance parameters.

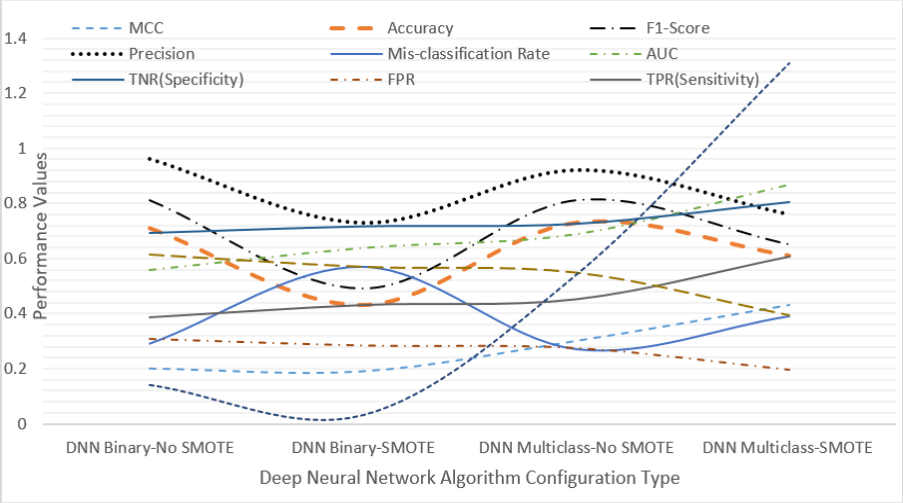

Figure 6: Deep neural network (DNN) performance results for different configurations.

Table 7: Performance metrics for deep neural network (DNN) with (SMOTE)/without SMOTE (No-SMOTE)

Deep Neural Network	MCC	Accuracy	F1-Score	Precision	Mis-classification Rate	AUC	Specificity(TNR)	FPR	TRP(Sensitivity)	FNR	Runtime(min)
DNN Binary-No SMOTE	0.2	0.71	0.81	0.96	0.29	0.56	0.69	0.31	0.39	0.61	0.14
DNN Binary-SMOTE	0.19	0.43	0.49	0.73	0.57	0.64	0.72	0.28	0.43	0.57	0.03
DNN Multiclass-No SMOTE	0.3	0.73	0.81	0.92	0.27	0.69	0.72	0.28	0.45	0.55	0.54
DNN Multiclass-SMOTE	0.43	0.61	0.65	0.76	0.39	0.87	0.80	0.20	0.61	0.40	1.31

Table 8: Performance metrics for naïve bayes (NB) with (SMOTE)/without SMOTE (No-SMOTE)

Naïve Bayes(NB)	MCC	Accuracy	F1-Score	Precision	Mis-classification Rate	ROC AUC	TNR(Specificity)	FPR	TRP(Sensitivity)	FNR	Runtime(min)
NB Binary-No SMOTE	0.18	0.69	0.75	0.83	0.31	0.64	0.71	0.29	0.41	0.59	0.01
NB Binary-SMOTE	+0.31	0.54	0.53	0.55	0.46	0.72	0.77	0.23	0.54	0.46	0.01
NB Multiclass-No SMOTE	+0.24	0.72	0.78	0.88	0.28	0.66	0.72	0.28	0.42	0.58	0.01
NB Multiclass-SMOTE	0.34	0.56	0.56	0.58	0.44	0.74	0.78	0.22	0.56	0.44	0.02

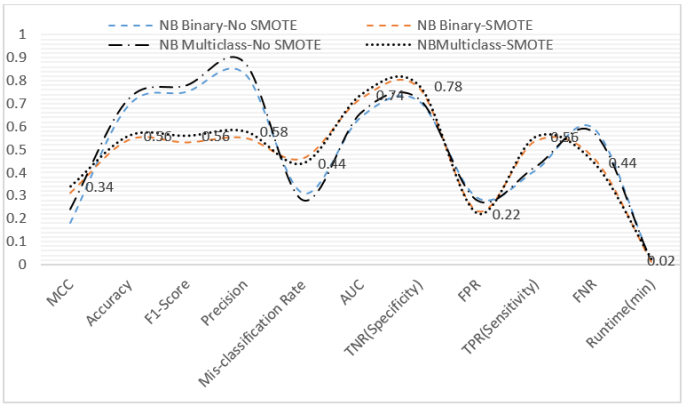

Figure 7: Different naïve bayes algorithm configuration results by performance parameters.

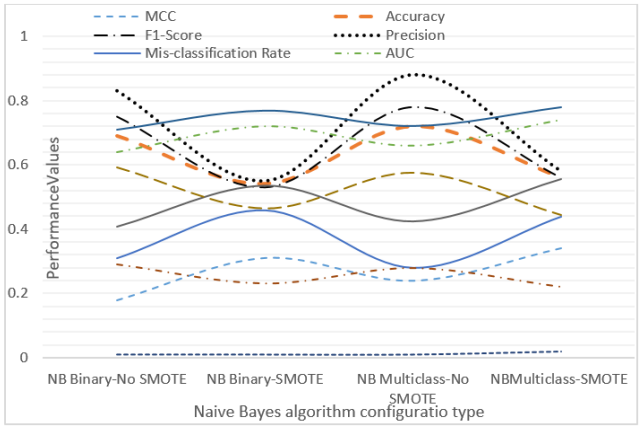

Figure 8: Naïve bayes performance results for different configurations.

Table 9: Performance metrics for decision trees (CART with (SMOTE)/without SMOTE (No-SMOTE)

Decision Tree(CART)	MCC	Accuracy	F1-Score	Precision	Mis-classification Rate	AUC	TNR(Specificity)	FPR	TPR(Sensitivity)	FNR	Runtime(min)
CART Binary-No SMOTE	0.34	0.74	0.78	0.85	0.26	0.64	0.75	0.25	0.51	0.49	0.39
CART Binary-SMOTE	0.53	0.68	0.69	0.71	0.32	0.77	0.84	0.16	0.68	0.32	0.02
CART Multiclass-No SMOTE	0.39	0.73	0.763	0.807	0.267	0.63	0.79	0.21	0.59	0.41	1.74
CART Multiclass-SMOTE	0.82	0.88	0.88	0.88	0.12	0.89	0.94	0.06	0.88	0.12	0.33

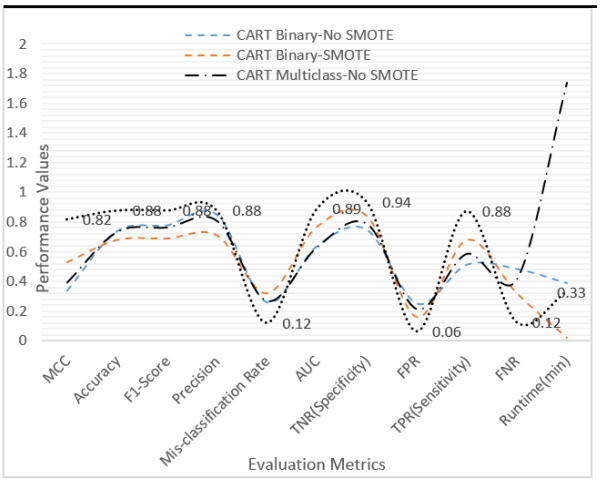

Figure 9: Different decision trees (CART) algorithm configuration results by performance parameters.

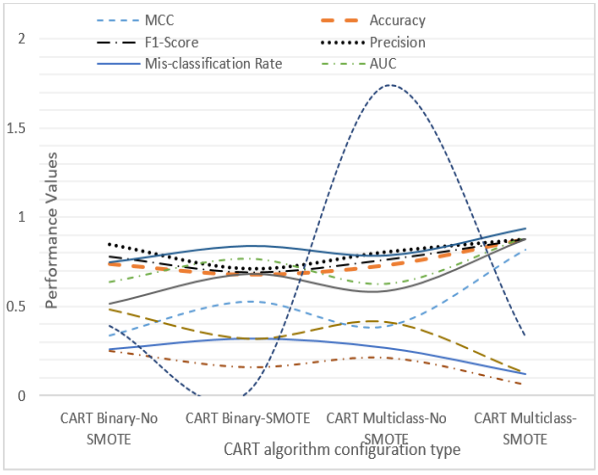

Figure 10: Decision trees (CART) performance results for different configurations.

NO-SMOTE on the dataset, presented an interesting experiment as it made good on a promise for the research under study. The algorithm showed the best performance for both the binary and multiclass algorithm configuration capabilities for the multiclass classification problem at hand. It performed as the best overall experiment for the cyber threat predictive model for MMS applicant cyber threats or fraud intent detection and prevention. The results are as shown in Table 10, Figures 11 and 12. The results are described as follows:

(i.) Accuracy and MCC: RF performed the best overall in all the simulation experiments and was recommended for predicting mobile money cyber threat detection. The algorithm had an accuracy of 0.91 and an MCC of 0.88 for the multiclass

configuration, with a minimal classification error rate of 0.09. These are the highest in the overall simulation experiments conducted.

(ii.) Precision, recall and F1-score: The high-performance result for the RF algorithm for the predictive model was also shown by the high multiclass configuration sensitivity and specificity of 0.90 and 0.95, respectively, as well as the F1-Score of 0.91 in the overall experiments.

(iii.) ROC and predicted probabilities: The ROC AUC for RF was 0.99 for multiclass configurations. This is also confirmation that the best algorithm for the Mobile Money customer onboarding predictive model is the RF classifier.

4.3 Result analysis by algorithm performance comparison

Comparing all the predictive models and simulation experiments in order of performance metrics, as shown in Table 11 and Figures 13 and 14, the best algorithm was Random Forest. The multiclass configuration of the algorithm with SMOTE performed overall best, while the binary configuration with SMOTE came in second.

The RF with SMOTE has the overall highest MCC of 0.88, accuracy of 0.91, precision of 0.93, the lowest classification error of 0.09, a ROC AUC of 0.99, specificity or true negative rate (TNR) of 0.95 (95%), and sensitivity or recall (TPR) of 0.90 (90%), while the binary configuration has an MCC of 0.86, accuracy of 0.90, precision of 0.92, the lowest classification error of 0.10, a ROC AUC of 0.98, specificity (TNR) of 0.95 (95%), and sensitivity or recall (TPR) of 0.90 (90%). However, the run duration for binary configuration was faster, with a total time of 0.18 min, than the multiclass configuration duration of 0.40 min.

The implication of the narrow differences in performance metrics between the multiclass and binary configurations with SMOTE reinforced, confirmed, and showed that Random Forest (RF) is a default multiclass classifier as it was able to predict the cyber threat risk levels into multiple classes according to dataset labels.

5 Conclusion

In order to prevent mobile money fraud and deal with anti-money laundering compliance, machine learning (ML) and artificial intelligence (AI) are becoming more and more widely accepted as essential. The fight against financial crime has always involved computational technology, but the development of ML and AI has given law enforcement a potent new weapon in the fight against mobile money fraud. Financial institutions can better understand their customers' demands and risk profiles by using AI to spot and highlight problematic conduct, such as large or unexpected transactions. Financial institutions may greatly enhance their capacity to prevent mobile money fraud and handle money laundering issues by leveraging the power of AI. This study attempts to develop a fraud detection model that will identify warning signs of fraud and money laundering in mobile money transfers using ML algorithms. More specifically, a collection of risk-based indicators was employed in this study to forecast the likelihood that a transaction would be fraudulent. SMOTE techniques were used to create artificial minority class samples in order to prevent dataset sub-structures that were either uninformative or poorly informative.

This work significantly contributes to the body of knowledge on how to detect suspicious activity in mobile money transfers in a number of ways. Theoretically, machine learning algorithms that rely on the more traditional rule-based benchmark methodology can get around the difficulties associated with trying to identify illicit transactions. The traditional rule-based benchmark technique uses established criteria based on mathematical circumstances to identify illicit transactions.

Table 10: Performance Metrics for Random Forest with (SMOTE)/without SMOTE (No-SMOTE)

Random Forest(RF)	MCC	Accuracy	F1-Score	Precision	Misclassification Rate	AUC	TNR (Specificity)	FPR	TPR (Sensitivity)	FNR	Runtime (min)
RF Binary-No SMOTE	0.50	0.79	0.86	0.96	0.21	0.78	0.77	0.23	0.56	0.44	0.53
RF Binary-SMOTE	0.86	0.90	0.90	0.92	0.10	0.98	0.95	0.05	0.90	0.10	0.18
RF Multiclass-No SMOTE	0.51	0.79	0.86	0.96	0.21	0.78	0.77	0.23	0.56	0.44	1.7
RF Multiclass-SMOTE	0.88	0.91	0.91	0.93	0.09	0.99	0.95	0.05	0.90	0.10	0.4

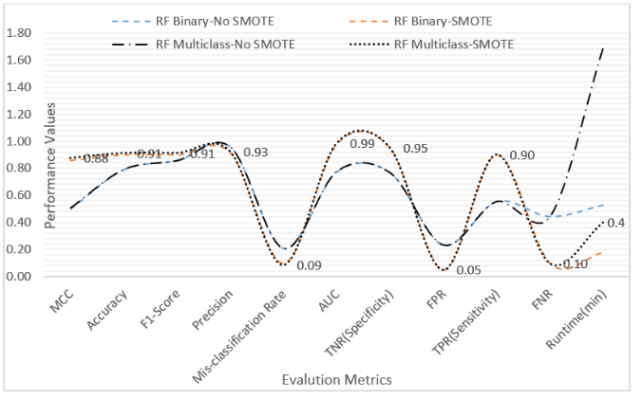

Figure 11: Different random forest algorithm configuration results by performance parameters.

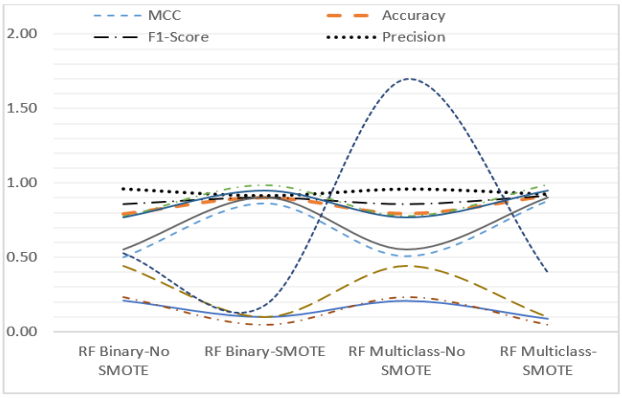

Figure 12: Random Forest performance results for different configurations

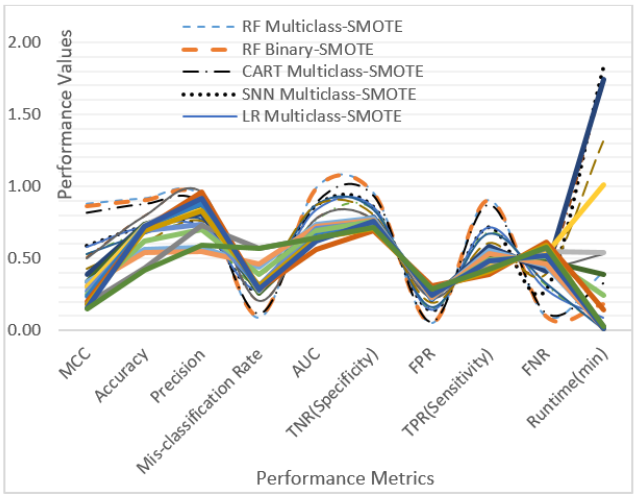

Figure 13: Different experimented algorithm configuration results by performance parameters

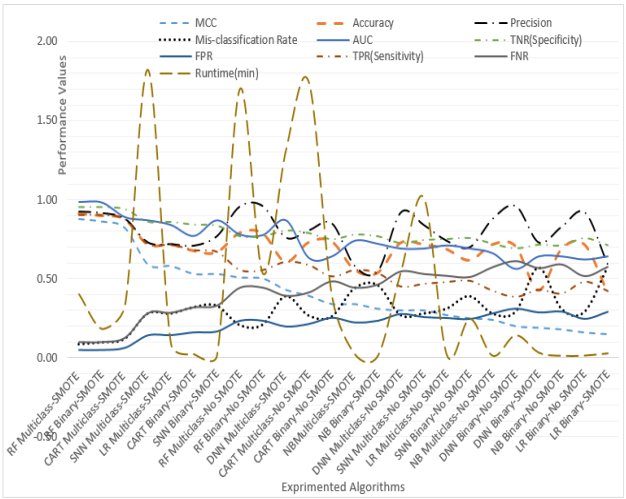

Figure 14: All algorithm performance results for different configurations.

Table 11: Ranking algorithm experimental performance with (SMOTE)/without SMOTE (No-SMOTE)

Algorithm	MCC	Accuracy	Precision	Misclassification Rate	AUC	TNR (Specificity)	FPR	TPR (Sensitivity)	FNR	Runtime (min)
RF Multiclass-SMOTE	0.88	0.91	0.93	0.09	0.99	0.95	0.05	0.90	0.10	0.40
RF Binary-SMOTE	0.86	0.90	0.92	0.10	0.98	0.95	0.05	0.90	0.10	0.18
CART Multiclass-SMOTE	0.82	0.88	0.88	0.12	0.89	0.94	0.06	0.88	0.12	0.33
SNN Multiclass-SMOTE	0.59	0.72	0.73	0.28	0.87	0.86	0.14	0.72	0.28	1.82
LR Multiclass-SMOTE	0.58	0.72	0.72	0.28	0.84	0.86	0.14	0.72	0.28	0.09
CART Binary-SMOTE	0.53	0.68	0.71	0.32	0.77	0.84	0.16	0.68	0.32	0.02
SNN Binary-SMOTE	0.53	0.67	0.77	0.33	0.87	0.83	0.17	0.67	0.33	0.01

RF Multiclass-No SMOTE	0.51	0.79	0.96	0.21	0.78	0.77	0.23	0.56	0.44	1.70
RF Binary-No SMOTE	0.50	0.79	0.96	0.21	0.78	0.77	0.23	0.56	0.44	0.53
DNN Multiclass-SMOTE	0.43	0.61	0.76	0.39	0.87	0.80	0.20	0.61	0.39	1.31
CART Multiclass-No SMOTE	0.392	0.733	0.807	0.267	0.63	0.79	0.21	0.59	0.41	1.74
CART Binary-No SMOTE	0.34	0.74	0.85	0.26	0.64	0.75	0.25	0.51	0.49	0.39
NB Multiclass-SMOTE	0.34	0.56	0.58	0.44	0.74	0.78	0.22	0.56	0.44	0.02
NB Binary-SMOTE	0.31	0.54	0.55	0.46	0.72	0.77	0.23	0.54	0.46	0.01
DNN Multiclass-No SMOTE	0.3	0.73	0.92	0.27	0.69	0.72	0.28	0.45	0.55	0.54
SNN Multiclass-No SMOTE	0.3	0.72	0.84	0.28	0.69	0.74	0.26	0.47	0.53	1.01
LR Multiclass-No SMOTE	0.27	0.69	0.74	0.31	0.71	0.75	0.25	0.48	0.52	0.02
SNN Binary-No SMOTE	0.25	0.62	0.7	0.39	0.69	0.76	0.24	0.49	0.51	0.24
NB Multiclass-No SMOTE	0.24	0.72	0.88	0.28	0.66	0.72	0.28	0.42	0.58	0.01
DNN Binary-No SMOTE	0.2	0.71	0.96	0.29	0.56	0.69	0.31	0.39	0.61	0.14
DNN Binary-SMOTE	0.19	0.43	0.73	0.57	0.64	0.72	0.28	0.43	0.57	0.03
NB Binary-No SMOTE	0.18	0.69	0.83	0.31	0.64	0.71	0.29	0.41	0.59	0.01
LR Binary-No SMOTE	0.16	0.72	0.92	0.29	0.62	0.76	0.24	0.48	0.52	0.01
LR Binary-SMOTE	0.15	0.42	0.59	0.57	0.64	0.71	0.29	0.42	0.58	0.03

Acknowledgment

This Research was funded by the TETFund Research Fund and Africa Centre of Excellence OAK-Park.

References

[1] Sam Castle, Pervaiz Fahad, Cassebeer Weld Galen, Roesner Franziska and Richard J. Anderson. Let's Talk Money: Evaluating the Security Challenges of Mobile Money in the Developing World. In Proceedings of the 7th Annual Symposium on Computing for Development (ACM DEV '16, 18 – 20 November 2016, Nairobi, Kenya, 1-10, 2016. https://doi.org/10.1145/3001913.3001919

[2] Alex Bara. Mobile money for financial inclusion: policy and regulatory perspective in Zimbabwe, African Journal of Science, Technology, Innovation and Development, 5(5): 345-354, 2013 https://doi.org/10.1080/20421338.2013.829287

[3] Sandra L, Suárez. Poor people's money: the politics of mobile money in Mexico and Kenya, Telecommunications Policy, 40 (10/11): 945-955, 2016. https://doi.org/10.1016/j.telpol.2016.03.001 Frederick Kanobe, Patricia M. Alexander, and Kelvin J. Bwalya. Policies, regulations and procedures and their effects on mobile money systems in Uganda, The Electronic Journal of Information Systems in Developing Countries, 83(1):1-15, 2017. https://doi.org/10.1002/j.1681-4835.2017.tb00615.x

[4] Isaac Akomea-Frimpong, Charles Andoh, Agnes Akomea-Frimpong, Yvonne Dwomoh-Okudzeto. Control of fraud on mobile money services in Ghana: an exploratory study, Journal of Money Laundering Control, 22(2):300-317, 2019. https://doi.org/10.1108/JMLC-03-2018-0023

[5] Claire Célerier and Adrien Matray. Bank-branch supply, financial inclusion, and wealth accumulation. Rev. Finan. Stud, 32(12):4767–4809, 2019. https://doi.org/10.1093/rfs/hhz046

[6] M. Mostak Ahamed and Mallick Sushanta. Is financial inclusion good for bank stability?. Journal of Economic Behavior & Organization, 157(1): 403–427, 2019. https://doi.org/10.1016/j.jebo.2017.07.027

[7] Joyce Koi Akrofi. Mobile Money Adoption in Africa: A Literature-Based Analysis, Texila

International Journal of Management, 8(2): 1-12, 2022. https://doi.org/10.21522/TIJMG.2015.08.02.Art014

[8] Pierre-Laurent Chatain, Andrew Zerzan, Wameek Noor, Najah Dannaoui, and Louis de Koker. Protecting Mobile Money against Financial Crimes: Global Policy Challenges and Solutions, World Bank Publications, New York, NY, 2011. https://doi.org/10.1596/978-0-8213-8669-9

[9] Mercy W. Buku and Rafe Mazer. Fraud in Mobile Financial Services: Protecting Consumers, Providers, and the System. World bank publications, 2017. https://documents1.worldbank.org/curated/en/249151504766545101/pdf/119208-BRI-PUBLIC-Brief-Fraud-in-Mobile-Financial-Services-April-2017.pdf

[10] Hakeem J. Pallangyo. Cyber Security Challenges, its Emerging Trends on Latest Information and Communication Technology and Cyber Crime in Mobile Money Transaction Services, Tanzania Journal of Engineering and Technology, 41(2):189-204, 2022. http://dx.doi.org/10.52339/tjet.v41i2.792.

[11] Denish Azamuke, Marriette Katarahweire and Engineer Bainomugisha. Scenario-based Synthetic Dataset Generation for Mobile Money Transactions. In Proceedings of the Federated Africa and Middle East Conference on Software Engineering, 64–72, 2022. https://doi.org/10.1145/3531056.3542774

[12] Mistura Laide Sanni., Bodunde Odunola Akinyemi**,** Dauda Akinwuyi Olalere, Emmanuel Adebayo Olajubu and Ganiyu Adesola Aderounmu. A Predictive Cyber Threat Model for Mobile Money Service. Annals of Emerging Technologies in Computing, 7(1): 40-60, 2023. https://doi.org/10.33166/AETiC.2023.01.004

[13] Stephen Ambore, Christopher Richardson, Huseyin Dogan, Edward Apeh and David Osselton. A resilient cybersecurity framework for Mobile Financial Services (MFS). Journal of Cyber Security Technology**,** 1(3-4): 202-224, 2017.

https://doi.org/10.1080/23742917.2017.1386483

[14] Maria Zhdanova, Jürgen Repp, Roland Rieke, Chrystel Gaber and Baptiste Hemery. No Smurfs: Revealing Fraud Chains in Mobile Money Transfers. In proceedings of the International Conference on Availability, Reliability and Security (ARES), Switzerland, 10, 2014. https://doi.org/10.1109/ARES.2014.10

[15] Francis Effirim Botchey, Zhen Qin, and Kwesi Hughes-Lartey. Mobile Money Fraud Prediction—A Cross-Case Analysis on the Efficiency of Support Vector Machines, Gradient Boosted Decision Trees, and Naïve Bayes Algorithms. Information, 11(8): 383, 2020. https://doi.org/10.3390/info11080383

[16] Ibukun Eweoya, Ayodele Adebiyi, Ambrose Azeta, Okesola Olatunji. Fraud Prediction in Bank Credit Administration: A Systematic Literature Review. Journal of Theoretical and Applied Information Technology, 97 (11): 3147-3169, 2019.

[17] Boluwaji A. Akinnuwesi, Stephen G. Fashoto, Andile S. Metfula and Adetutu N. Akinnuwesi. Experimental Application of Machine Learning on Financial Inclusion Data for Governance in Eswatini. In: Hattingh, M., Matthee, M., Smuts, H., Pappas, I., Dwivedi, Y.K., Mäntymäki, M. (eds) Responsible Design, Implementation and Use of Information and Communication Technology. I3E 2020. Lecture Notes in Computer Science, 12067. Springer, Cham, 2020. https://doi.org/10.1007/978-3-030-45002-1_36

[18] Francis Effirim Botchey, Zhen Qin, Kwesi Hughes-Lartey and Ernest Kwame Ampomah. Predicting Fraud in Mobile Money Transactions using Machine Learning: The Effects of Sampling Techniques on the Imbalanced Dataset. Informatica, 45: 45–56, 2021. https://doi.org/10.31449/inf.v45i7.3179.

[19] Majda Omer Albasheer and Eihab B. M. Bashier. Enhanced Model for PKI Certificate Validation in the Mobile Banking. in Proceedings of the 2013 International Conference on Computing, Electrical and Electronics Engineering (ICCEEE), 26-28 August 2013, Khartoum, Sudan, 470–476, 2013. https://doi.org/10.1109/ICCEEE.2013.6633984

[20] Shaik Shakeel Ahamad, V. N. Sastry, and Madhusoodhnan Nair. Biometric Based Secure Mobile Payment Framework. in Proceedings 2013 4th International Conference on Computer and Communication Technology (ICCCT), 20-22 September 2013, Allahabad, India, 239-246, 2013. https://doi.org/10.1109/ICCCT.2013.6749634

[21] C. Narendiran, S. Albert Rabara, and N. Rajendran. Public Key Infrastructure for Mobile Banking Security. in Proceedings of the 2009 Global Mobile Congress, 12-14 October 2009, Shanghai, China, 1–6, 2009. https://doi.org/10.1109/GMC.2009.5295898

[22] Mangala Belkhede, Veena Gulhane, and Preeti Bajaj. Biometric Mechanism for Enhanced

Security of Online Transaction on Android System: A Design Approach, in Proceedings of the 2012 14th International Conference on Advanced Communication Technology (ICACT), 19-22 February 2012, PyeongChang, South Korea, 1193 – 1197, 2012.

[23] Min Hee Yeon, Park Jin Hyunga and Kim In Seok. Outlier Detection Method for Mobile Banking with User Input Pattern and E-finance Transaction Pattern. Journal of Internet Computing and Services, 15(1):157–170, 2014. https://doi.org/10.7472/JKSII.2014.15.1.157

[24] Adam B. Mtaho. Improving Mobile Money Security with Two-Factor Authentication. International Journal of Computer Applications, 109(7): 9-15, 2015, https://doi.org/10.5120/19198-0826

[25] Peter Tobbin and John K.M. Kuwornu. Adoption of Mobile Money Transfer Technology: Structural Equation Modelling Approach. European Journal of Business and Management, 3(7):59–77. 2011.

[26] Adeyinka Adedoyin, Stelios Kapetanakis, Georgios Samakovitis and Miltos Petridis. Predicting fraud in mobile money transfer using case-based reasoning. In proceedings of the Artificial Intelligence XXXIV: 37th SGAI International Conference on Artificial Intelligence, AI 2017, Cambridge, UK, December 12-14, 2017. https://doi.org/10.1007/978-3-319-71078-5_28.

[27] Simon Delecourt and Li Guo. Building a Robust Mobile Payment Fraud Detection System with Adversarial Examples. In proceedings of the 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), 03-05 June 2019. https://doi.org/10.1109/AIKE.2019.00026.

[28] Md. Alamgir Hossain. Security Perception in the Adoption of Mobile Payment and the Moderating Effect of Gender, PSU Research Review, 3(3):179-190, 2019. https://doi.org/10.1108/PRR-03-2019-0006

[29] Suleiman Ali Alsaif and Adel Hidri. Impact of data balancing during training for best predictions, Informatica, 45(2): 223–230, 2021. https://doi.org/10.31449/inf.v45i2.3479.

[30] Ibtissam Benchaji, Samira Douzi, and Bouabid ElOuahidi. Using Genetic Algorithm to Improve Classification of Imbalanced Datasets for Credit Card Fraud Detection, in Proceedings of the 2018 2nd Cyber Security in Networking Conference (CSNet), Paris, France, 1-5, 2018. https://doi.org/10.1109/CSNET.2018.8602972

[31] Bashir S., and Ghous H., Detecting Mobile Money Laundering Using Genetic Algorithm as Feature Selection Method with Classification Method. LC International Journal of STEM, 1(4):121-129, 2021. https://doi.org/10.5281/zenodo.5149794

[32] Shamila Bashir, Dr. Hamid ur Rehman. Detecting Mobile Money Laundering Using KPCA as Feature Selection Method. LC International Journal of STEM, 2(3):1-8, 2021. https://doi.org/10.5281/zenodo.5751721

[33] John O. Awoyemi; Adebayo O. Adetunmbi; Samuel A. Oluwadare. Credit card fraud detection using machine learning techniques: A comparative analysis, in proceedings of the IEEE 2017 International Conference on Computing Networking and Informatics (ICCNI). 1–9, 2017. https://doi.org/10.1109/ICCNI.2017.8123782

[34] Varmedja, D., Karanovic, M. Sladojevic, S., Arsenovic, M., and Anderla, A. Credit card fraud detection-machine learning methods, In proceedings of the IEEE 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH), 1–5, 2019. https://doi.org/10.1109/infoteh.2019.8717766

[35] Nana Kwame Gyamfi, and Jamal-Deen Abdulai. Bank Fraud Detection Using Support Vector Machine, in proceedings of the 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 37-41, 2018 https://doi.org/10.1109/IEMCON.2018.8614994 .

[36] Dongfang Zhang, Basu Bhandari, and Dennis Black. Credit Card Fraud Detection Using Weighted Support Vector Machine. Applied Mathematics, 11, 1275-1291, 2020. https://doi.org/10.4236/am.2020.1112087

[37] Olawale Adepoju, Julius Wosowei, Shiwani lawte and Hemaint Jaiman. Comparative Evaluation of Credit Card Fraud Detection Using Machine Learning Techniques, In proceedings of the 2019 Global Conference for Advancement in Technology (GCAT), Bangalore, India, 1-6, 2019. https://doi.org/10.1109/GCAT47503.2019.8978372.

[38] Samidha Khatri, Aishwarya Arora, and Arun Prakash Agrawal. Supervised Machine Learning Algorithms for Credit Card Fraud Detection: A Comparison, In proceedings of the 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 680-683, 2020. https://doi.org/10.1109/Confluence47617.2020.9057851.

[39] Iddi S. Mambina, Jema D. Ndibwile, and Kisangiri F. Michael. Classifying Swahili Smishing Attacks for Mobile Money Users: A Machine-Learning Approach, IEEE Access, 10, 83061-83074, 2022. https://doi.org/10.1109/ACCESS.2022.3196464

[40] N. NishaBalani, MeherBhawnani M., and AnkitaKamle M. Implementation and Design on Fraud Detection and Prediction of Mobile Money Transaction Using ML Techniques. Annals of the Romanian Society for Cell Biology, 24(2):261 – 269, 2021.

[41] Evgenia Novikova and Igor Kotenko. Visual Analytics for Detecting Anomalous Activity in Mobile Money Transfer Services. In Proceedings of the International Cross-Domain Conference and Workshop on Availability, Reliability, and Security (CD-ARES), Sep, Fribourg, Switzerland. 63-78, 2014. https://doi.org/10.1007/978-3-319-10975-6_5

[42] Muhammad R. Khan and Joshua E. Blumenstock. Predictors without borders: behavioral modeling of product adoption in three developing countries. In Proceedings of the ACM 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 145–154, 2016. http://dx.doi.org/10.1145/2939672.2939710

[43] Simone Centellegher, Giovanna Miritello, Daniel Villatoro, Devyani Parameshwar, Bruno Lepri and Nuria Oliver. In Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies. 2(7):1–18, 2018. https://doi.org/10.1145/3287035.