Financial Fraud Detection Using Rich Mobile Money Transaction Datasets

Denish Azamuke(✉), Marriette Katarahweire, and Engineer Bainomugisha

Makerere University, Pool Road, Kampala, Uganda
denishazamuke@gmail.com, baino@mak.ac.ug
https://cs.mak.ac.ug/

Abstract. In an era marked by the rise of digital transactions, mobile money platforms continue to experience rampant fraud and thus effective fraud detection approaches are key for maintaining the integrity of financial systems, especially in the Sub-Saharan region. This study simulates known fraudulent scenarios found in mobile money platforms in Sub-Saharan Africa using a multi-agent-based simulation platform called MoMTSim. MoMTSim generates rich synthetic mobile money transaction datasets that are statistically close to the real mobile money transaction data. The study examines common classification models including Logistic regression, Gradient boosting, Decision trees, AdaBoost, XGBoost, and Random forest for financial fraud detection. The models were evaluated using several performance metrics including Precision, Recall, F1-score, AUC-ROC, and notably, the Matthews correlation coefficient (MCC), which is particularly effective for imbalanced classes common in financial data. The results demonstrate that all tested models are capable of identifying fraudulent transactions, with varying degrees of success. The XGBoost model stood out with the highest MCC (0.82) and AUC of 0.97, indicating superior overall performance. Meanwhile, the Logistic regression model served as a benchmark with an MCC of 0.67, revealing the performance enhancements offered by more complex models. However, the study also underscores the importance of considering the computational costs associated with more complex models. The findings affirm the potential of machine learning algorithms for fraud detection and provide valuable insights into model selection based on performance and computational requirements.

Keywords: Mobile money transactions · Simulation · Agent-based modelling · Fraud detection · Machine learning

1 Introduction

The challenge of increasing financial fraud in mobile money transactions in the Sub-Saharan region has numerous consequences on the economy and existing programmes that aim to promote financial inclusion. Mobile money technology

A. Sere et al. (Eds.): AFRICOMM 2023, LNICST 588, pp. 190–208, 2025.
https://doi.org/10.1007/978-3-031-81573-7_16

has been leveraged on many occasions for instance during the Covid-19 pandemic [17,20], to disburse relief cash to vulnerable communities. The service providers (telco operators) and financial institutions are at the centre of securing transactions happening on their platforms with guidance and regulation from the central banks. The service providers mostly rely on rule-based expert systems to detect incidences of financial fraud [6,18]. The challenge with that approach is the resulting high false positive rates due to the ineffectiveness of the rules on complex fraud patterns. Also with the dynamic nature of fraud in financial systems whereby fraudsters tend to be more adaptive than the service providers, controls need to be adjusted to suit this behaviour otherwise the race becomes unfair [5,24,26]. The changing patterns of fraud render historical data kept by the service providers obsolete for financial fraud detection even if the researcher is able to access the dataset. The financial records for mobile money transactions are very sensitive and often kept private denying the chance for outside researchers to participate in offering solutions to the fraud challenges. Besides, no diverse categorised fraud scenarios can be found which would inform the tuning of the existing financial fraud controls as well as the opportunity to develop better fraud detection techniques using computational methods [24].

Machine learning algorithms composed of Logistic regression, Random forest and Decision trees can be trained on data with labelled instances of financial fraud to detect future occurrences of the crime including complex fraud patterns. Such endeavour requires diverse, well-labelled data that is often difficult to obtain and at the same time, the data should be rich enough in terms of fraud cases for the intended tasks [24]. Owing to the intrinsically private nature of mobile money financial datasets and the class imbalances in the real datasets, this study generates diverse synthetic mobile money transaction datasets using a financial simulation platform [30]. MoMTSim is designed and calibrated based on real transaction data and its outputs are evaluated using the sum of squared errors (SSE) method by computing the difference between the real and synthetic data. With the agent-based modelling techniques used in its development, this study leverages simulation to model known fraudulent behaviours from the real ecosystem to enrich the synthetic datasets by defining specific fraud parameters in the model. Using the rich synthetic transaction datasets, this study performs financial fraud detection using common machine learning algorithms and evaluates the efficacy of the models in identifying unique fraudulent patterns in mobile money transactions.

1.1 Unique Fraudulent Behaviours in Mobile Money Transactions

Split Deposit Fraud. Split deposit fraud involves the mobile money merchant who acts as an intermediary between the customer and the mobile money system, facilitating the conversion of hard cash into electronic money and vice-versa. In this scenario, a dishonest mobile money merchant splits cash deposits into the client's account in the form of small chucks to enable earning of higher commission because of the many deposits made. These transactions happen in short time intervals involving a particular mobile money account. This fraudulent activity

reduces the revenue of the service provider and large sums of money are lost when many mobile money merchants take part in it. Even though some service providers tried to put a rule-based approach of a time frame to isolate these transactions, a number of the practitioners quickly learned about the measure and adapted in terms of the schedules to commit fraud [5,31].

Refund Fraud. This scenario is commonly practised by mobile money clients (end-users of the service). It involves the fraudster making a payment for goods or services using their mobile money account. Then a refund or reversal is requested leading to a transfer transaction that is fraudulent. The fraudster keeps track of merchants that easily fall for this kind of fraud and aims to carry out as many transactions as possible including with potential new victims and eventually withdraws their gains out of the mobile money system [5,31].

2 Simulation and Fraud Detection Approaches

The use of simulation for fraud detection research has been presented by several studies [23–26]. The work in this paper expands on the capabilities of simulation using agent-based modelling techniques to develop models of current fraudulent tactics in mobile money services. The efforts in our study mainly focus on unique fraud schemes that are present in the Sub-Saharan context. Documented fraud scenarios by related studies [5,31] form the basis for modelling the unique fraud patterns in the real mobile money ecosystem.

2.1 Approaches for Financial Fraud Detection

Deterministic, Rule-Based Approach. Rule-based fraud detection is one of the common approaches used in low-resource settings. It is concerned with predefined transactional rules that are usually set by the service provider or financial institution in order to identify potentially fraudulent transactions [2,36]. It requires historical data that is usually available in financial institutions, expert knowledge and regulatory requirements often issued by designated regulatory authorities. This approach is widely used by financial institutions in Sub-Saharan Africa because of the ease of implementation, being relatively straightforward to understand and it can quickly identify basic fraud schemes. Moreover, this approach is often compliant with industry regulations and best practices that are at the forefront of the operations of financial institutions.

However, the rule-based approach is prone to producing many false positives in the event the rules are very strict or many false negatives when the rules are lenient. Too many false positives discourage the usage of financial services among customers. Financial institutions suffer from fraudsters who are usually very adaptive and since the rule-based approaches are manual, the systems can hardly adapt, and require expert knowledge to tune them. This, therefore, renders the rule-based approach very ineffective against evolving fraud schemes [1].

Scenario-Based and Risk-Weighted Approaches. This approach involves the formulation of scenarios that represent potential fraud patterns in the financial ecosystem. With this approach, more complex financial fraud patterns can be identified as compared to rule-based expert systems given that the scenarios serve the purpose of analyzing transactions and detecting suspicious activities fitting the patterns. Also, this approach can potentially uncover emerging fraud schemes [35].

Even though the scenario-based approach promotes pro-activeness in financial fraud detection, the development of accurate and relevant scenarios that suit the context necessitates significant domain expertise. Unknown fraud schemes can still be missed and updating scenarios is largely time-consuming [11,35].

The risk-weighted approach aims to assign a risk score to every transaction based on transaction amount, frequency, and location among other things and thus flag financial transactions associated with higher risk scores for further investigations. This allows for the prioritisation of resources based on the level of risk corresponding to a transaction which makes it more flexible than rule-based and scenario-based approaches. Oftentimes, financial institutions have specific risk appetite and tolerance levels, thus this approach can be customized for specific needs [44].

Depending on the accuracy of the risk model that has been put in place, the approach can still produce false positives or false negatives. Besides it requires expertise to develop and sustain risk-scoring models given that they can be complex, and further investigations are ultimately resource-intensive [14,43,44].

Machine Learning (ML) Approach. ML-based fraud detection is concerned with algorithms capable of learning from historical financial data in order to identify patterns and anomalies that are reflective of fraudulent activities. These algorithms include; Random forest [7,9,15], Logistic regression [15], XGBoost, Gradient boosting, AdaBoost and Decision trees [22]. They can learn human behaviour [10,42] and detect new and evolving fraud patterns in financial transactions. With the use of simulation, rich synthetic outputs can easily be used to train a number of ML algorithms for fraud detection. The researcher might not spend time for instance to clean the data since the simulations can be performed with contextual relevance to the task of financial fraud detection. In regard to other approaches, ML algorithms are more effective in dealing with large and complex datasets and they require minimal effort to maintain [3,33].

However, these algorithms may suffer over-fitting and thus isolate specific fraud patterns limiting their scope to detect different fraud schemes. Also, large amounts of data are required in model training and in the event of no historical data, the approach might not be feasible unless the financial institution can invest in synthetic data generation [39]. Financial institutions consider a number of factors before they commit themselves to a given approach for fraud detection. Some of these considerations include the volume of data that is to be processed, the cost of implementing a fraud detection measure and its maintenance, the

level of expertise required and the desired level of accuracy and tolerance in a given regulatory environment.

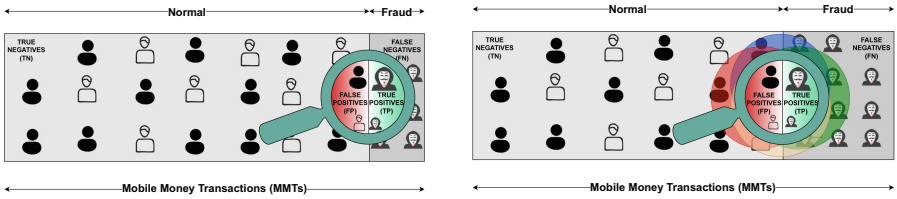
Our study combines the simulation of contemporary fraudulent schemes in mobile money services with novel computational methods consisting of common ML classifiers for automated fraud detection. Besides, it addresses the challenge of class imbalance in real data through the generation of synthetic financial data that statistically resembles real transaction data.

3 The Challenge of Class Imbalance in Financial Data for Fraud Detection

Mobile money service providers and financial institutions face challenges with enriching their own data for effective fraud detection using machine learning techniques. At times, better algorithms need to be developed for efficient fraud detection with low false positives. The use of simulation to generate financial data with known fraud instances has been a major breakthrough for the financial industry. Simulation environments that are agent-based have yielded significant results in this domain [24,26]. For instance, Lopez-Rojas et al. [26] simulate a known financial crime pattern in the mobile money financial domain in order to generate fraudulent behaviour for fraud detection. Real financial datasets are highly imbalanced in nature and this makes detection of complex fraud patterns extremely difficult. Most studies in this domain rely on the use of the synthetic minority over-sampling technique (SMOTE) [8,27] to resolve the class imbalance in the real financial data with no consideration for the quality of the synthetic samples [4,19,37,40]. Simulation plays a crucial role in addressing this challenge given that the documented fraudulent behaviours in mobile money systems are accessible to the research community.

Figure 1a represents a typical real mobile money transaction data residing in the data warehouse of a service provider in the Sub-Saharan region. The rectangular block is a collection of genuine and very few fraudulent transactions. Clearly, the dataset exhibits a high-class imbalance for fraud detection using machine learning techniques. This challenge has made many of the service providers in the region use rule-based approaches that often result in many false positives even though they are easy to set up. Complex fraud patterns have been hardly studied and fraudsters are always adapting to the relatively straightforward control measures set by the service providers [26].

Figure 1b shows synthetic mobile money transaction data generated using agent-based modelling techniques with a sufficient amount of fraudulent transactions based on fraudulent behaviours in the real ecosystem. This approach ultimately solves the high-class imbalance problem in financial datasets as well as the obsoleteness of the historical data for studying new fraud scenarios. Simulation allows tuning of existing financial crime controls by modelling anticipated fraud activity. Besides, it ensures calibration of fraud control systems in order to enable them to adapt to emerging fraudulent behaviours, and the changing regulatory dynamics [26].



(a) Data with high-class imbalance. (b) Data with sufficient fraud.

Fig. 1. Mobile money transaction datasets for financial fraud detection

4 Methods

4.1 Financial Fraud Simulation Using MoMTSim

MoMTSim [30] is a multi-agent-based simulation (MABS) platform designed and calibrated using real mobile money transaction data to output diverse synthetic financial data. The core model in MoMTSim represents interactions of agents including clients and mobile money merchants based on probabilities extracted from the real data. A client has a profile, starting balance and other files containing properties of mobile money transactions; transaction types and aggregated transactions that are used during the simulations. Clients also participate in future transactions based on probabilities and their states are adjustable during a simulation. The entire simulation model is similar to a Markov process and agents carry common transactions that are present in the real mobile money ecosystem. The transaction types modelled include a deposit which is concerned with a client loading electronic money into their account via a mobile money merchant. A withdrawal is the opposite of a deposit, debit involves moving money from a mobile money account to a bank account. A transfer is concerned with the movement of electronic funds from one mobile money account to another account, while a payment includes the purchase of goods and services using electronic money in a mobile money account.

The object code implementation for the MoMTSim simulation platform uses a generic agent-based simulation toolkit MASON [13,28,29] which is fast enough and capable of handling large custom simulations. Besides MASON is multi-platform, supports parallelism and is capable of reinforcing computationally expensive simulations unlike NetLogo, Repast and AnyLogic [23,26]. A step in the simulation represents an hour in the real mobile money ecosystem(real world).

Fraud modelling in MoMTSim was carried out by defining specific fraud parameters for the fraud schemes discussed in Subsect. 1.1. Besides, transaction rules based on the fraudulent behaviours were defined in MoMTSim and a fraudulent client carries transactions in parallel with normal clients during a simulation. These transactions are contingent on probabilities of committing fraud, fraudster finding new victims and the chances of previous victims being at high risk for future fraud. We scheduled the fraudsters to fiercely carry out

transactions by specifying higher probabilities of committing fraud. This allowed the generation of sufficient instances of fraudulent transactions in order to obtain rich synthetic datasets for fraud detection. Upon completion of all interactions in the simulation platform, diverse synthetic transaction files were written together with the parameter history and other log files as output.

Calibration and Validation of Simulations. Calibration of simulation parameters was aimed at making sure that agents do not exhibit unusual behaviours [16,32]. Calibration also focused on avoiding the normative behaviour of agents since with agent-based modelling, various entities were modelled based on specific characteristics as observed from the real ecosystem. During this process, parameter sets leading to behaviours not present in the real ecosystem were removed. Documented practices guided by expert opinions and our understanding of the mobile money ecosystem were used to verify agent behaviours in simulations and statistical methods were used to assess the closeness of the synthetic data to the real data.

The conformity of the synthetic transaction datasets to the real data was measured using the sum of squared errors (SSE) method. We computed the difference between the real and synthetic data and the dataset with the least total error was selected for financial fraud detection using machine learning classifiers [26]. Other synthetic datasets that were not used for fraud classification still registered relatively low total errors implying they could as well be used for the same task.

4.2 Data Description, Cleaning and Preprocessing

With MoMTSim [30], we simulated 1,040,000 rows of mobile money transactions enriched using fraud schemes in Subsect. 1.1. The dataset contained 768,248 legitimate transactions while 271,752 were fraudulent transactions. The features in the synthetic data were based on those found in the real data with an addition of the target variable (label for fraud) in order to facilitate financial fraud detection using machine learning algorithms. The independent features and the dependent variable in the data are presented in Table 1. Unlike real data, incidences of missing data points in the synthetic data were eliminated during the design of the financial simulation platform. This implies that the traditional approaches for data cleaning do not apply to the resulting datasets which is unarguably one of the advantages of working with well-labeled synthetic datasets.

The identifiers for the client starting a transaction, *startingClient* and the recipient, *destinationClient* were removed prior to model building as they did not possess attributes that would affect fraud detection results. A common challenge with real financial data is the high-class imbalance, usually, researchers adopt for instance the SMOTE [8,27] to up-sample the minority class. However, the use of simulation addressed this challenge by generating sufficient instances of fraudulent transactions (see section 3) to enrich the data for financial fraud detection.

Moreover, the simulated data used in our study contained 26.13% of fraudulent transactions making the synthetic data rich enough for fraud classification. The *transactionType* is categorical; deposit, withdrawal, transfer, payment and debit, and it was one-hot encoded in order to allow the machine learning models to utilise the feature as well as to improve model predictions.

Table 1. The independent features and the dependent variable in the synthetic mobile money transaction data

Feature/Variable	Description	Measure
step	This maps a unit of time, a step in the simulation is an equivalent of one hour in the real world	Continuous
transactionType	Includes deposit, withdrawal, transfer, debit, and payment	Categorical
amount	Funds associated with a transaction type	Continuous
startingClient	Mobile money customer who initiates a transaction	Continuous
oldBalStartingClient	The starting balance of the client before initiating a transaction	Continuous
newBalStartingClient	The new balance of a client after initiating a transaction	Continuous
destinationClient	The recipient of funds after a transaction has taken place	Continuous
oldBalDestinationClient	The initial balance of the recipient client before a transaction is delivered	
newBalDestinationClient	The new balance of a recipient client after a transaction has taken place	Continuous
isFraud	The target variable, label 1 for fraud and 0 for a legitimate transaction	Categorical

4.3 Feature Selection

Additional new features were created to improve model predictions. The balance differences between the old balance and the new balance for both the client starting the transaction and also for the recipient of a transaction were determined to form new features. This was aimed at identifying any significant changes in the account balance for the clients, potentially those associated with fraudulent transactions. The ratio of the transaction amount to the old balance of the starting client was also determined. A high transaction amount compared to the initial balance could be a signal of a fraudulent transaction. Similarly, the ratio between the transaction amount and the new balance of the starting client was determined. Also, large transactions were of interest, a transaction that

was greater than two standard deviations above the mean transaction amount. Transactions that were abnormally large were more likely to be fraudulent and a feature was created. The dataset was normalized using the min-max scaler method since mobile money transactions do not follow the normal distribution. The dataset was split into a training set (70%) and a test set (30%), each containing legitimate and fraudulent transactions to ensure that training and testing were performed using distinct sets.

Fundamental metrics encompassing the true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) form the basis for the model performance evaluation metrics that were used in the study. TP is concerned with an ML algorithm predicting that a transaction is fraudulent and the outcome is indeed fraud. On the other hand, FP involves the algorithm predicting a transaction to be fraudulent when actually it is not fraudulent. TN deals with a transaction predicted to be not fraudulent and there was no fraud while the FN which is the *hidden fraud* embraces the prediction of no fraud yet there was a fraudulent transaction. Common machine learning algorithms were adopted for financial fraud detection and their performances were evaluated to determine a consistent algorithm for the task [41].

4.4 Model Performance Evaluation

Precision. This is widely used and it is the ratio of correctly predicted positive observations to the total predicted positive observations given by the relation

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{1}$$

In practice, a balance needs to be established between the precision and recall for a financial fraud classification algorithm.

Recall. Sometimes referred to as sensitivity is the ratio of correctly predicted positive observations to all observations in the actual class, which is given by

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{2}$$

Usually, a fraud classification algorithm with a higher recall but lower precision will correctly discern more of the fraudulent transactions and incorrectly predict more transactions to be fraudulent resulting in false positives. A classifier with a higher precision but lower recall will miss some fraudulent transactions but will not incorrectly predict too many transactions as fraudulent. Therefore, service providers and financial institutions aim to register a balance between the two metrics (1),(2) in order to achieve better results.

F1-Score. This is another common model performance evaluation metric concerned with the weighted average of precision and recall. F1-score balances the

two metrics (1),(2) and it is very useful especially when a financial institution is interested in a single model performance evaluation metric that combines recall and precision. Good financial fraud classification algorithms should have high scores for precision, recall and f1-score metrics [22]. The f1-score is expressed as

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3}$$

Mathew Correlation Coefficient (MCC). The MCC is a measure of the quality of binary classification. Unlike accuracy which deals with the proportion of correct predictions over all of the predictions, the MCC is a better measure and regarded as a balanced measure for classes with different sizes [12,38]. The capacity of the MCC to work well in scenarios where one class is more frequent than the other makes it a suitable metric for financial institutions to use [12,38]. The score is in the range of [-1,1], where a +1 corresponds to a perfect prediction while a −1 indicates an inverse prediction and a coefficient of 0 represents a random prediction [21]. The MCC is given by the relation

$$\text{MCC} = \frac{(\text{TP} \times \text{TN} - \text{FP} \times \text{FN})}{\sqrt{((\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FP}) \times (\text{TN} + \text{FN}))}}. \tag{4}$$

Receiver Operating Characteristics (ROC). Besides other common evaluation metrics, the ROC plots the true positive rate (TPR) against the false positive rate (FPR) at various threshold values. The ROC curve that is closer to the left corner of the graph represents a good classification model while the one closer to the diagonal line represents a random model. The area under the ROC curve (AUC-ROC) was also used to compare the different fraud classification algorithms and its value ranges from 0 to 1. A value closer to 1 indicates good prediction while one closer to 0 is otherwise [10,22].

5 Results and Discussion

5.1 Conformity of Synthetic Datasets to Real Data

We executed MoMTSim [30] a number of times and several output files were written including synthetic mobile money transaction logs. Different simulations used different seed data as input. A complete simulation was executed for 720 steps, given that a step in the simulation platform represents one hour in the real world. This implies that a single simulation run represents one month (30 days) of transaction activity in the real ecosystem. The number of agents; mobile money merchants and clients were adjusted to output $1,040,000$ rows of transactions, sufficient for machine learning tasks. The resulting datasets were evaluated using the sum of squared errors (SSE) method and we obtained a dataset named MoMTSim_202306 to mean simulated for the month of June 2023 using MoMTSim. The dataset we picked for analysis had the least total error even though

other datasets registered relatively low total errors. The aggregated transactions for the real and synthetic datasets were visualised in order to compare the trends of transactions in the datasets. The transaction count, the aggregate transaction value, the average transaction value and the standard deviation for the real mobile money data and synthetic data were considered as shown in Figs. 2, 3 and 4. The trends in the payments, transfers and deposits were more of interest to show given that the fraud schemes discussed in Subsect. 1.1 mainly affected them. Other transaction types including debit and withdrawal that were present in the simulations showed statistical closeness even though their plots have not been included in this paper since they largely remained unaffected by the fraudulent activities. The uniform green line in all the plots indicates the trends of the real data while the dashed brown line indicates the trends of the synthetic data. Clearly, the trends are similar for both datasets and the small variations indicate that the datasets are not exactly the same. With this observation, MoMTSim generates synthetic datasets that statistically resemble the real data. Therefore, financial fraud classification using machine learning classifiers and rich synthetic data followed the assessment of the statistical closeness of the generated data to the real data.

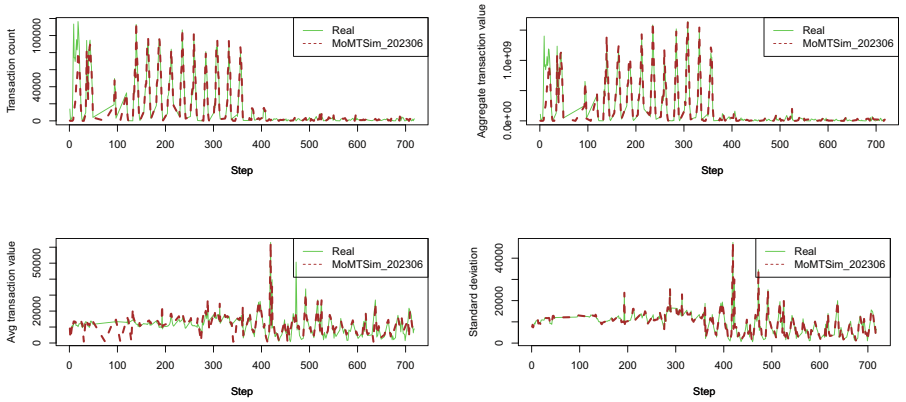


Fig. 2. Trends of payment transactions.

5.2 Mobile Money Fraud Classification Results

Synthetic Transactions. The simulated data containing 1,040,000 transactions had five transaction types composed of deposit, withdrawal, debit, payment and transfer. As shown in Fig. 5, 33.52% of the transactions are payments, followed by 29.37% deposits, 24.96% transfers, 11.47% withdrawals and 0.68% debits. In the simulated data, payment, deposit, and transfer transactions outnumber withdrawal and debit transactions. This is because the fraud schemes

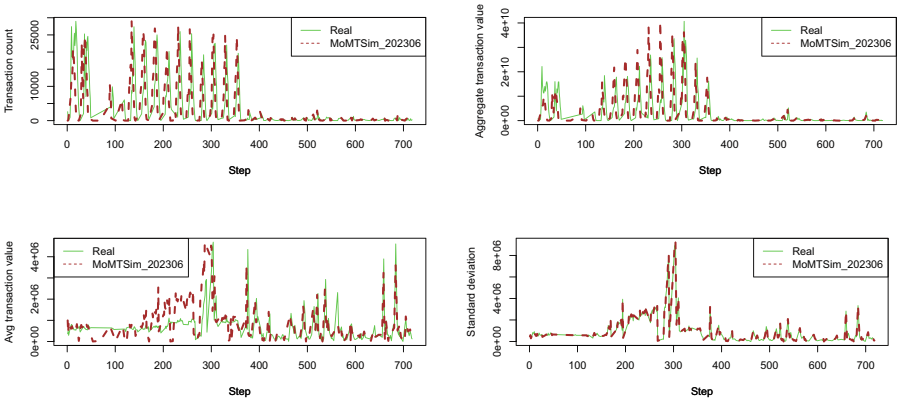


Fig. 3. Trends of transfer transactions.

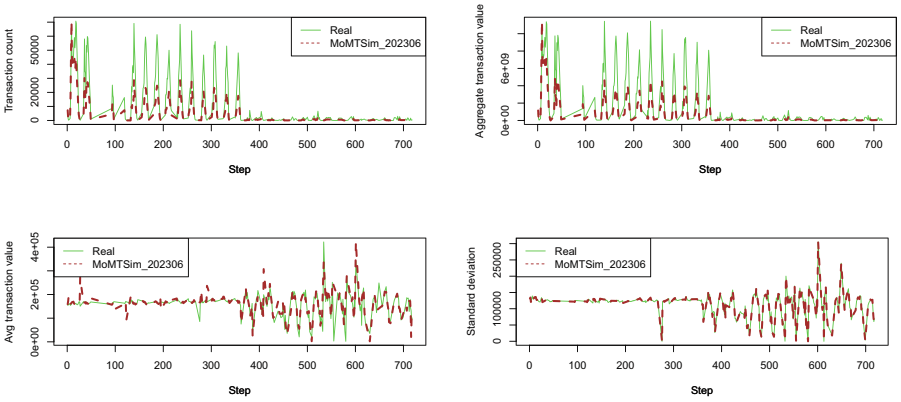


Fig. 4. Trends of deposit transactions

injected into MoMTSim mainly affected deposits, payments and transfers. Moreover, the fraudsters aim to carry out more transactions so as to increase their gains. A deposit is an entry point for hard cash into the mobile money system in the form of electronic money carried with the help of a mobile money merchant who facilitates the conversion. This implies that for other transactions to happen within the mobile money system, a deposit must have occurred initially. The debit transactions in the simulated data are the least frequent due to the clients avoiding high transaction charges associated with them. Usually, a client owning a bank account would prefer free deposits at the bank than a debit from their mobile money account which involves a service charge. Withdrawal is an exit point for converting electronic funds to hard cash ultimately a way to take money off the mobile money system. Usually, a number of clients engage in withdrawal transactions since cash is largely used in the Sub-Saharan region for daily purchases of goods and services.

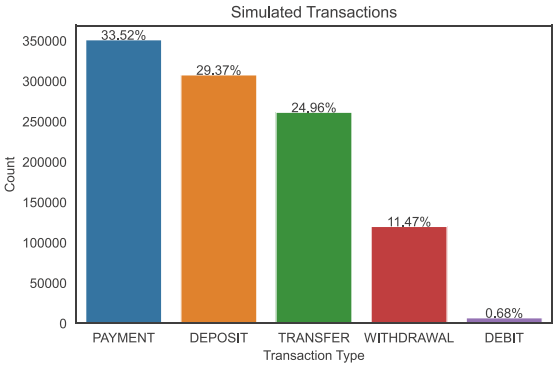


Fig. 5. Proportion of transactions simulated in MoMTSim_202306

Model Performance. The model performance results presented in this study are for the testing set and the Logistic regression model was used as the base model owing to its simplicity and interpretability. In the experiment, the Logistic regression model achieved an MCC of 0.67. The Logistic regression was able to detect fraudulent mobile money transactions with a high recall of 0.86, it was less precise, with a precision of 0.68 thus this performance served as the baseline. This implies that the model is good at identifying actual fraud cases, but it might also include more false positives, resulting in a lower precision. The model performances for all classifiers used in the experiment are shown in Table 2 and the ROC curves in Fig. 6. Other models including Gradient boosting, Decision trees, AdaBoost, XGBoost, and Random forest were then evaluated against the baseline model and they all demonstrated improvements with MCC scores ranging from 0.75 to 0.82. More specifically, the XGBoost model significantly outperformed the baseline model with an MCC of 0.82, revealing the effectiveness of more complex models in the realm of mobile money financial fraud detection. The XGBoost provides the best balance between precision and recall, among the models used. With the highest MCC (0.82), the highest precision of 0.98 for the fraudulent transactions and a higher AUC of 0.97, XGBoost offers a good balance, making it a preferred choice for the task compared to other models. By design, the XGBoost model supports parallel processing, making it quickly process large financial datasets and tree pruning allows for depth-first growth of trees and then pruning them, leading to more optimal trees, unlike Gradient boosting. Besides, it is flexible, allowing the definition of custom optimization objectives and evaluation criteria in order to fine-tune the model for specific needs. Moreover, it incorporates L1 (Lasso) and L2 (Ridge) regularization to prevent overfitting by minimizing the complexity of the model and distributing the weights more evenly across all the features [34,45,46]. Therefore, the XGBoost registers high performance on structured data compared to state-of-the-art deep learning models which often perform better on unstructured data.

Table 2. Model performance results

Model	Precision	Recall	F1-score	MCC
XGBoost	0.98	0.75	0.85	0.82
Gradient Boosting	0.97	0.74	0.84	0.81
AdaBoost	0.96	0.74	0.84	0.80
Random Forest	0.93	0.76	0.84	0.79
Decision Tree	0.81	0.82	0.81	0.75
Logistic Regression	0.68	0.86	0.76	0.67

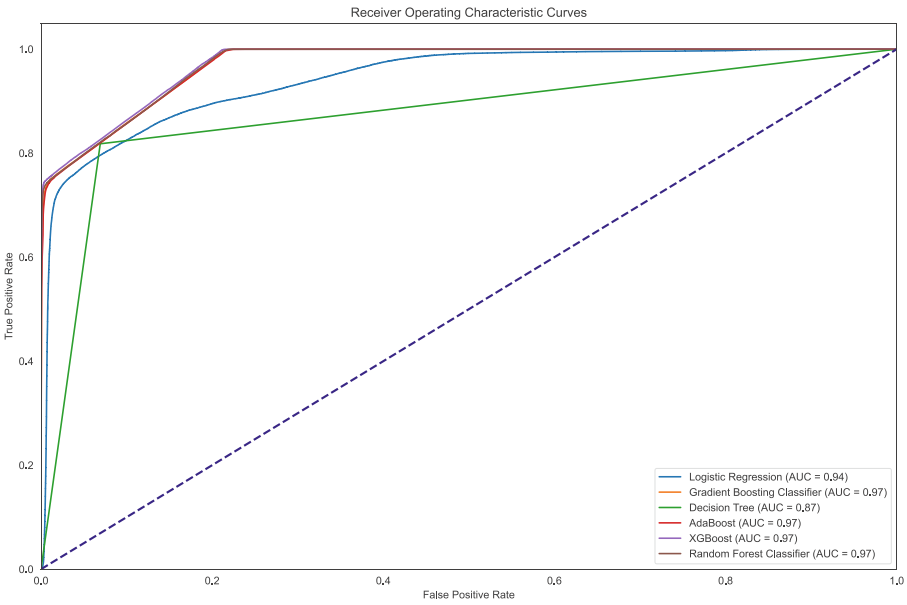


Fig. 6. ROC curves for the classification models used

Rule-based approaches underperform on either data mainly because of their inability to adapt to unanticipated scenarios in financial transactions.

The Gradient boosting classifier and AdaBoost also performed well, with MCCs of 0.81 and 0.80, respectively. In particular, the Gradient boosting classifier has a high MCC (0.81) and AUC of 0.97 implying overall good performance. The model has high precision which means the Gradient boosting classifier is good at identifying fraudulent mobile money transactions. AdaBoost showed similar performance to the Gradient boosting model, with a slightly lower MCC of 0.80. The Random forest classifier has an MCC of 0.79, which is good but still less than that one of XGBoost. It also has a high precision of 0.93 for fraudulent transactions, making it reliable in identifying fraudulent mobile money transactions. The Decision tree classifier has an MCC of 0.75, a balanced preci-

sion and recall for fraudulent transactions but it has less effective performance than the boosting models. Even though the Logistic regression and Decision tree algorithms registered relatively lower MCC scores, they still showed remarkable performance for the task.

Financial institutions and service providers usually consider a number of factors for instance the computational costs and the complexity of a model before committing themselves to use it in real-world applications. However, most of them rely on model performance evaluation metrics that provide a balanced measure and a model that offers superior performance. The XGBoost would be a model of choice for production though it is more computationally intensive than Logistic regression or Decision trees. Security teams and managers shall be capable of making decisions on what model to adopt especially where specific requirements are involved, considering a simpler model if computational resources are a constraint.

6 Conclusions and Future Work

This study demonstrates that financial institutions can stay ahead of fraudsters by simulating unique fraudulent behaviours in mobile money services using the MoMTSim platform. Besides, the study shows that synthetic data that statistically resembles real data can be used for research in the absence of real data.

This work provides a comprehensive evaluation of common machine learning algorithms including Logistic regression, Gradient boosting, Decision trees, AdaBoost, XGBoost, and Random forest, in terms of their capacity to detect fraudulent mobile money financial transactions. By using Logistic regression as a baseline model, the study offers a benchmark against which the performance of more complex models can be compared. This provides a clear insight into the improvements possible with more sophisticated techniques that might be of interest to researchers, service providers or financial institutions. The study emphasizes the use of the MCC metric, a more balanced measure for classification problems as in the case of mobile money fraud, especially in scenarios with imbalanced classes including financial fraud detection. More so, the study identifies the XGBoost model as the most effective algorithm for this particular task, as it achieved the highest MCC (0.82), high precision and recall scores. The high performance of the model on structured data is attributed to its capabilities of parallel processing, and regularisation that integrates L1 and L2 regularisation to prevent overfitting. The flexibility of the model allows the definition of custom optimization objectives and evaluation criteria more easily, unlike the other algorithms. Also, the study highlights the importance of considering the computational costs associated with different models, which is crucial when considering the practical application of the algorithms. Ultimately, this work contributes to the growing body of evidence supporting the use of machine learning algorithms for detecting mobile money fraud in financial transactions, especially in the Sub-Saharan context.

Our future work shall focus on incorporating other machine learning models and the simulation of more fraudulent scenarios for the task of fraud detection as well as preserving privacy in the synthetic data.

Acknowledgements. This research was made possible in part by the Digital Credit Observatory (DCO), a program of the Center for Effective Global Action (CEGA), with support from the Bill & Melinda Gates Foundation; JPMorgan Chase & Co.; and Google PhD Fellowship Program. Any views or opinions expressed herein are solely those of the authors listed, and may differ from the views and opinions expressed by any funder or its affiliates. A number of financial institutions in the Sub-Saharan region provided expert opinions on the dynamics of the real financial ecosystem.

References

1. Adewumi, A.O., Akinyelu, A.A.: A survey of machine-learning and nature-inspired based credit card fraud detection techniques. Int. J. Syst. Assur. Eng. Manag. **8**, 937–953 (2017). https://doi.org/10.1007/s13198-016-0551-y
2. Aftabi, S.Z., Ahmadi, A., Farzi, S.: Fraud detection in financial statements using data mining and gan models. Expert Syst. Appl. **227**, 120144 (2023). https://doi.org/10.1016/j.eswa.2023.120144
3. Alghofaili, Y., Albattah, A., Rassam, M.A.: A financial fraud detection model based on lstm deep learning technique. J. Appl. Secur. Res. **15**(4), 498–516 (2020). https://doi.org/10.1080/19361610.2020.1815491
4. Ali, A.A., Khedr, A.M., El-Bannany, M., Kanakkayil, S.: A powerful predicting model for financial statement fraud based on optimized xgboost ensemble learning technique. Appl. Sci. **13**(4), 2272 (2023). https://doi.org/10.3390/app13042272
5. Ali, G., Ally Dida, M., Elikana Sam, A.: Evaluation of key security issues associated with mobile money systems in Uganda. Information **11**(6), 309 (2020). https://doi.org/10.3390/info11060309
6. Apiors, E.K., Suzuki, A.: Effects of mobile money education on mobile money usage: Evidence from ghana. Eur. J. Dev. Res. 1–28 (2022). https://doi.org/10.1057/s41287-022-00529-x
7. Aslam, N., et al.: Anomaly detection using explainable random forest for the prediction of undesirable events in oil wells. Appl. Comput. Intell. Soft Comput. **2022** (2022). https://doi.org/10.1155/2022/1558381
8. Aswathi, M., Ghosh, A., Namboothiri, L.V.: Borda count versus majority voting for credit card fraud detection. In: Karuppusamy, P., Perikos, I., García Márquez, F.P. (eds.) Ubiquitous Intelligent Systems. SIST, vol. 243, pp. 319–330. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-3675-2_24
9. Bagga, S., Goyal, A., Gupta, N., Goyal, A.: Credit card fraud detection using pipeling and ensemble learning. Procedia Comput. Sci. **173**, 104–112 (2020). https://doi.org/10.1016/j.procs.2020.06.014
10. Botchey, F.E., Qin, Z., Hughes-Lartey, K.: 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

11. Chhabra Roy, N., Prabhakaran, S.: Internal-led cyber frauds in Indian banks: an effective machine learning-based defense system to fraud detection, prioritization and prevention. Aslib J. Inf. Manag. **75**(2), 246–296 (2023). https://doi.org/10.1108/AJIM-11-2021-0339
12. Chicco, D., Jurman, G.: An invitation to greater use of matthews correlation coefficient (mcc) in robotics and artificial intelligence. Front. Rob. AI, 78 (2022). https://doi.org/10.3389/frobt.2022.876814
13. Cordasco, G., Scarano, V., Spagnuolo, C.: Distributed mason: a scalable distributed multi-agent simulation environment. Simul. Model. Pract. Theory **89**, 15–34 (2018). https://doi.org/10.1016/j.simpat.2018.09.002
14. Danenas, P.: Intelligent financial fraud detection and analysis: a survey of recent patents. Recent Patents Comput. Sci. **8**(1), 13–23 (2015)
15. Dighe, D., Patil, S., Kokate, S.: Detection of credit card fraud transactions using machine learning algorithms and neural networks: a comparative study. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1–6. IEEE (2018). https://doi.org/10.1109/ICCUBEA.2018.8697799
16. Fehler, M., Klügl, F., Puppe, F.: Techniques for analysis and calibration of multi-agent simulations. In: Gleizes, M.-P., Omicini, A., Zambonelli, F. (eds.) ESAW 2004. LNCS (LNAI), vol. 3451, pp. 305–321. Springer, Heidelberg (2005). https://doi.org/10.1007/11423355_22
17. Gelb, A., Mukherjee, A.: Digital technology in social assistance transfers for covid-19 relief: lessons from selected cases. CGD Policy Paper **181** (2020)
18. Kanobe, F., Bwalya, K.J.: Snags in mobile money in developing economies. Electron. J. Inf. Syst. Dev. Countries **88**(3), e12181 (2022). https://doi.org/10.1002/isd2.12181
19. Kosolwattana, T., Liu, C., Hu, R., Han, S., Chen, H., Lin, Y.: A self-inspected adaptive smote algorithm (sasmote) for highly imbalanced data classification in healthcare. BioData Mining **16**(1), 15 (2023). https://doi.org/10.1186/s13040-023-00330-4
20. Lawson-Lartego, L., Cohen, M.J.: 10 recommendations for African governments to ensure food security for poor and vulnerable populations during covid-19. Food Secur. **12**(4), 899–902 (2020). https://doi.org/10.1007/s12571-020-01062-7
21. Lin, J.: Backtracking search based hyper-heuristic for the flexible job-shop scheduling problem with fuzzy processing time. Eng. Appl. Artif. Intell. **77**, 186–196 (2019). https://doi.org/10.1016/j.engappai.2018.10.008
22. Lokanan, M.E., Sharma, K.: Fraud prediction using machine learning: the case of investment advisors in Canada. Mach. Learn. Appl. **8**, 100269 (2022). https://doi.org/10.1016/j.mlwa.2022.100269
23. Lopez-Rojas, E., Elmir, A., Axelsson, S.: PaySim: a financial mobile money simulator for fraud detection. In: 28th European Modeling and Simulation Symposium, EMSS, Larnaca, pp. 249–255. Dime University of Genoa (2016)
24. Lopez-Rojas, E.A., Barneaud, C.: Advantages of the PaySim simulator for improving financial fraud controls. In: Arai, K., Bhatia, R., Kapoor, S. (eds.) CompCom 2019. AISC, vol. 998, pp. 727–736. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-22868-2_51
25. Lopez-Rojas, E.A.: Extending the retsim simulator for estimating the cost of fraud in the retail store domain. In: The 27th European Modeling and Simulation Symposium-EMSS, Bergeggi, Italy (2015)

26. Lopez-Rojas, E.A., Axelsson, S., Baca, D.: Analysis of fraud controls using the PaySim financial simulator. Int. J. Simul. Process Model. **13**(4), 377–386 (2018). https://doi.org/10.1504/IJSPM.2018.093756
27. Luengo, J., Fernández, A., García, S., Herrera, F.: Addressing data complexity for imbalanced data sets: analysis of smote-based oversampling and evolutionary undersampling. Soft. Comput. **15**, 1909–1936 (2011). https://doi.org/10.1007/s00500-010-0625-8
28. Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., Balan, G.: Mason: a multiagent simulation environment. Simulation **81**(7), 517–527 (2005). https://doi.org/10.1177/0037549705058073
29. Luke, S., et al.: The MASON simulation toolkit: past, present, and future. In: Davidsson, P., Verhagen, H. (eds.) MABS 2018. LNCS (LNAI), vol. 11463, pp. 75–86. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-22270-3_6
30. MoMTSim, Inc: Mobile Money Transaction Simulator (2023). https://github.com/aiinfinancegroup/MoMTSim, version 0.1.0
31. Mudiri, J.L.: Fraud in mobile financial services. Rapport technique, MicroSave **30** (2013)
32. Muthali, A., et al.: Multi-agent reachability calibration with conformal prediction. arXiv preprint arXiv:2304.00432 (2023). https://doi.org/10.48550/arXiv.2304.00432
33. Narayan, A., Madhu Kumar, S., Chacko, A.M.: A review of financial fraud detection in e-commerce using machine learning. In: Intelligent Data Engineering and Analytics: Proceedings of the 10th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA 2022), pp. 237–248. Springer, Heidelberg (2023). https://doi.org/10.1007/978-981-19-7524-0_21
34. Nti, I.K., Somanathan, A.R.: A scalable rf-xgboost framework for financial fraud mitigation. IEEE Trans. Comput. Social Syst. (2022). https://doi.org/10.1109/TCSS.2022.3209827
35. Nunes, R.P.M., Bonacin, R., de Franco Rosa, F.: Methods for detecting fraud in civil and military service examinations: a systematic mapping. In: Latifi, S. (ed.) ITNG 2021 18th International Conference on Information Technology-New Generations. AISC, vol. 1346, pp. 203–208. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-70416-2_26
36. Öztürk, M.S., Usul, H.: Detection of accounting frauds using the rule-based expert systems within the scope of forensic accounting. In: Contemporary Issues in Audit Management and Forensic Accounting, vol. 102, pp. 155–171. Emerald Publishing Limited (2020). https://doi.org/10.1108/S1569-375920200000102013
37. Park, J., Kwon, S., Jeong, S.P.: A study on improving turnover intention forecasting by solving imbalanced data problems: focusing on smote and generative adversarial networks. J. Big Data **10**(1), 1–16 (2023). https://doi.org/10.1186/s40537-023-00715-6
38. Powers, D.M.: Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061 (2020). https://doi.org/10.48550/arXiv.2010.16061
39. Raiter, O.: Applying supervised machine learning algorithms for fraud detection in anti-money laundering. J. Modern Issues Bus. Res. **1**(1), 14–26 (2021). https://doi.org/10.17613/2g0z-0814
40. Shahana, T., Lavanya, V., Bhat, A.R.: State of the art in financial statement fraud detection: a systematic review. Technol. Forecast. Soc. Chang. **192**, 122527 (2023). https://doi.org/10.1016/j.techfore.2023.122527

41. Singh, A., Jain, A., Biable, S.E.: Financial fraud detection approach based on firefly optimization algorithm and support vector machine. Appl. Comput. Intell. Soft Comput. **2022** (2022). https://doi.org/10.1155/2022/1468015
42. Singh, K., Best, P.: Anti-money laundering: using data visualization to identify suspicious activity. Int. J. Account. Inf. Syst. **34**, 100418 (2019). https://doi.org/10.1016/j.accinf.2019.06.001
43. Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C.B., Goldstein, T.: Saint: improved neural networks for tabular data via row attention and contrastive pre-training. arXiv preprint arXiv:2106.01342 (2021). https://doi.org/10.48550/arXiv.2106.01342
44. Soni, V.D.: Role of artificial intelligence in combating cyber threats in banking. Int. Eng. J. Res. Dev. **4**(1), 7–7 (2019)
45. Tang, Q., et al.: Prediction of casing damage in unconsolidated sandstone reservoirs using machine learning algorithms. In: 2019 IEEE International Conference on Computation, Communication and Engineering (ICCCE), pp. 185–188. IEEE (2019). https://doi.org/10.1109/ICCCE48422.2019.9010785
46. Zhang, Y., Tong, J., Wang, Z., Gao, F.: Customer transaction fraud detection using xgboost model. In: 2020 International Conference on Computer Engineering and Application (ICCEA), pp. 554–558. IEEE (2020). https://doi.org/10.1109/ICCEA50009.2020.00122