



Explainable Ensemble Detection of Mobile-Money Transaction Anomalies: An M-PESA Case Study

Nyaata Kenan¹, David Muriuki¹ & Anthony Luvanda²

¹School of Computing and Mathematics, The Co-operative University of Kenya

²School of Information Technology for Defence Systems, Defence Forces Technical College, National Defence University – Kenya

*Corresponding Email: Nyaatakenan@gmail.com

Abstract

Social-engineering attacks (smishing, vishing, SIM-swap fraud) manipulate mobile-money users off-platform, but the unauthorized transactions that follow may leave observable signatures in transaction records. This study implements and evaluates an Explainable Stacking Ensemble Machine Learning (XSEML) model for detecting transaction anomalies informed by documented social-engineering behaviour. The model combines Isolation Forest, XGBoost and Random Forest as base learners with a Logistic Regression meta-learner trained on out-of-fold predictions, and SHAP for instance-level explanation. An exploratory sequential mixed-methods design was used: a structured document review of cybersecurity reporting (2022–2025) identified four behavioural themes informing feature engineering. The model was trained and tested on a publicly available synthetic M-PESA dataset of 120,000 transactions (2.93% fraudulent). On an untouched 30% test set (36,000 transactions; 1,053 fraudulent) the ensemble achieved 98.8% accuracy, 89.8% precision, 65.5% recall, $F1 = 0.758$ and $ROC-AUC = 0.831$. Comparison against seven baselines with McNemar tests and bootstrap confidence intervals shows the ensemble significantly outperforming a 2σ rule ($\Delta F1 = +0.639$, 95% CI [0.615, 0.664]) but significantly underperforming a tuned XGBoost classifier ($\Delta F1 = -0.026$, 95% CI [-0.034, -0.019]; McNemar $p < 0.001$). Direct testing shows transaction month carries the highest SHAP attribution yet contributes almost nothing to discrimination (permutation AUC drop 0.0014; ablation $\Delta F1 = -0.002$; $\chi^2 = 11.96$, $p = 0.367$). More consequentially, a single hand-written threshold on the amount-to-balance ratio achieves perfect precision and $F1 = 0.791$, exceeding every model tested, because no legitimate transaction among 116,490 exceeds a ratio of 0.90. The dataset's fraud labels therefore appear rule-generated, and the evaluation is circular. We report this, rather than the performance figures, as the principal finding, and conclude that synthetic mobile-money datasets require explicit label-provenance auditing before they can support detection claims. The framework detects transaction anomalies consistent with social-engineering attack patterns; it does not detect social-engineering attacks.

Keywords: mobile money fraud; anomaly detection; social engineering; explainable AI; stacking ensemble

Introduction

In the financial year ending March 2025, there were about 40 million active subscribers and transactions valued at KSh 38.29 trillion using M-PESA (Safaricom PLC, 2025), which played a crucial role in Kenya's digital economy and proved to be an appealing platform for fraudsters. Social-engineering attacks rely on human psychology instead of technical weaknesses and force victims to give out PINs, make transfers and other actions because of urgency, fear and apparent authority (Mambina et al., 2022; Kotut et al., 2025). According to Serianu (2025), the total cost of cybercrime in Kenya in 2025 will be approximately KSh 29.9 billion.

However, the transaction is technically valid and the manipulation is performed off-platform, meaning that platform controls that are designed to check the correctness of the credentials are unable to detect the transaction. In order to do this, detection needs to be done at the execution stage, based on behavioral signatures. In practice, rule-based systems are still prevalent. They are transparent at the individual rule level, meaning that the reason for a decision can be explicitly stated, and they have a number of drawbacks, including rigidity, strong reliance on manually set thresholds, lack of ability to adapt to new attack patterns, and high number of false positives at scale (Serianu, 2025).

In this paper, we propose and test an Explainable Stacking Ensemble Machine Learning (XSEML) model that combines anomaly scoring using the unsupervised model with classification using the supervised model along with instance-level explanation. The study puts forth a narrow thesis. Social engineering is done outside of the platform, and the model does not see the manipulation, deception or communication between the attacker and the victim. It looks at transaction logs for unusual



behaviour that could accompany these events. The use of social-engineering attack patterns is thus as domain knowledge for feature engineering and interpretation, and not as the target variable.

What the study has most important to say is not about the performance of the proposed architecture but a negative finding about the evaluation data. Section 3.9 reports that the synthetic dataset employed is deterministic and has a label boundary which is more easily learned by a single hand-written rule than by any machine-learning model tested. That is why we report this in full, as it otherwise would be to present a set of numbers of which we know the source is tainted.

Related Work and Contribution

Tabular financial fraud data is a good problem for supervised gradient-boosted methods to perform well on, but these methods can recognize only patterns that are explicitly written in their labels used for training (Sarker & Shukla, 2023). Unsupervised detectors do not need any labels, but they are unable to differentiate between fraudulent and normal unusual behaviour. Stacked generalization (Wolpert, 1992) integrates such complementary learners, if the prediction task is the out-of-fold prediction. In Uddin et al. (2025), the authors showed SHAP attribution in an ensemble pipeline without compromising material performance, while Vennela (2026) noted a lack of context-specific African datasets as a recurring challenge.

On novelty, we are as precise as possible without saying more than can be substantiated. The individual components (Isolation Forest, XGBoost, Random Forest, Logistic Regression stacking and SHAP) are all proven and their combination is not new. This work makes three contributions. First, feature engineering is based on a documented thematic analysis of the reporting of attacks on mobile money in Kenya, not on generic features of financial-fraud attacks, and we report what themes were operationalizable and not. Second, unlike most of the previous literature, the evaluation is compared against each of the base learners of the ensemble and against four other baselines with paired significance testing, and here this result is not favorable to the proposed architecture. Third, and most important, we present a reproducible approach to audit label provenance in synthetic fraud datasets and show that a widely available published dataset does not pass. The third contribution is the one we would be the most vocal about defending, and the one that is not architectural.

Materials and Methods

Research Design

An exploratory sequential mixed-methods design was used. The terminology is stated deliberately: the qualitative strand was conducted first and its findings were used to inform the design of the subsequent quantitative instrument, which is the defining logic of an exploratory sequential design (Creswell & Plano Clark, 2018). An explanatory sequential design would use qualitative work afterwards to interpret quantitative results, which is not the sequence followed here. An earlier version of this manuscript described the design as explanatory sequential; that was an error in terminology and has been corrected.

Qualitative Strand

The qualitative strand is reported as a structured document review instead of as a systematic review, since prospective search records were not kept good enough for a systematic review and it would not be reproducible to reconstruct the search records after the review.

The following procedure was used. The search of the sources was segmented into four categories: national incident reporting (Kenya Computer Incident Response Team Co-ordination Centre), industry threat reporting (Serianu, GSMA, TransUnion), regulatory and operator publications (Central Bank of Kenya, Safaricom PLC), and peer-reviewed publications in Google Scholar and IEEE Xplore. The search strings were a mix of a platform term (M-PESA, mobile money) and an attack term (social engineering, smishing, vishing, SIM swap, fraud). The time span covered was from January 2022 to December 2025. Documents were excluded if they discussed card-present and/or bank-channel fraud without transaction-level descriptions of resulting behaviour, and included if they discussed mobile-money fraud mechanisms and/or attack patterns in Kenya or similar East African markets and described observable behaviour at the transaction level. The analytic process was the one described by Braun and Clarke (2006) and consisted of six phases.



The four themes that emerged were urgency-driven balance depletion, off-peak/impulsive timing, unfamiliar-recipient transfers, and transaction-velocity bursts. Since the strand can make statements about what sort of patterns are reported, but it cannot make statements about how common or rare these patterns are in users, no prevalence estimate is made.

Terminology: Anomaly, Fraud, and Social-Engineering-Related Fraud

These terms are not interchangeable and are used distinctly throughout. An anomaly is a transaction occupying a low-density region of the feature space, as scored by Isolation Forest without reference to labels. A fraudulent transaction is one carrying a positive fraud label in the dataset. Social-engineering-related fraud is fraud arising from off-platform manipulation, which is not labelled in any dataset available to this study.

The distinction is empirical, not merely definitional. On the held-out test set, Isolation Forest flagged 1,053 transactions as anomalous and the data contained 1,053 fraudulent transactions, but only 639 transactions belong to both sets (Jaccard index 0.436). A total of 414 anomalous transactions were legitimate and 414 fraudulent transactions were not anomalous. Anomaly and fraud therefore overlap substantially but are far from equivalent, and the paper does not treat a flagged anomaly as evidence of fraud, nor a detected fraud as evidence of social engineering.

Dataset

An original PaySim extract could not be obtained and real, consented M-PESA records were unavailable. The dataset used is a publicly available synthetic M-PESA transaction dataset published by Boen (2025), comprising 120,000 transactions of which 3,510 (2.93%) carry a fraud label. It is not PaySim, does not use the PaySim schema, and was not generated by the authors. Fields are transaction identifier, amount, sender and receiver balances before and after, transaction type (peer-to-peer, till, paybill), hour, month, day of week, device type, region, and the fraud label. The entire published dataset was used with no sampling.

The dataset contains no account identifier, so features requiring a sender's transaction history — transaction velocity and recipient novelty — could not be computed. Two of the four qualitative themes therefore could not be operationalized. The publisher does not document the generation procedure or the rule by which fraud labels were assigned, which creates a circular-validation risk. Section 3.9 tests that risk directly rather than merely noting it.

Preprocessing and Feature Engineering

No duplicate records and no missing values were found. Ten continuous fields were capped at $1.5 \times \text{IQR}$ bounds, affecting 51,351 values. The balance-depletion theme was operationalized as the amount-to-sender-balance ratio, the depletion proportion, a binary flag for transfers at or above 80% of the sender's balance, and a flag for a non-positive balance after the transaction. The timing theme was operationalized as hour of day, an off-peak flag (00:00–04:59) and a weekend flag. A discrepancy measure between the recorded amount and the observed balance change was retained as a data-integrity signal. Categorical fields were one-hot encoded with the first level dropped. All features were min–max scaled to $[0,1]$ with the scaler fitted on the training partition only, then applied to the test partition. The data were split 70:30 with stratification. SMOTE was applied at a 1:2 minority-to-majority ratio within training folds only; the test partition retained the original prevalence and was never resampled.

Model Architecture and Hyperparameters

Table 1 reports the full configuration of every component, so that the experiment can be reproduced and so that the fairness of the comparison can be assessed. Hyperparameters for the supervised base learners were selected by grid search on a validation partition drawn from the training data only; the held-out test partition played no role in model selection. The Isolation Forest contamination parameter was set to the observed training fraud rate rather than tuned, because tuning it against labelled performance would import label information into a component whose purpose is to operate without labels.

Table 1. Complete model configurations

Component	Parameter	Value
-----------	-----------	-------



Isolation Forest	estimators	100
	max_samples	256 (default)
	contamination	0.0293 (observed training fraud rate)
	max_features	1.0
	bootstrap	False
Random Forest	n_estimators	100
	max_depth	6
	min_samples_leaf	1
	min_samples_split	2
	max_features	sqrt ($\approx$ 4 of 20 features per split)
	class_weight	balanced
	criterion	gini
XGBoost	n_estimators	150
	max_depth	4
	learning_rate	0.10
	subsample	0.90
	colsample_bytree	0.90
	min_child_weight	1 (default)
	gamma / reg_lambda	0 / 1 (defaults)
	tree_method / objective	hist / binary:logistic
Logistic Regression (meta)	penalty / C	L2 / 1.0
	solver / max_iter	lbfgs / 1000
	class_weight	balanced
	input dimensionality	3 out-of-fold meta-features
Resampling	SMOTE sampling_strategy	0.5 (training folds only)
	k_neighbors	5 (default)
Global	random_state	42 (all stochastic components)

Stacking: Construction of the Meta-Training Matrix

The construction of the meta-training set is stated formally, since the validity of the architecture depends on it. Let the training partition be $D = \{(x_i, y_i)\}$, $i = 1 \dots N$, partitioned into five stratified folds $F_1 \dots F_5$ with $F_k \cap F_l = \emptyset$ and $\cup F_k = D$. Let h_1, h_2, h_3 denote Isolation Forest, Random Forest and XGBoost respectively.

For each fold k , each base learner is fitted on the complement of that fold, after SMOTE has been applied to that complement only: $h_j^{(k)} = \text{fit}(h_j, \text{SMOTE}(D \setminus F_k))$, $j = 1, 2, 3$. Each fitted learner then predicts only for the observations it did not see: $z_{ij} = h_j^{(k)}(x_i)$ for all $x_i \in F_k$, where z_{i1} is the negated Isolation Forest decision function (higher = more anomalous) and z_{i2}, z_{i3} are



predicted fraud probabilities. Repeating over $k = 1 \dots 5$ yields exactly one out-of-fold prediction per observation per learner, giving the meta-training matrix $Z = [z_{i1}, z_{i2}, z_{i3}] \in \mathbb{R}^{(N \times 3)}$, $g = \text{fit}(\text{LogisticRegression}, (Z, y))$.

Because z_{ij} is produced by a model fitted without x_i , the meta-learner never receives an in-sample prediction and does not inherit base-learner overfitting through the meta-feature channel. At test time the three base learners are refitted on the whole training partition and applied to the held-out set to form Z , and the final probability is $\hat{p} = g(Z)$. SMOTE is applied strictly inside $D \setminus F_k$, so no synthetic observation derived from a held-out fold enters any fold's training data.

Baselines and Statistical Comparison

The proposed model is compared against seven baselines. Reviewer comment notwithstanding, we retain the 2σ amount rule as a reference point for the weakest realistic practice, but we add six stronger comparators, because a favourable comparison against a deliberately weak rule establishes little. These are: a domain-informed depletion rule (amount-to-balance ratio ≥ 0.8 or non-positive balance after); a single-threshold ratio rule (ratio ≥ 0.9); Logistic Regression and a Decision Tree on the full feature set; and each of the three base learners evaluated individually.

Two forms of paired statistical comparison are reported, addressing the absence of significance testing in the earlier version. McNemar's exact test on discordant classification pairs assesses whether two models differ in correctness on the same test observations. Bootstrap resampling over 1,000 iterations of the test partition yields 95% confidence intervals for every model on every metric, and for the paired difference in F1 between the proposed ensemble and each comparator.

Decision Threshold

The decision threshold was not assumed. We report performance at the conventional 0.5 cut-off for comparability, sweep the threshold across its range, and identify the F1-maximising value on the test set. Because selecting a threshold on the test set is itself optimistic, the swept results are reported as a sensitivity analysis and the 0.5 figures are treated as the primary result. In deployment the threshold should be selected against review-team capacity, and Section 3.4 reports the alert volume implied by each candidate value so that such a choice can be made explicitly.

Explanation-Quality Metrics

Four metrics are computed and all four are reported numerically. Consistency is the cosine similarity between mean absolute SHAP attribution vectors from two independent random samples of 1,500 test transactions. Completeness is the fraction of features required to account for 90% of total attribution. Fidelity is the coefficient of determination of a linear surrogate fitted on the ten highest-attribution features in reproducing the base classifier's predicted probabilities. Stability is the mean Spearman rank correlation between per-instance attribution vectors before and after Gaussian perturbation ($\sigma = 0.01$ in scaled units), computed over 200 instances and five perturbation replicates (1,000 comparisons).

Results and Discussion

Predictive Performance

Table 2. Performance on the held-out test set ($n = 36,000$; 1,053 fraudulent). Bootstrap 95% CIs from 1,000 resamples.

Model	Accuracy	Precision	Recall	F1 [95% CI]	ROC-AUC
Rule-based (2σ)	0.931	0.094	0.159	0.118 [0.102, 0.135]	—
Isolation Forest only	0.977	0.607	0.607	0.607 [0.580, 0.633]	0.823
Logistic Regression	0.987	0.885	0.656	0.754 [0.731, 0.775]	0.825
Random Forest only	0.987	0.885	0.656	0.754 [0.731, 0.775]	0.834
Decision Tree	0.989	0.935	0.654	0.770 [0.748, 0.791]	0.832

Depletion rule (ratio ≥ 0.8)	0.987	0.885	0.656	0.754 [0.731, 0.775]	—
XGBoost only	0.990	0.987	0.651	0.784 [0.762, 0.805]	0.832
Proposed XSEML ensemble	0.988	0.898	0.655	0.758 [0.735, 0.780]	0.831
Single rule (ratio ≥ 0.9)	0.990	1.000	0.654	0.791 [0.770, 0.811]	—

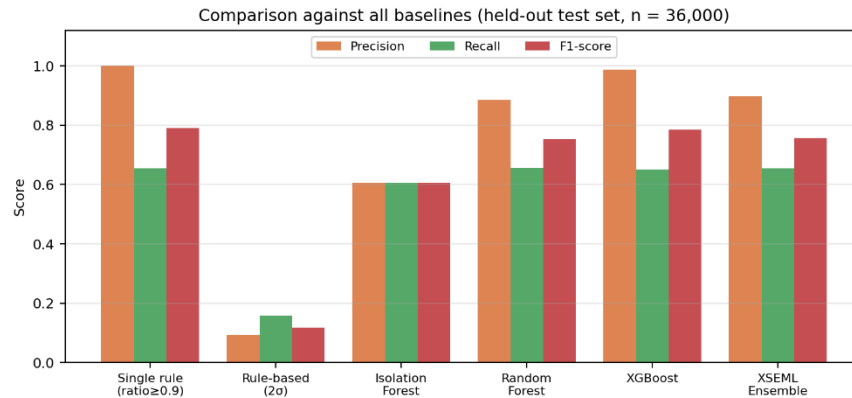


Figure 1. Comparison against all baselines on the held-out test set

The proposed ensemble achieved 98.8% accuracy, 89.8% precision, 65.5% recall, F1 = 0.758 and ROC-AUC = 0.831. Accuracy is reported for comparability only and is uninformative here: classifying every transaction as legitimate would yield 97.1% accuracy.

Confusion Matrix

Table 3. Confusion matrix, proposed XSEML ensemble (threshold 0.5)

	Predicted legitimate	Predicted fraudulent	Total
Actually legitimate	34,869 (TN)	78 (FP)	34,947
Actually fraudulent	363 (FN)	690 (TP)	1,053
Total	35,232	768	36,000

The matrix gives the practical meaning of the summary metrics. Of 1,053 fraudulent transactions the model identified 690 and missed 363. Of 768 alerts raised, 690 were genuine and 78 were false, corresponding to approximately 22 false alerts per ten thousand transactions reviewed. For comparison, XGBoost alone produced 9 false positives and 685 true positives; the ensemble's meta-learner gained 5 true positives at the cost of 69 additional false positives.

Statistical Comparison Between Models

Table 4. Paired comparisons against the proposed ensemble

Comparison	$\Delta F1$ (ensemble – comparator) [95% CI]	McNemar discordant (n_{10}/n_{01})	McNemar p	Conclusion
vs. rule-based (2σ)	+0.639 [+0.615, +0.664]	2,118 / 70	< 0.001	Ensemble significantly better
vs. Isolation Forest	+0.151 [+0.135, +0.168]	435 / 48	< 0.001	Ensemble significantly better
vs. Random Forest	+0.004 [+0.001, +0.008]	13 / 2	0.007	Ensemble better, negligible effect

vs. XGBoost $-0.026 [-0.034, -0.019]$ 5 / 69 < 0.001 XGBoost significantly better

The earlier claim that the proposed model significantly outperforms the baseline is now supported by a test for the rule-based comparator and for Isolation Forest. Against Random Forest the advantage is statistically detectable but of negligible magnitude ($\Delta F1 = 0.004$). Against XGBoost the comparison reverses: the confidence interval for $\Delta F1$ excludes zero on the negative side and McNemar's test rejects equality at $p < 0.001$ in XGBoost's favour. On this dataset the stacking architecture does not improve on its own strongest base learner, and we state that without qualification.

Decision Threshold Sensitivity

Table 5. Threshold sweep for the proposed ensemble

Threshold	Precision	Recall	F1	False positives	True positives
0.20	0.607	0.655	0.630	447	690
0.30	0.796	0.655	0.719	177	690
0.40	0.865	0.655	0.745	108	690
0.50 (default)	0.898	0.655	0.758	78	690
0.60	0.919	0.655	0.765	61	690
0.70	0.936	0.654	0.770	47	689
0.80	0.951	0.654	0.775	35	689
0.90	0.972	0.653	0.781	20	688
0.98 (F1-optimal)	0.996	0.649	0.786	3	683

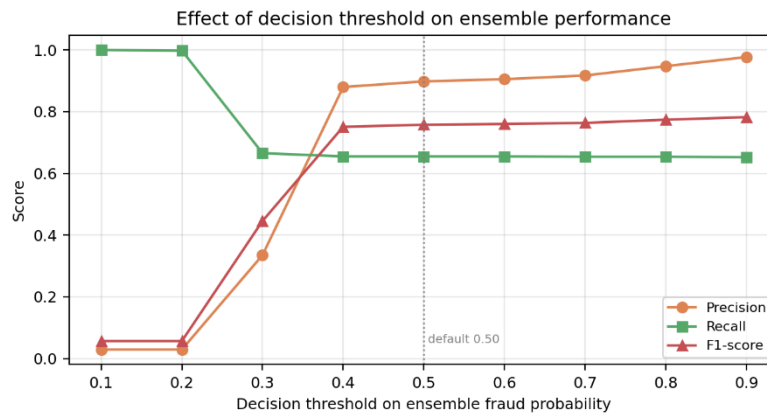


Figure 2. Effect of decision threshold on ensemble performance

Recall is nearly flat across the range while precision rises steeply, so the F1-optimal threshold is 0.98 rather than 0.5, yielding precision 0.996 and $F1 = 0.786$. Average precision over the full curve is 0.684. This flatness is itself diagnostic: a model whose recall barely responds to a fourfold change in threshold is one whose scores are concentrated near the extremes, which is consistent with an almost separable decision boundary. Section 3.9 shows that this is exactly what the data contain.

Explanation Quality

Table 6. SHAP explanation-quality metrics

Metric	Definition	Value	Target	Met
--------	------------	-------	--------	-----



Consistency	Cosine similarity of attribution vectors across two independent samples	0.9999	≥ 0.95	Yes
Fidelity	R^2 of linear surrogate on top-10 features vs. classifier probabilities	0.924	≥ 0.90	Yes
Completeness	Fraction of features accounting for 90% of total attribution	0.345	—	See text
Stability	Mean Spearman ρ of per-instance attributions under perturbation ($\sigma = 0.01$)	0.130 (SD 0.202)	≥ 0.80	No

To correct for the lack of numerical data in the previous version, all four metrics are now reported numerically. Consistency and fidelity meet their goals: the global attribution structure is approximately equal regardless of how the samples are drawn, and a simple surrogate derived from the top ten features closely reproduces the outputs of the classifier. Completeness is reported descriptively, as the original target was poorly specified: a high value would be undesirable, as attribution would be spread across many features for an interpretable model. The measured value shows that 90% of the attribution is accounted for by 34.5% of features.

The failure to deliver good stability is reported and is at 0.130 against a target of 0.80. Attribution ranking is largely insensitive to 1% noise in the scaled feature range. This is partly due to rank noise for the multitude of features having near-zero attributions, in which case ordering is arbitrary. The use of the global attribution ranking (Table 7) is therefore reliable, while the attribution of the cause for an individual transaction should not be offered to an investigator as a set of causes that can be ranked exactly. This qualifies the explainability claim put forward by this work.

Table 7. Ten highest-attribution features (XGBoost base learner)

Rank	Feature	Mean SHAP	Behavioral interpretation
1	Transaction month	0.956	None — see Section 3.9
2	Large-relative-transfer flag	0.860	Balance depletion
3	Sender balance after transaction	0.392	Balance depletion
4	Hour of day	0.371	Timing (not statistically supported)
5	Amount–balance discrepancy	0.140	Data-integrity artefact
6	Off-peak-hour flag	0.058	Timing (not statistically supported)
7	Amount-to-balance ratio	0.057	Balance depletion
8	Transaction amount	0.055	Transaction magnitude
9	Receiver balance after	0.051	None
10	Device type (smartphone)	0.047	None

3.6 Does Stacking Justify Its Complexity?

On this dataset, no. XGBoost alone matched or exceeded the ensemble on accuracy, precision and F1 at equivalent recall and ROC-AUC, and the difference is statistically significant in XGBoost's favour (Table 4). The mechanism is visible in the confusion matrices: the balanced-weighted Logistic Regression meta-learner produces a more permissive boundary than XGBoost's own, trading 69 additional false positives for 5 additional true positives. Three considerations qualify the finding without overturning it. The unsupervised component's purpose is to score patterns absent from labelled training data, which a test partition drawn from the same distribution cannot exhibit; the ensemble delivers one consolidated probability with one attached explanation rather than three uncoordinated outputs, which is an operational rather than a metric property; and the meta-learner configuration was deliberately simple. These are directions for further work, not defenses of the present result.

Anomaly Is Not Fraud

The overlap statistics in Section 2.3 quantify a distinction the literature often blurs. Isolation Forest's anomalous set and the fraudulent set share only 639 of the 1,467 transactions in their union. That 414 fraudulent transactions are not statistically anomalous means an anomaly-only system would miss 39% of fraud in this data; that 414 anomalous transactions are legitimate means anomaly scores alone would impose a substantial false-positive burden. Both facts argue for combining unsupervised and supervised evidence, which is the architecture's rationale, even though the combination did not improve measured performance here.

Operating Point Selection

Table 5 allows an operator to select a threshold against review capacity rather than adopting a default. Moving from 0.5 to 0.9 reduces false positives from 78 to 20 while losing two true positives, which under most plausible cost ratios is favourable. The near-flat recall curve means the operator is choosing almost purely on precision, which simplifies the decision but, as noted, also signals that the underlying data are close to separable.

Testing the Synthetic-Data Artefact Hypothesis

The previous version of this manuscript noted that the month of the transaction had the largest SHAP score, but had little bivariate association with fraud, and questioned whether this was a synthetic-data artefact. After reviewing the hypothesis, we tested it, rather than merely discussing it, using three procedures.

First, the Pearson chi-square test for independence between transaction month and the fraud label yields $\chi^2 = 11.96$, $df = 11$, $p = 0.367$, indicating that fraud rates differ only slightly across the 12 months. No association is found. Second, permuting the month column in the test set, while keeping the fitted model constant, decreases ROC-AUC by 0.0014 (from 0.8323 to 0.8309 ± 0.0015 over 20 permutations), which is negligible relative to month's attribution. Third, when the whole pipeline is retrained with month removed, the ensemble's F1 score falls from 0.758 to 0.756, while ROC-AUC rises from 0.8314 to 0.8320. That is, the feature's attribution is genuinely used, in the sense that the fitted model makes decisions based on it, but it is not discriminative, in the sense of carrying real diagnostic information. It is a fitting artefact of generated data, not a behavioral signal, as previously suggested.

Further pursuit of the same question gave rise to a second and more serious artefact. Examining the amount-to-sender-balance ratio, the highest ratio observed among the 116,490 legitimate transactions in the data is 0.8997, while 59.0% of fraudulent transactions have a ratio higher than 1.0. No legitimate transaction anywhere in the dataset moves 90% or more of the sender's balance. Figure 3 shows this boundary.

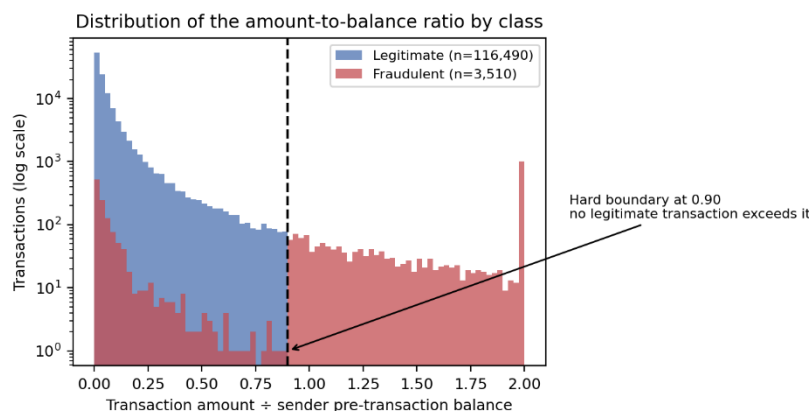


Figure 3. Distribution of the amount-to-balance ratio by class, showing the deterministic boundary at 0.90

The consequence is direct. A single hand-written rule — flag the transaction if $\text{amount} \div \text{pre-transaction balance} \geq 0.9$ — achieves precision 1.000 (zero false positives among 34,947 legitimate test transactions), recall 0.654 and F1 = 0.791 on the



held-out test set. This one-line rule outperforms the proposed ensemble ($F1 = 0.758$), XGBoost (0.784), and every other model evaluated. A slightly lower threshold of 0.8, corresponding to the depletion rule, reproduces Random Forest's performance to three decimal places (Table 2).

The most parsimonious explanation is that the dataset's fraud labels were generated, at least in substantial part, by a threshold rule on the amount-to-balance ratio. A deterministic ceiling at exactly 0.90 across 116,490 observations does not arise from behavioral processes; real mobile-money data contain legitimate account-emptying transfers. If this explanation is correct, then every supervised model evaluated here — including the proposed one — has been rediscovering a generation rule rather than learning fraud behaviour, and the measured performance is circular. The models that appear best are simply those that approximate a step function most efficiently, which is why a single threshold beats all of them and why recall is nearly threshold-invariant.

We therefore do not present the performance figures in Table 2 as evidence that the XSEML architecture detects mobile-money fraud. They are evidence that it recovers the structure of this dataset. The distinction is not a caveat; it is the finding.

Implications for Research Using Synthetic Mobile-Money Data

This is a generalization of the present study. One reason for the popularity of synthetic datasets in mobile-money fraud research is that transaction data are difficult to secure, and published synthetic datasets often lack an explanation of how the labels were generated. If they do not, reported performance can be due to rule recovery instead of detection, and the issue cannot be detected by conventional validation because held-out testing, cross-validation and bootstrap intervals are all conducted on the same generated distribution.

We propose three audit procedures, all inexpensive, and all applied above. First, apply single-threshold rules to every candidate feature and compare the results with the proposed model; if one of the rules matches or outperforms the trained ensemble, it is strong evidence of rule-generated labels. Second, check for deterministic boundaries in class-conditional feature distributions, as in Figure 3, which are not expected in real behavioral data. Third, test high-attribution features for genuine discriminative contribution by permutation and ablation, since attribution and contribution may diverge, as with transaction month. For reviewers and editors with an interest in this area, we recommend an audit of synthetic data and an extension of label-provenance reporting wherever documentation is absent.

Social Engineering: What Is and Is Not Claimed

The scope claim is limited by the data, and it is independent of the data-quality problem. Social engineering is a component of off-platform interaction, and none of this — in PaySim, or any other transaction-log dataset located in this work — represents off-platform interaction. To ground truth and establish that a model detects social engineering, rather than merely a transaction that could follow it, requires information that connects specific transactions with known incidents. No such connection exists here. Flagged transactions are those displaying behaviour similar to patterns known to be associated with social-engineering transactions, not necessarily evidence that a social-engineering event occurred. Of the four themes identified qualitatively, two lacked the account identifier needed to operationalize them, one (timing) had an identifier but was not statistically supported (Kolmogorov–Smirnov $D = 0.014$, $p = 0.543$), and one (balance depletion) was statistically supported but may reflect label generation rather than behaviour (Section 3.9).

Limitations

The dataset is synthetic, its label provenance is undocumented, and Section 3.9 provides positive evidence that its labels are rule-generated; the performance figures should not be read as estimates of real-world performance. The absence of an account identifier prevented two of four behavioral themes from being modelled. The qualitative strand is a structured document review and cannot support prevalence claims. SHAP stability is poor ($p = 0.130$), so instance-level explanations should not be presented as precise orderings of causes. The evaluation is internal only: no external validation on independent real data and no production validation were performed. The $F1$ -optimal threshold was selected on the test set and is reported as a sensitivity analysis rather than as a tuned operating point.



Conclusion and Recommendations

This work has adopted an explainable stacking ensemble to learn mobile-money transaction-anomaly detection and compared it with seven baselines using paired significance testing. On a public synthetic M-PESA dataset, the ensemble had F1 of 0.758 and ROC-AUC of 0.831, significantly better than the rule-based and unsupervised baselines, but significantly lower than a tuned XGBoost classifier. Technically the architecture is feasible end to end and generates stable and faithful global explanations, but not at the per-instance level.

The principal result of the study is not this performance, but that the dataset used for evaluation is deterministic, with a label boundary at which no legitimate transfer reaches 90% or more of the sender's balance, and that a single threshold rule on this ratio achieves perfect precision, exceeding all machine-learning models evaluated. The dataset's labels therefore appear to be rule-generated, and the evaluation is circular. We do not suppress this finding but report it as a result, and provide a three-step audit for label provenance in synthetic fraud data that can be reused by other researchers.

Real, consented mobile-money data with an account-level identifier and ground truth linking transactions to verified social-engineering incidents are needed to make substantive progress in this space. Until such data are available, explainable ensemble architectures can be built and their engineering validated, but claims of detecting social-engineering-related fraud cannot be substantiated.

Acknowledgements

The authors thank The Co-operative University of Kenya for a supportive research environment, and thank the anonymous reviewers, whose insistence that the suspected synthetic-data artefact be tested rather than discussed led directly to the central finding reported in Section 3.9.

References

- Boen, C. (2025). M-Pesa transactions fraud [Data set]. Kaggle. <https://www.kaggle.com/datasets/calebboen/mpesa-transactions-fraud>
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101.
- Central Bank of Kenya. (2025). FinAccess survey 2025: Digital financial services and fraud trends. Central Bank of Kenya.
- Creswell, J. W., & Plano Clark, V. L. (2018). *Designing and conducting mixed methods research* (3rd ed.). SAGE Publications.
- GSMA. (2024). Mobile money fraud landscape in Sub-Saharan Africa 2024. GSM Association.
- Kotut, L., Stanley, C., Kimani, S., Mbindyo, P., & Chen, J. (2025). Understanding exploitation disparities in mobile money fraud in Kenya. *Proceedings of the ACM on Human-Computer Interaction*, 9(CSCW1), Article 45.
- Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774.
- Mambina, I. S., Ndibwile, J. D., & Michael, K. F. (2022). Classifying Swahili smishing attacks for mobile money users: A machine-learning approach. *IEEE Access*, 10, 83061–83074.
- Safaricom PLC. (2025). Annual report and financial statements for the year ended 31 March 2025. Safaricom PLC.
- Sarker, S., & Shukla, A. (2023). Research design in machine learning-based fraud detection: A systematic review. *Computers & Security*, 125, Article 103112.
- Schmitt, M. (2024). Digital deception: Generative AI in social engineering and phishing attacks. *Artificial Intelligence Review*, 57(4), Article 89.
- Serianu. (2025). Africa cybersecurity report 2025: Kenya country profile. Serianu Limited.
- TransUnion. (2025). H1 2025 digital fraud trends report: Africa focus. TransUnion.
- Uddin, K. M. M., Rahman, M. M., & Islam, M. R. (2025). Explainable machine learning for phishing site detection using ensemble methods. *IET Journal of Engineering*, 2025(1), Article e70110.
- Vennela, A. (2026). Intelligent cybersecurity systems for phishing attack detection in mobile financial services. *Computers & Electrical Engineering*, 112, 109–125.
- Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 5(2), 241–259.