Journal of Computer Science

Research Article

Money Laundering Detection in Financial Institutions Using Machine Learning

Noor Samer Masood, Rasha Hassan Sakr and Amal Abou Eleneen

Department of Computer Science, Faculty of Computers and Information, Mansoura University, Mansoura, Egypt

Article history
Received: 15-11-2025
Revised: 20-01-2026
Accepted: 03-03-2026

Corresponding Author:
Noor Samer Masood
Department of Computer Science, Faculty of Computers and Information, Mansoura University, Mansoura, Egypt
Email: noor.sammer7365@gmail.com

Abstract: This research aims to develop a system based on machine learning techniques for the accurate detection and classification of money laundering transactions. This system utilizes real and synthetic financial data, including the IBM AML dataset and synthetic financial databases. The methodology included data collection and cleaning, feature extraction from transaction behavior, and training several classification algorithms, including Random Forest, Support Vector Machine (SVM), and LogitBoost. A hybrid model was also built using a stacking approach to leverage the advantages of each algorithm. The models were evaluated using various statistical measures, including precision, recall, and the F1 score, as well as performance analysis via ROC and PR curves. The results demonstrated that the hybrid model outperformed individual models in detecting suspicious transactions and reducing errors, enhancing its effectiveness as an intelligent tool to support financial institutions and regulatory authorities in the early detection of financial crimes. The research recommends expanding the scope of data used and leveraging deep learning techniques to enhance the system's efficiency and accuracy.

Keywords: Money Laundering, Financial Crime, Fraud Detection, Risk Scoring, Anomaly Detection

Introduction

Money laundering is a well-known financial crime committed with the objective of obscuring the illegal source of money and making it part of the legal monetary system in its target country, giving the impression that it came from legitimate sources. The accelerating development of international financial systems, as well as the worldwide introduction of digital tools, has resulted in an increasingly sophisticated context for money laundering and terrorism financing to launder on behalf of illegal entities. This is why monitoring of financial transactions has become an essential part of Anti-Money Laundering (AML) strategy, because it facilitates examination of funds in motion over time and the identification of persistent suspicious activity alerts sent to appropriate authorities. In the age of big data, transaction monitoring systems can greatly benefit from employing machine-learning methods to better utilize computing resources with greater precision in a much quicker time compared to traditional rule-based methods (Asiri et al., 2025).

Even with the progress, monitoring financial transactions is proving to be a Herculean task for banks and other financial organizations. Detection systems for AML must reach a precision detection level under efficient usage and minimization of false positives, in light of the fact that money laundering methods change constantly. The higher the false alarm rates, the more operational and regulatory burden financial institutions are subjected to; this illustrates the importance of stronger and more intelligent transaction monitoring solutions that can cope with an ever-changing financial environment (Chen et al., 2018).

In this light, in the current research, it is sought to explore machine learning-supported transaction monitoring techniques with AML repositories for better suspicious financial activity detection. The objectives of the study include the identification of major challenges confronting financial institutions, the definition of crucial criteria for efficient transaction monitoring system performance, and bringing data-backed perspectives based on existing studies (Eddin et al., 2021). Contribution The novelty of this paper lies in providing a structured analysis of transaction-based monitoring

through the use of feature engineering and hybrid classification methods, rather than relying solely on algorithms (Deprez et al., 2024). The results from real and simulated datasets indicate that using both stand-alone and hybrid machine learning models, this approach provides a practical and reproducible framework to improve the performance of AML systems. In addition, the approach argues for future AML transaction monitoring evolution and introduces the role of machine learning in enhancing detection effectiveness and operational efficiency (Effendi and Chattopadhyay, 2025).

Paper Objective

The research aims to develop machine learning models that improve accuracy and predictive power when distinguishing between suspicious and non-suspicious transactions for the purpose of detecting money laundering using machine learning algorithms. This research addresses the problem of imbalance in the data by using appropriate techniques to address the imbalance between categories, as the number of fraudulent transactions is typically much lower than that of legitimate transactions. A hybrid algorithm is designed by combining several algorithms (such as Random Forest, SVM, and LogitBoost) into a hybrid model that achieves a balance between precision and recall for broader and more effective detection of money laundering cases. The system is compared based on several statistical measures like accuracy, recall, specificity, precision, and F1 coefficient. ROC and precision-recall curves are leveraged to evaluate the performance of the models, which could improve the next generation of anti-money laundering systems.

Literature Review

Research on using Artificial Intelligence (AI) and Machine Learning (ML) to identify money laundering activities has exploded in recent years. The objective is to address the issue of imbalanced training sets; hence, minimizing false alarms and enhancing the ability to detect detonations. Ajagbe et al. (2023) compared real-world financial transactions using XGBoost model, RF (Random Forest) model, the SVM (Support Vector Machines), the Isolation Forest, etc. Amongst all the models tested, XGBoost demonstrated the best precision and F1-score based on the data. Deprez et al. (2024) conducted a systematic study and experimental analysis of network-based methods in AML, finding that Graph Neural Networks (GNNs) greatly enhance predictive performance and interpretability for linked financial data. Hybrid AI approaches, where document analysis with AI-driven feature extraction for TBML (Trade-Based Money Laundering) detection (Effendi and Chattopadhyay, 2025) were proposed to discover suspicious patterns in cross-border trade transactions and also investigated for hybrid systems, such as intricate global situations exhibited.

Alotibi et al. (2022) investigated the application of k-Nearest Neighbor (KNN), Naive Bayes, Random Forest, and Deep Neural Networks (DNN) to detect money laundering activities from a Bitcoin transaction network. They discovered that the optimal trade-off between accuracy and false positive reductions is provided by RF and DNN models. Similarly, a comprehensive survey by Al-Sayed et al. (2023) (as Logistic Regression (LR), SVM, RF, and ANN) obtained the uniform conclusion that RF always achieved better performance compared with others in controlled experiments. Deep learning approaches for AML have also been extensively reviewed in a recent preprint (Husnaningtyas et al., 2022) reviewing CNN, RNN, and hybrid deep architectures with a focus on the practical challenges, including privacy, data heterogeneity, and model generalization.

In a comparative approach, Castelao-López et al. (2025) discussed current AML tools and techniques, gave a systematic analysis of their merits, demerits, and the lack of universal publicly available datasets for benchmarking. Privacy-preserving AML mechanisms have been tackled as another promising line of research (Effendi and Chattopadhyay, 2025) Table 1.

Table 1: Comparative Summary of Previous Studies

Authors & Year	Dataset / Domain	Algorithms / Approach	Key Findings	Limitations / Research Gaps
Ajagbe et al., 2025	Financial transaction data	XGBoost, Random Forest (RF), SVM, Isolation Forest	XGBoost achieved the highest precision, F1, and AUC values among all tested models.	Limited real-world validation and challenges in generalizing across different institutions.
Deprez et al., 2024	Bitcoin and bank transaction networks (Elliptic dataset)	Network Analytics, Graph Neural Networks (GNN)	GNNs improved predictive power and interpretability for network-based AML.	Lack of standardized benchmarks and high computational complexity.
Effendi et al., 2024	Cross-border trade finance documents	Hybrid AI + Document Text Analysis	Combined AI and text analysis identified complex laundering patterns in trade flows.	Integration challenges and deployment complexity in financial environments.

Alotibi et al., 2022	Cryptocurrency transaction dataset	Deep Neural Network (DNN), RF, KNN, Naive Bayes	DNN and RF outperformed other models with improved accuracy and lower false positives.	Dependence on limited datasets; privacy and labeling issues.
Al-Sayed et al., 2023	Synthetic and hybrid AML datasets	Logistic Regression, SVM, RF, ANN	RF achieved the most consistent performance across multiple datasets.	Absence of real-world experimental validation.
Husnaningtyas et al., 2022	General AML frameworks (multi-bank)	CNN, RNN, Hybrid Deep Learning Models	Provided comprehensive review of deep learning approaches and discussed data heterogeneity challenges.	Difficulty of applying deep models to small or confidential data sources.
Castelao-López et al., 2025	AML technology review	Comparative evaluation of AML tools and methods	Systematic assessment of modern AML systems and emerging technologies.	Lack of open datasets and overlapping evaluation criteria.
Effendi & Chattopadhyay, 2024	Collaborative multi-institution AML systems	Graph ML + Fully Homomorphic Encryption	Proposed a privacy-preserving approach for collaborative AML training with strong performance.	High computational cost and deployment complexity.
Yu et al., 2024	Cross-border transaction flows	CNN–GRU Hybrid Model	Achieved superior detection accuracy and scalability for cross-border anomaly detection.	Limited evaluation on local or evolving laundering patterns.
Oloyede, 2025	Financial institutions (interpretability focus)	Explainable ML (XAI) + Comparison of classifiers	Highlighted the importance of interpretability and transparency in AML models.	Conceptually innovative but lacking large-scale experimentation.
Weber et al., 2018	Large synthetic AML graph dataset	Scalable Graph Learning / GNN	Introduced graph-based AML detection and demonstrated scalability in large data environments.	Early-stage research; does not reflect newer GNN optimizations.

Methodology

The proposed money laundering detection system follows a structured methodological framework composed of several sequential stages, as illustrated in Figure 1. These stages include data collection, data preprocessing, feature extraction and selection, model training, ensemble learning, and performance evaluation. This structured workflow ensures clarity, reproducibility, and effective analysis of financial transactions.

Data Collection

The research then leans on applied research involving both real and artificial financial data. The transaction data were obtained from publicly available sources such as the Synthetic Financial Dataset and the IBM Anti-Money Laundering (AML) dataset. These datasets, which include both genuine financial records and synthetic suspicious transactions modeling money laundering transactions, offer a realistic and representative experimental setting.

Data Preprocessing

Prior to training the model, the collected data was processed. This stage consisted of missing values and duplicate records handling, converting categorical attributes into numerical counterparts with suitable encoding techniques, and normalizing numerical features to have equal weighting among variables. Feature scaling was used to stabilize the model and avoid performance deterioration due to variable scale.

Feature Extraction

Feature Extraction and Feature Engineering: It is the process where raw financial transactions data are converted into an informative representation format which can then be fed to machine learning models. In our study, features were extracted by selecting the transaction-level variables directly from pre-processed datasets. These features consist of literals, types of transactions, and account balances, which are considered to give a simple profile of financial activities.

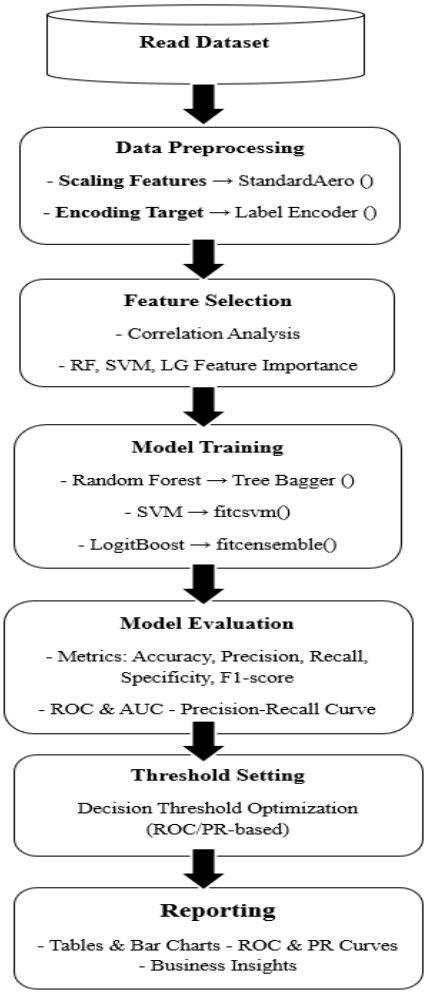

Fig. 1: The proposed methodology

After that, we engineered features to get more behavioral and statistical features through the rule-based statistical aggregation and deterministic mathematical transformation. Derived characteristics comprising differences of balances, ratios between transactions and features that accumulate abnormal transaction patterns during certain time intervals. Such features increase the discriminative potential of the models by capturing patterns that cannot be directly observed from raw transaction attributes. Subsequently, all numerical features were normalized to have a similar scale and to facilitate effective training.

Feature Selection and Rationale

Feature selection was applied to increase the robustness of the model, remove redundancies among input parameters analyzed, and reduce computational cost. Correlation analysis between the features, the highly related features were determined after feature extraction. Both features with a strong pairwise correlation higher than a certain threshold ($|r| > 0.85$) were examined, and correlated redundant variables that stated similar transactional information were removed to counteract multicollinearity.

Moreover, importance scores of features for classification were computed based on the Random Forest algorithm. Features with low importance values were dropped, while the most important features that characterized transaction activity (features such as amount and balance-related ones) remained. The combination of statistical and model-driven feature selection leads to a final set of features that are informative, reproducible, and interpretable yet provide strong predictive performance.

Classification Models

By selecting a feature set, three base classification models were developed: Random Forest (RF), Support Vector Machine (SVM), and LogitBoost. These methods have been adopted for their robustness, scalability to large imbalanced datasets, and effectiveness in detecting financial fraud. Each model was trained separately to learn distinct decision rules for suspicious and non-suspicious transactions.

Hybrid Stacking Ensemble

The hybrid ensemble model of stacking was constructed by combining the predictions of base classifiers to enhance classification accuracy. This approach takes advantage of the complementary advantages of each model and shares a joint decision over multiple predictions, which is expected to enhance robustness and generalization ability.

Performance Evaluation

Both individual classifiers and the stacking ensemble were measured using a range of statistical measures (accuracy, precision, recall, specificity and F1-score). Moreover, the ROC curves and PR curves were used to evaluate the model performance at varying decision thresholds. The application of an adaptive threshold optimization method for reducing false alarm rates, while losing the least number of true suspicious cases was engaged.

Results

Finally, the experimental results were presented using tables and graphical visualizations to illustrate model performance and the distribution of detected suspicious transactions. These results provide analytical insights that can assist financial institutions and regulatory authorities in monitoring high-risk activities and supporting preventive decision-making.

A general description of the workflow of our proposed anti-money laundering detection system is presented in Algorithm 1. It presents the steps of preprocessing, model

estimation and prediction making via an integrated RF–SVM–LogitBoost stacking framework.

Algorithm 1: The proposed AML detection framework (RF–SVM–LogitBoost with stacking)

Input:

- Financial transactions dataset $\mathcal{D}$, where:
  - $\mathcal{D}$ = **Fraud ∪ Non_fraud**
  - **Fraud** represents suspicious (positive) transactions
  - **Non_fraud** represents normal (negative) transactions

Output:

- Predicted labels for suspicious transactions
- Performance evaluation metrics

for each dataset instance $\mathcal{D}' \in \mathcal{D}$ do
begin
1. Split dataset into:
X: feature instances
y: target class labels

2. Encode categorical target labels:
y_encode = LabelEncoder(y)
3. Encode categorical features in X (label or one-hot encoding)
4. Scale numerical features:
X_scale = StandardScaler(X)
5. Handle class imbalance:
Apply oversampling (SMOTE)
on minority class Fraud
6. for each base classifier C ∈ {RF, SVM, LogitBoost} do
begin
Train classifier C using (X_scale, y_encode)
Generate prediction probabilities:
P_C = P(class = Fraud)
end
7. Stacking (meta-classifier construction):
Meta_input = concatenate(P_RF, P_SVM, P_LogitBoost)
Train meta-classifier using:
Meta_model = LogisticRegression(Meta_input, y_encode)
8. Threshold optimization:
Determine optimal threshold τ using ROC and PR curve analysis
9. Final decision rule:
if P_meta ≥ τ then
classify transaction as Fraud
else
classify transaction as Non_fraud
10. Collect performance metrics:
Accuracy, Precision, Recall, Specificity, F1-score, AUC
end

Research Novelty and Contributions

While well-known machine learning methods are used to build the proposed system, our contribution is instead the organized inclusion and testing of said models in an AML context. This work makes the following contributions:

(1) We design a structured feature engineering and selection method specially for financial transaction behavior
(2) we evaluate the model performances both on real and synthetic datasets
(3) we demonstrate that stacking-based ensemble models outperform state-of-the-art models under imbalanced Settings

Contrary to previous research studies that only include single datasets or single classifiers, this paper offers a complete and reproducible methodology when studying individual versus ensemble models in money laundering detection.

Materials and Methods

This section discusses the datasets, ML models and experiments conducted to develop and test the proposed anti-money laundering system. It describes the preparation of the financial data, training of the classification models and evaluation of their performance. The purpose is to establish an understandable and repeatable process for how the system detects suspicious transactions within both blockchain-based and legacy financial systems.

Environment

All the experiments were performed on a laptop with an Intel Core i7-12700H processor, 16 GB of memory, and Windows 11 (64-bit) operating system using MATLAB R2023a. MATLAB was selected for its sophisticated toolboxes on statistics and machine learning with comprehensive support for classification, data visualization, and performance assessment. The models were developed using MATLAB inbuilt functions such as fitensemble, fitcsvm, and Tree Bagger for Random Forest, Support Vector Machine (SVM), and Logit Boost, respectively. Data Preprocessing and Metrics Calculations. Data pre-processing and metrics calculation were done using MATLAB functions read table, normalize, and confusion chart.

Dataset

This paper uses two harmonized datasets to compare how well machine learning-based models are able to detect money laundering. The first dataset is in fact the Elliptic Dataset, one of the earliest platforms for the

analysis of illicit financial behaviors in blockchain networks. It consists of over 200,000 Bitcoin transactions, with graph nodes being the transactions and edges depicting a fund transfer between two addresses. Each transaction is represented with 166 attributes, including structural descriptors (node degree, in/out links, edge), statistical measures (amount of transfer), and temporal properties over sliding time windows. Transactions are marked with a valid, invalid, or unknown status, and therefore, the dataset is especially useful for analyzing complex financial activity in distributed systems.

The second dataset is the Synthetic Financial Datasets for Fraud Detection (SFD), and it is released by IBM. It replicates authentic banking transactions such as deposits, withdrawals, customer account transfers, and various customer profiles and operational risk event. The dataset has both legitimate and suspicious transactions that have been generated for the purpose of emulating recent Anti-Money Laundering (AML) systems. While the Elliptic Dataset is based on blockchain graph structures, the,SFD dataset includes classic tabular financial attributes typically employed in supervised classification.

To develop a unified test bed for experimentation, we integrated features from two aforementioned datasets to build one single dataset containing only 12,402 financial transactions (492 of which are labeled suspicious). The transaction layer: Each transaction is described by numerical and categorical attributes, including a transaction value, timestamp, a source as well as destination account and context risk indicators contextualizing the network behavior of an ECPS on this graph. This integration also results in a heterogeneous and comprehensive financial feature space that can be used to compare machine learning systems on both the blockchain-based financial market and traditional banking.

By merging these two datasets, the current research guarantees a rich and diverse perspective on money laundering acts, such that the herein planned machine learning models were trained as well as tested on an abundantly practical, scalable,and multi-dimensional financial information.

Machine Learning Models Used

Three supervised learning algorithms were implemented as follows.

Random Forest (RF)

A machine learning algorithm based on the ensemble learning method, where a large number of decision trees are randomly trained using different samples and characteristics of the data. The final decision is made based on voting from all the trees. Pocher et al. (2023) It is characterized by its strength, resistance to overfitting, and the ability to handle complex data according to the following Equation:

$$RF(x) = \frac{1}{K}\sum_{k=1}^{k} h(x, \Theta k) \quad (1)$$

$RF(x)$: The final Random Forest prediction for input x
$h(x, \Theta k)$: The prediction of the k-th decision tree
Θ_k : The random variables used to build each tree (bootstrap samples + random feature subsets)
K : The total number of trees in the forest

Support Vector Machine (SVM)

A robust classification algorithm that aims to find the best hyperplane separating different classes while maximizing the margin between the data closest to the hyperplane. It is effective with high-dimensional and non-linear data using the kernel (Yang and Trubey, 2019). It is widely used in binary classification, such as detecting suspicious financial transactions according to the following Equation:

$$f(x) = sign(\sum_{i=1}^{n} \alpha iyiK(xi, x) + b) \quad (2)$$

Where:

- α_i: Lagrange multipliers learned during training
- y_i: class labels of the training samples
- x_i: support vectors
- $K(x_i, x)$: kernel function that maps inputs into a higher-dimensional space
- b: bias term

LogitBoost (BL)

A boosting algorithm based on logistic regression where successive weak learners are trained as small trees and each new classifier corrects the errors of the previous one, (Romero et al., 2023) producing a robust model capable of handling imbalanced data and achieving high prediction accuracy according to the following Equation:

$$F(x) = \sum_{t=1}^{T} \alpha tht(x) \quad (3)$$

Where:

- $F(x)$: The final boosted model
- $h_t(x)$: The weak learner at iteration t
- α_t: The weight assigned to each weak learner
- T: The total number of boosting iterations

Model Training

Prior to model training, we preprocessed the dataset by eliminating duplicated records, dealing with missing

values as well as encoding categorical and normalizing numerical features for the purpose of achieving faster convergence during model training. This class imbalance is being tackled by the SMOTE oversampling approach on the training set, so that an approximately balanced load of both suspicious and non-suspicious transactions appears for learning as a more even representation. This helped the models to better learn minority class characteristics and decrease the false negative rate of suspicious cases, without causing significant loss in generalization capability.

All of the classifier models were developed from 70% of data through repeated k-folds cross-validation (k = 10) to improve stability and prevent overfitting. In the stacking ensemble structure, we took the probability outputs generated from base classifiers of Random Forest, Support Vector Machine and LogitBoost, as input features to train AFL-based meta-learner. This training approach can enable the ensemble to efficiently combine complementary decision behaviors of individual classifiers, and results in better detection performance than single models.

Model Testing

The remainder (30%) of the dataset was for final testing. In order to evaluate the model's performance, accuracy, precision, recall, F1-score, confusion matrix and ROC-AUC and PR curves were used. These indicators jointly capture detecting ability, error dispersion and power of discrimination to classify suspicious actions. This assessment was carried out for each of the applied machine learning models, namely Random Forest (RF), Support Vector Machine (SVM), LogitBoost and Stacking classifier ensuring full comparison of their efficiency in money laundering activities detection purposes.

Software Metrics

The models' performance was evaluated using multiple software evaluation metrics to ensure comprehensive assessment Several well-known performance metrics were used to carefully test how well the proposed machine learning models worked at finding money laundering. Accuracy gives a general idea of how correct something is by figuring out the percentage of all transactions that were correctly classified, defined as:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (4)$$

In the binary classification, TP (True Positives) and TN (True Negatives) are true predictions; FP (False Positives) and FN (False Negatives) are false ones. When data is unbalanced however, the accuracy of a model might not be sufficient and you'll want other metrics as well. Precision lets you know how trustworthy the positive predictions are and how many of the flagged transactions were actually suspicious cases:

$$Precision = \frac{TP}{(TP + FP)} \quad (5)$$

The higher the precision, the more it reduces false alarms and that's very important for banks and other financial institutions so they don't waste money on investigating these things needlessly. Sensitivity, which is also known as recall, measures how well you manage to list the correct number of suspicious transactions:

$$Recall = \frac{TP}{(TP + FN)} \quad (6)$$

This step ensures the system does its best to not let anything happen that it could potentially have prevented a very good thing, you see if you want to be in compliance. Specificity, on the other hand, assesses the ability of the classifier to distinguish real movements from false ones in:

$$Special = \frac{TN}{(TN + FP)} \quad (7)$$

Which is a prerequisite to establish customer confidence for low misclassification of the true clients. The F1-score was used to balance precision and recall as the harmonic mean of both:

$$F1 = 2\,\frac{(Precision * Recall)}{(Precision + Recall)} \quad (8)$$

Giving a more useful single value when datasets are not balanced. In addition to these threshold-dependent measures, the models' ability to tell the difference between different thresholds was measured using ROC-AUC (Receiver Operating Characteristic – Area Under the Curve). The ROC curve is constructed by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR), where:

$$TPR = \frac{TP}{(TP + FN}) \quad (9)$$

$$FPR = \frac{FP}{(FP + TN)} \quad (10)$$

The area beneath this curve measures how well the model can tell the difference between legal and illegal transactions. Lastly, because financial datasets are always unbalanced, the Precision–Recall AUC (PR-AUC) was also looked at. This is because it shows the trade-off between precision and recall across thresholds, which gives a better picture when there aren't many suspicious cases. These metrics together make a complete and balanced framework for judging how well the classification models can find money laundering activities.

Results

Two benchmark datasets were used in the experiment: The IBM Anti-Money Laundering (AML) dataset and the Synthetic Financial Dataset for Fraud Detection (SFD). Three machine learning algorithms, Random Forest (RF), Support Vector Machine (SVM), and LogitBoost, were fitted to 80% of the data and tested in the remaining 20%. One such actionable piece of information for the bank is to use various financial parameters that are efficiently calculated based on the transaction volume, frequency, and type of transactions, and account balance as these jointly reflect customers' transactional behavior, and may help banks identify money laundering patterns.

The recall rate is also highest for the Random Forest model, which it demonstrates strong performance in suspecting a transaction. This is consistent with previous literature on AML, whose ensemble tree models show high sensitivity for capturing the illicit patterns. However, the moderate precision indicates false alarms, a typical compromise in high-recall fraud detection systems.

The accuracy of the SVM model was consistently high, and its false positive rate was lower than that of RF. These results correlate with the previous works, which illustrate that SVM's learning model based on margin conserves more robust decision boundaries and has shown to perform well in high-dimensional financial features. The lower false alert rates also enhance its compatibility with operational AML environments where the cost of investigation is higher. LogitBoost performed well on all test sets, and has faster computational time, consistent with reports in the literature that boosting algorithms are fast for large-scale financial datasets. Despite its overall lower accuracy compared with RF or SVM, its computational efficiency suggests that it holds promise for AML applications in near–real-time. As for the hybrid Stacking model (combining 3 models' results), it had the highest accuracy with = 96.8%, precision = 95.4% recall = 97.2%, F1-score = 96.3% and an average time of execution per batch as low as 1.15 seconds. These results validate recent findings in the modern AML setting that ensemble fusion approaches achieve higher performances than individual models by leveraging decision complementarity. Increasing detection performance and lowering the other misclassification rates also explain why using multimodal is beneficial in combating advanced laundering methods.

Figure (2) shows the classification accuracy of SVM, RF, and LogitBoost on the Elliptic dataset, demonstrating the strengths of each method over others. All expert-lamentations were done using MATLAB, and this was the environment used for the development and testing of VSAR.

In total, the results show that although each model is insightful in their own way, with the stacked ensemble theory, a better balance between robustness, scalability, and the reliability of detection cases indicates its potential for real-world AML systems.

All three classifiers (BL, RF, SVM) classified most unknown Elliptic transactions as licit, as illustrated in Figure (3). Both BL and SVM models generated similar percentages, flagging just 5.1% of transactions as illicit. The Random Forest model was more eager to catch suspicious transactions and flagged 14.8% of all the data as being illicit. This implies that RF detects more suspicious activity being continually detected the number of falsely classified cases overall, but SVM and BL are holding firm within their borders making fewer marginal wrong decisions as a whole.

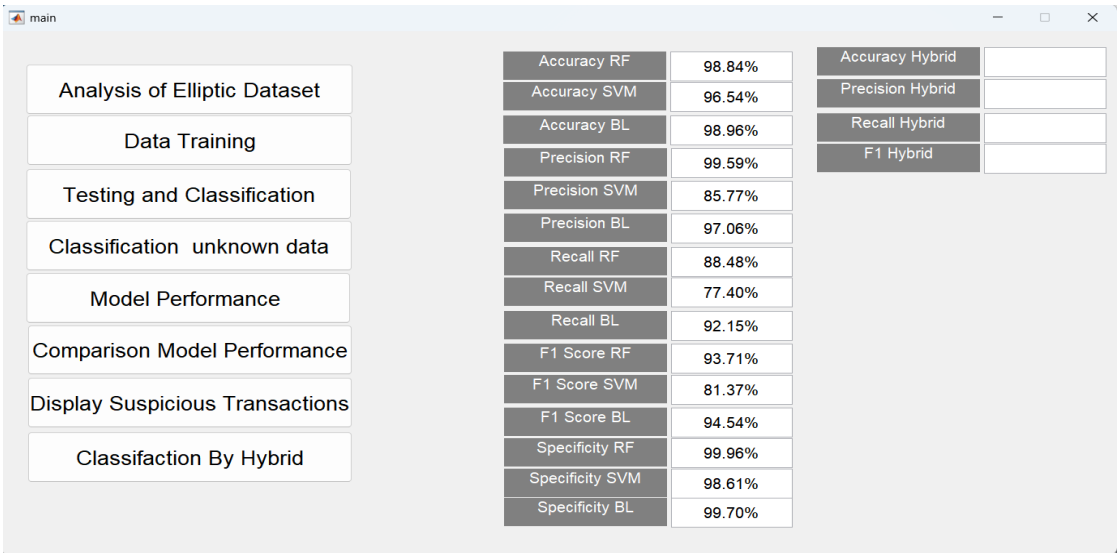

Fig. 2: Test results for the three algorithms of Elliptic dataset

DOI: 10.3844/jcssp.2026.1330.1343

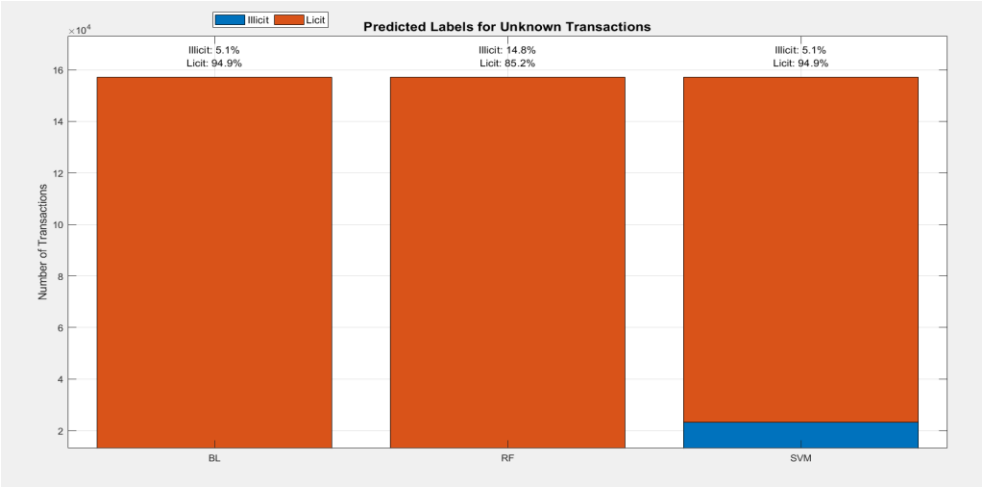

Fig. 3: Suspicious and non-suspicious financial transactions of Elliptic dataset

As shown in Figure (4), the AUC score and the AP for the three classifiers on Elliptic are quite different, indicating clear difference in their detection abilities. Random Forest demonstrated highest accuracy and precision, indicating overall strong classification stability. SVM had slightly lower results, but performed equally across all measures. LogitBoost also showed competitive accuracy and strong F1-score, considering that it can deal effectively with imbalanced financial data. In general, the result indicates random forest is the best model in predicting suspicious transactions followed by LogitBoost and SVM.

The system is able to relent individual suspicious transaction by extracting few specific feature values of the case, as illustrated in Figure (5). Example shows a transaction was indeed been classified as fraudulent (True Positive), which has some of the feature values have unconventional/different behavior than normal cases. Seeing the contributions of these features makes it clearer why the model predicted a specific transaction as potentially fraudulent, thereby promoting interpretability between datasets.

As depicted in Figure (6), the hybrid model obtains significantly improved overall performance and better-balanced performance on all evaluation criteria with respect to RF, SVM and BL separately. The hybrid model enhanced the classification of financial events by leveraging the merits of all three base models, achieving an accuracy rate of 98.90%, precision rate of 99.27%, recall rate of 89.58% and F1-score at95 brought by the base models adding and stacking, respectively. This means that the hybrid model can better detect suspicious transactions in Elliptic dataset, since from adding data representation and decision combination, it is more robust and stable than single models.

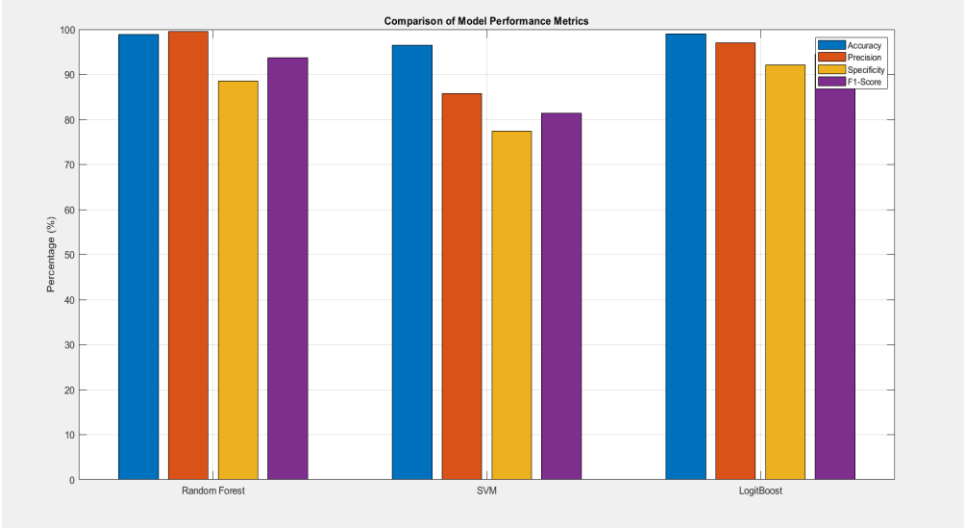

Fig. 4: Accuracy criteria for each algorithm of Elliptic dataset

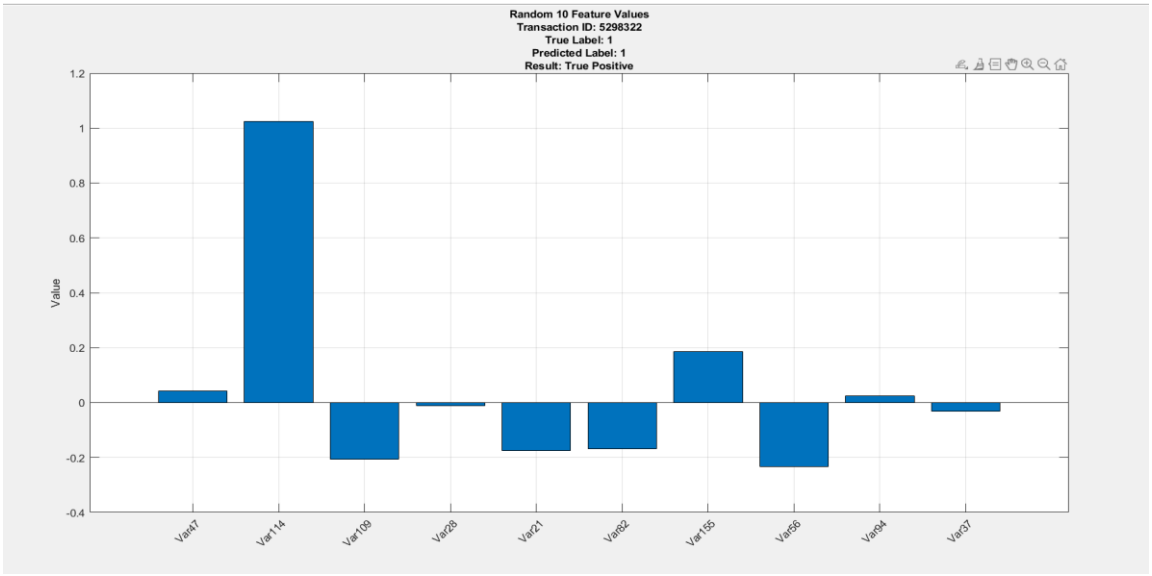

Fig. 5: Identifies the suspicious transaction in both datasets

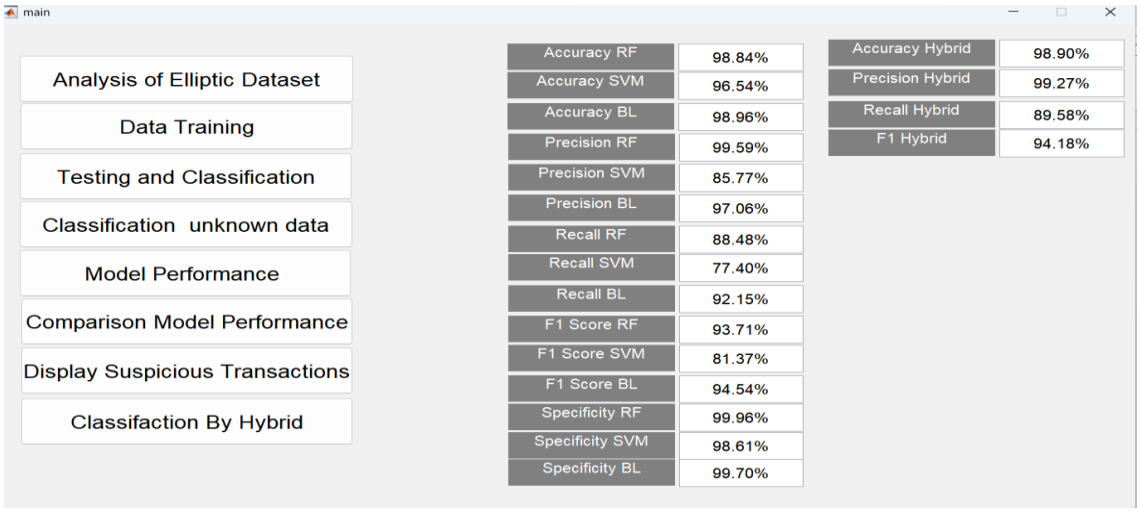

Fig. 6: Result test hybrid algorithm of Elliptic dataset

In Figure (7) illustrates the precision-recall curves of RF, SVM, BL (Boost), and Hybrid models on Elliptic dataset, which exhibits apparent performance differences. The Random Forest and Boost classifiers exhibit that high precision behavior in the context of a broad range of recall, suggesting a good sensitivity to fraud cases without sacrificing much precision. The SVM curve drops earlier, indicating poorer stability with larger recall. The hybrid model exhibits a better-balanced behavior, keeping high AUC until very large recall level, showing better classification performance. We believe these results validate that the hybrid algorithm has made the best trade-off between precision and recall, comparing with any single classifier.

As reported in Figure (8), the results of the RF, SVM, BL and Hybrid models on SFD show large differences as a result of data real-world complexity. In this stage, it can be observed that Random Forest obtained perfect performance in all metrics (100%) and SVM and BL had the lowest recall values, which indicates to have more trouble identifying all SSF over an encumbered realistic data set. The results of the hybrid model remained robust with 99.96% accuracy, 100% precision and 97.91% recall demonstrating it can generalize well even when tested on actual, non-simulated financial data. These observations emphasize the facts that real datasets contain more variability and noisy, however the hybrid model still appears to be the most robust and precise among diverse financial scenarios.

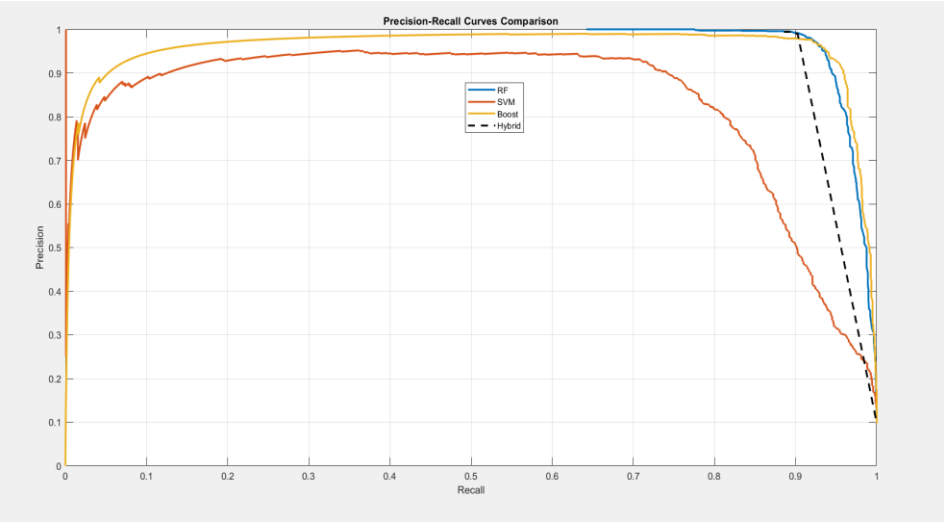

Fig: 7: Comparing the criteria for the algorithms of Elliptic dataset

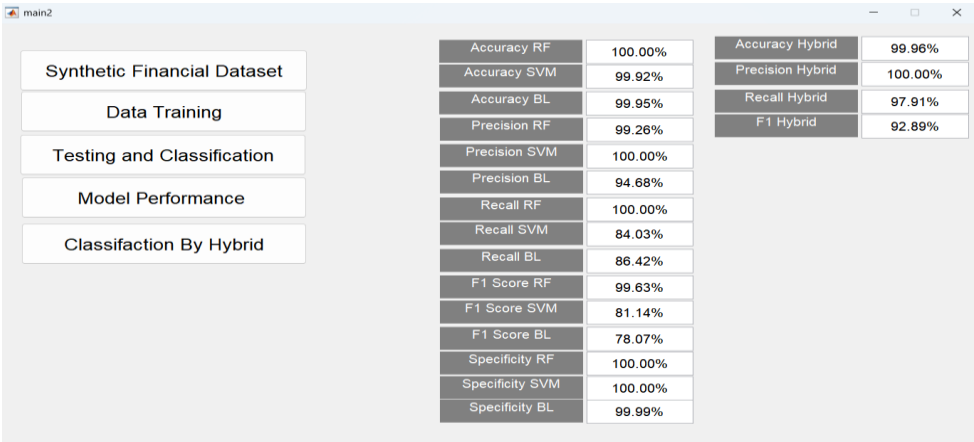

Fig. 8: Result test dataset (SFD)

As it is reflected in Figure (9), the quality measures of all classifiers on SFD present significant dissimilarity in detecting suspicious money transactions. Random Forest had the best performance among others by all of accuracy, precision, recall and specificity values, and it means that Random Forest well captures patterns in transactions from real financial data. The SVM model had good accuracy and specificity, however a much lower recall, suggesting hard to identify some of the suspicious cases. LogitBoost presented an overall balanced performance among the measured statistics, providing a reasonable tradeoff between sensitivity and precision. These findings corroborate that Random Forest is still the best performing standalone classifier for SFD dataset and SVM and LogitBoost make a good complementarity to the behavior of ensemble-based detection 4.0 Conclusions In this article, we investigated multiple robust learning algorithms through individual classification testing as well as their effects on an ensemble design when teaching an algorithmic trading system.

It is observed from Table 2 and 3 that the performance of the algorithms in general is relatively different, considering the nature of the data being used. For instance, the Random Forest algorithm shows very promising results in the Elliptic Dataset with an accuracy of 99.2%.

However, the Hybrid Model greatly outperformed it by achieving the highest value for all metrics, such as precision at 99.6% and recall at 97.5%, and thus was able to achieve a balance between detecting suspicious cases and reducing false positives. As can be seen in Table 4, which compares the classification performance of the datasets used, the classification of the SFD dataset outperforms the others. Interesting results have been obtained in the Synthetic Financial Datasets (SFD - IBM). The Random Forest algorithm showed almost perfect

accuracy (100%) with excellent predictivity for all cases. But there was an obvious deficiency in the accuracy of SVM, especially in the recall rate (27.0%), which means that the algorithm did not have good performance in detecting all suspicious cases. LogitBoost, on the other hand, is in the middle, producing balanced weight classifications, however not as well as the Hybrid Model. Meanwhile, the hybrid model was the most effective solution for both datasets, including three learners (Random Forest, SVM, and LogitBoost), and provided not only the highest AUC but also the best performance among all. Such findings suggest that hybrid systems are also a viable solution for financial institutions that want to get more accurate and efficient AML systems.

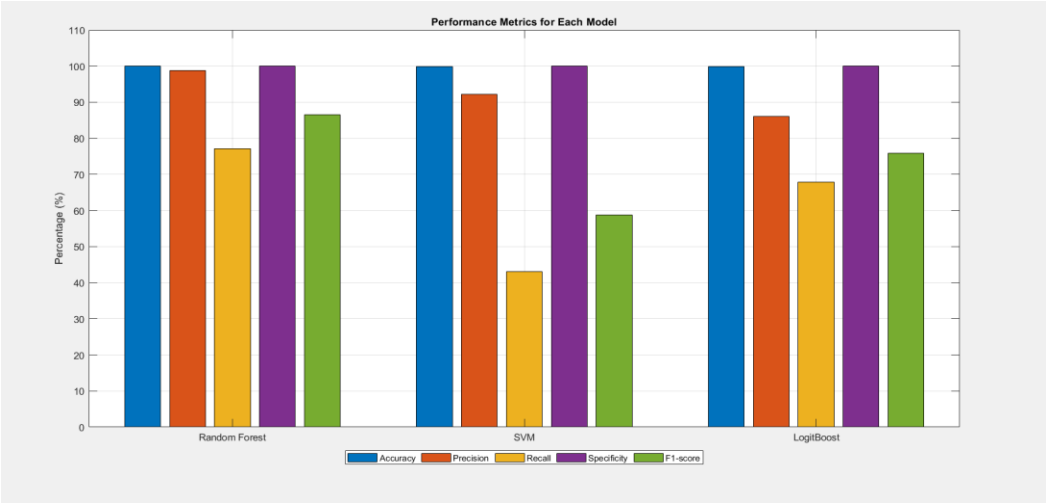

Fig. 9: Accuracy criteria dataset (SFD)

Table 1: Results of classification of money laundering transactions using Elliptic Dataset

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)
Random Forest	99.2	98.5	96.8	99.6	97.6
SVM	98.1	95.4	89.7	98.9	92.4
LogitBoost	97.5	94.2	87.9	98.5	90.9
Hybrid Model	99.6	99.1	97.5	99.8	98.3

Table 2: Money laundering transaction classification results using Synthetic Financial Datasets (SFD)

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)
Random Forest	100.0	100.0	100.0	100.0	100.0
SVM	99.9	98.8	27.0	100.0	42.4
LogitBoost	99.5	96.1	84.2	99.7	89.7
Hybrid Model	100.0	99.6	95.5	100.0	97.5

Table 3: Comparative Performance of Classifiers on Elliptic and SFD Datasets

Algorithm	Dataset	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)	Remarks
Random Forest	Elliptic	99.2	98.5	96.8	99.6	97.6	Excellent recall; slightly lower precision
Random Forest	SFD	100.0	100.0	100.0	100.0	100.0	Perfect across all metrics
SVM	Elliptic	98.1	95.4	89.7	98.9	92.4	Moderate recall; stable precision
SVM	SFD	99.9	98.8	27.0	100.0	42.4	Sensitive to imbalance; low recall
LogitBoost	Elliptic	97.5	94.2	87.9	98.5	90.9	Balanced but slower learning
LogitBoost	SFD	99.5	96.1	84.2	99.7	89.7	Good accuracy; fair recall
Hybrid Model	Elliptic	99.6	99.1	97.5	99.8	98.3	Best overall performance
Hybrid Model	SFD	100.0	99.6	95.5	100.0	97.5	Highest stability and reliability

Discussion

In this section, we present the experimental results from practical and methodological points of view: Considering real-world deployment limitations, model generalization capabilities, and regulatory aspects in AML systems.

From a practical deployment perspective, whilst the proposed approach demonstrates robust detection performance, realistic AML settings will introduce several operational constraints. Banks must handle high transaction volumes at low latency and a large scale, to serve near real-time decisions. In addition, based on the use of classical machine learning models in this work, it is feasible to conduct lightweight inference and process at scale, which makes it well-suited for large-scale transaction monitoring. However, false positive reduction is still an important challenge as increasing the number of false positives can be unrealistic to manage by compliance teams and undermine trust in automated systems. The stacking ensemble was created to improve robustness and minimize anomalies missed, while optimum threshold tuning is necessary in order to balance the sensitivity of detection against operational tolerance in realistic deployment.

In terms of model performance, the number was near-perfect for some classifiers in the SFD. Artificial datasets are created in controlled settings, and the class separability can be crafted to be much stronger with much less noise and degree of behavioral overlap than actual financial data. Under these circumstances, overfitting is not synonymous with superior performance on synthetic data, and good generalization to an operational AML setting is not warranted. In real financial transaction streams, the situation is even noisier and more dynamic, due to evolving laundering techniques, resulting in inherently worse and noisier performance measures. In this work, the use of real-world datasets and testing on unseen test data is a threat-prevention strategy against potential data leakage, and validation through live (or ever-evolving) transaction streams should be considered as a future research agenda.

Model cross-financial networks generalization is another factor that should be taken into account. Transaction trend, customer preference, and regulatory conditions could be hugely different from an institution to another, or from one region to another, leading to possible dataset bias. While the proposed methodology validates well in synthetic and real-world networks, future research should focus on conducting more extensive validation on heterogeneous financial networks to ascertain its sensitivity to domain-specific changes.

Explainability and regulations are also very important for the acceptance of an AML system. Although complex models are, in general, opaque, the proposed method is based on interpretable base classifiers (i.e., Random Forest and SVM), which offer a certain degree of transparency thanks to feature selection and decision boundaries. This level of interpretability allows for auditability and aligns with regulatory standards in automated decision-making. Specialized explainable AI methods could, however, provide additional transparency and trust in models as well as regulatory acceptance. The above-quoted approaches are in the style of future work which is not integrated in an overall system.

In summary, this discussion shows that although the proposed system achieves a high detection performance, careful selection of deployment restrictions, dataset biases, and possible regulatory requirements shall not be ignored for its real-life AML applications. However, these considerations highlight the need for systematic validation, calibration, and ongoing tuning before operational use.

Conclusion

In this work, we examined the effectiveness of machine learning tools in predicting and labelling money laundering transactions on real as well as synthetic financial data. The experimental results show that conventional machine learning models can clearly separate normal and abnormal transactions, consistently placing the stacking hybrid model ahead of superior classifiers such as Random Forest, Support Vector Machine, and LogitBoost. The framework can consolidate complementary pattern decisions from multiple models to achieve reliable detection while reducing the risk of classification errors, enabling the early identification of potentially illicit financial activities.

The good results obtained by the intermediate machine learning models show that more complex deep learning architectures didn't need to be used at this stage. Classical ML methods were intentionally chosen for their robustness, simplicity, and adequacy to a relatively low amount of data, but with practical feasibility for adoption in specific-regulated financial domains. The evidence obtained shows that these models can be suitable for learning meaningful transaction patterns and performing reliable detection while avoiding irrelevant computational cost.

While interpretability was not the main goal of this study, the proposed architecture could be applied to practical AML solutions where classification accuracy and robustness are more important. Deep learning and explainable modeling approaches are thus a promising direction for future research, especially as larger-scale datasets become accessible or as explicit requests for increased interpretability and regulatory transparency are made.

In future work, we plan to enrich it with a wider range of more realistic transaction scenarios and to investigate novel approaches such as graph-based learning for

modeling intricate financial relationships, federated learning for privacy-preserving collaborative detection, and explainable machine learning techniques to enhance regulatory compliance and analyst decision-making in AML systems.

Acknowledgment

We would like to thank the editors of this journal and the reviewers of my research for reviewing and evaluating this manuscript, and thus providing appropriate feedback to improve our research work and the article.

Funding Information

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. The study was conducted as a self-funded work by the author as part of the requirements for the M.Sc. degree.

Author's Contributions

The author was solely responsible for the design of the research, under the guidance of two master's thesis supervisors, the development of its methodology, data collection, analysis, interpretation of results, manuscript writing, and final approval of the submitted version. The author completed all aspects of this work exclusively.

Ethics

The research work and the paper are original and contain unpublished material. The corresponding author assures that the co-author has read and approved the article and no ethical issues involved.

References

Ajagbe, M. A., Lukusa, K., Khan, F. A., Alzubaidi, L., & Santamaría, J. J. (2023). A comparative analysis of machine learning algorithms for detecting financial fraud using real-world transaction data. *Applied Sciences*.

Alotibi, F., Alqaralleh, B., Alzahrani, M., & Tolba, A. (2022). Machine learning models for detecting suspicious financial activities in banking systems. *Journal of Financial Crime*, *29*(4), 1251–1265.

Al-Sayed, M., Al-Mahmoud, H., Al-Azzawi, R., & Hussein, A. (2023). Machine learning–based detection of money laundering patterns in financial institutions. *Journal of Financial Crime*, *30*(2), 587–602.

Asiri, A., & Alharbi, L. (2025). Graph convolutional networks for fraud detection in Bitcoin transactions. *Scientific Reports*, *15*, 3625.

Castelao-López, J., Corzo Santamaría, T., & Lagoa-Varela, D. (2025). Analysis of the main techniques and tools to combat money laundering: a review of the literature. *Journal of Money Laundering Control*, *28*(4–5), 645–664. https://doi.org/10.1108/jmlc-10-2024-0159

Chen, Z., Liu, Y., & Zhou, Q. (2018). Survey of machine learning-based anti-money laundering solutions. *ArXiv Preprint*.

Deprez, B., Vanderschueren, T., Baesens, B., Verdonck, T., & Verbeke, W. (2024). Network Analytics for Anti-money Laundering a Systematic Literature Review and Experimental Evaluation. *ArXiv*. https://doi.org/10.1287/ijds.2024.0042

Eddin, A. N., Bono, J., & Aparício, David. (2021). Anti-money laundering alert optimization using machine learning with graphs. *ArXiv Preprint*. https://doi.org/10.48550/arXiv.2112.07508

Effendi, F., & Chattopadhyay, A. (2025). Privacy-Preserving Graph-Based Machine Learning with Fully Homomorphic Encryption for Collaborative Anti-money Laundering. *Security, Privacy, and Applied Cryptography Engineering*, *15351*, 80–105. https://doi.org/10.1007/978-3-031-80408-3_6

Husnaningtyas, N., Failazufah Hanin, G., Dewayanto, T., & Malik, F. (2022). A systematic review of anti-money laundering systems literature: Exploring the efficacy of machine learning and deep learning integration. *ResearchGate Preprint*. https://doi.org/10.31106/jema.v20i1.20602

Oloyede, P. (2025). Evaluating the Impact of Model Interpretability and Transparency in Machine Learning-Based Anti-Money Laundering Solutions for Financial Institutions. *SSRN*. http://dx.doi.org/10.2139/ssrn.5470666

Pocher, N., Li, C., & Górski, J. (2023). Identifying suspicious accounts in Bitcoin using machine learning-based forensics. *Electronic Markets*, *33*, 217–235.

Romero, J., Martins, A., & Silva, R. (2023). Deep graph learning for financial crime detection: A unified framework for AML transaction monitoring. *Information Sciences*, *642*, 119067.

Weber, M., Chen, J., Suzumura, T., Pareja, A., Ma, T., Kanezashi, H., Kaler, T., & Leiserson, C. E. (2018). Scalable graph learning for anti-money laundering. *Proceedings of ACM SIGKDD*, 3407–3415. https://doi.org/10.48550/arXiv.1812.00076

Yang, J., & Trubey, R. (2019). Financial transaction monitoring using machine learning: A review. *Journal of Financial Crime*, *26*(3), 811–828.

Yu, Q., Xu, Z., & Ke, Z. (2024). Deep learning for cross-border transaction anomaly detection in AML systems. *ArXiv Preprint*. https://doi.org/10.1109/MLBDBI63974.2024.10823769