International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Unsupervised Anomaly Detection In Financial Transactions Using Machine Learning

Dr A Vikram M.E,Ph D.,
Assistant Professor
Department of Information Technology
Aditya Engineering College (A),
Surampalem,AP, India
drvikram171028@gmail.com

S.Manohar
Student
Department of Information Technology
Aditya engineering College (A),
Surampalem,AP, India
sasubillimanohar1@gmail.com

P.Sri Harshitha
Student
Department of Information Technology
Aditya engineering College (A),
Surampalem,AP, India
Sriharshitha2308@gmail.com

L.Divya
Student
Department of Information Technology
Aditya engineering College (A),
Surampalem,AP, India
divyalysetti2005@gmail.com

T.Uday Kiran
Student
Department of Information Technology
Aditya engineering College (A),
Surampalem,AP, India
udaykiran2606@gmail.com

***Abstract*— Financial transaction systems are increasingly vulnerable to sophisticated fraudulent and anomalous activities, necessitating intelligent and scalable detection mechanisms. Machine learning–based anomaly detection has emerged as a critical solution for identifying high-risk financial behaviors in large-scale transactional environments. This study presents an advanced anomaly detection framework utilizing the *Financial Transaction and Risk Management* dataset from Kaggle. The workflow incorporates comprehensive data preprocessing, feature engineering (temporal extraction, group-based aggregation, log transformation, and interaction features), encoding, ADASYN-based resampling, and Min–Max normalization. Multiple machine learning algorithms, including Random Forest, XGBoost, Extra Trees, K-Nearest Neighbors, Voting Classifier, and Stacking Classifier, are implemented for comparative evaluation. Experimental results demonstrate that the Stacking Classifier achieves superior performance with 95.4% accuracy, 95.7% precision, 95.4% recall, 95.5% F1-score, and 99.2% ROC–AUC, indicating strong discriminative capability in identifying anomalous transactions. Model interpretability is enhanced using LIME and SHAP to provide local and global explanations of prediction outcomes. Furthermore, a deployment interface is developed using Flask with SQLite integration to enable secure user authentication and real-time transaction risk prediction. The proposed framework ensures robust, interpretable, and scalable financial anomaly detection suitable for practical applications.**

Keywords*— *Financial Anomaly Detection, Machine Learning, Stacking Classifier, ADASYN, Explainable Artificial Intelligence, SHAP, LIME, Flask Deployment.

I. INTRODUCTION

The rapid growth of digital banking, online payment systems, and electronic financial services has drastically increased the number and complexity of financial transactions globally. While this transformation provides convenience and accessibility, it also exposes financial systems to fraudulent activities, money laundering, and anomalous transaction patterns. Fraudulent behavior can cause substantial financial losses, damage institutional credibility, and undermine regulatory compliance. Traditional rule-based auditing systems often fail to adapt to the dynamic nature of digital transactions and evolving fraud techniques, making automated detection increasingly necessary [1].

Recent research has focused on leveraging unsupervised learning to identify anomalies in financial networks without relying on labeled data. Techniques such as Antibenford subgraphs analyze transaction behavior to uncover irregular patterns, enabling detection of unusual activities in complex financial systems [2]. In the cryptocurrency domain, unsupervised anomaly detection has been applied to identify suspicious behaviors in decentralized networks, addressing the challenges of high-frequency, pseudonymous transactions [3]. Surveys of financial anomaly detection methods highlight the limitations of supervised models and emphasize the importance of adaptive, data-driven approaches for detecting new or emerging fraud patterns [4].

Advanced detection frameworks integrate anomaly feature identification with machine learning classifiers to enhance prediction accuracy. For example, CoDetect applies feature-based anomaly analysis to detect irregular transactions effectively [5]. Real-time detection approaches have employed graph neural networks to monitor transaction networks, enabling instant identification of fraudulent activities [6]. Unsupervised learning in banking transactions has revealed complex illicit patterns, demonstrating the efficacy of ensemble and hybrid approaches in capturing hidden anomalies [7]. Additionally, novel ensemble methods

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

and convolutional neural network architectures have been proposed to improve detection rates and handle large-scale transactional datasets [8][9][10].

The objective is to develop an intelligent system capable of analyzing financial transaction data to identify suspicious and anomalous activities accurately. The system will incorporate comprehensive data preprocessing, feature engineering, encoding, resampling, and normalization to enhance model performance. Multiple classifiers will be integrated to provide robust predictions, while explainable artificial intelligence techniques such as LIME and SHAP will ensure transparency. Furthermore, a web-based interface will facilitate secure user input and real-time risk assessment, supporting efficient monitoring and decision-making in digital financial environments.

II. Related Work

Financial transaction monitoring has become a critical area of research due to the rise of digital banking, online payments, and real-time money transfers. Traditional auditing methods relying on rules and static thresholds are insufficient to capture evolving fraudulent behaviors and complex transaction patterns. Machine learning approaches have therefore emerged as a significant solution, offering automated and adaptive detection capabilities. Desai, Kosse, and Sharples [11] proposed a machine learning framework for anomaly detection in payment systems, demonstrating how structured transactional features can be analyzed to identify subtle irregularities that might be missed by conventional techniques. This framework highlights the importance of data-driven approaches in uncovering hidden patterns within high-volume transaction streams.

Oliveira, Sant'Anna, and Ferreira [12] introduced complex networks-based anomaly detection to tackle anti-money laundering challenges. By modeling financial transactions as networks, this approach detects unusual connectivity patterns and deviations in transaction flows, providing deeper insights into fraudulent behavior beyond simple transactional attributes. Emmanuel [13] explored deep learning architectures for real-time anomaly detection, emphasizing the need for models that can handle streaming data and adapt to evolving behaviors without significant delays. These architectures utilize temporal and contextual information to capture deviations dynamically, enhancing responsiveness in financial monitoring systems.

Chen, Jiang, Wang, Li, Yu, Shen, and Du [14] proposed generative adversarial synthetic neighbors-based unsupervised anomaly detection, which leverages generative models to simulate normal transaction patterns. Deviations from these synthetic representations are then flagged as anomalies. This approach demonstrates the power of unsupervised learning in settings where labeled fraudulent data is scarce or unavailable, providing a scalable solution for high-dimensional financial datasets. Shanaa and Abdallah [15] presented a hybrid framework combining supervised and unsupervised learning to enhance credit card fraud detection, effectively integrating labeled and unlabeled data to improve detection rates while maintaining adaptability.

Latif and Kaplan [16] focused on high-frequency cryptocurrency markets, highlighting the challenges of anomaly detection in volatile and pseudonymous transaction environments. Their unsupervised learning approach identifies deviations in transaction sequences and trading behavior, offering valuable insights for emerging financial ecosystems. Bosnyaková, Babič, Adam, and Biceková [17] applied unsupervised learning to blockchain networks, detecting irregular patterns and potential attacks by monitoring decentralized ledger activities.

Zakaria, Rahman, Rahman, and Rafi [18] integrated graph neural networks with anomaly detection techniques, enabling real-time monitoring of interconnected transaction networks. Davitaia [19] emphasized the broader application of artificial intelligence and machine learning in digital payment fraud detection, reviewing how advanced computational models improve both speed and accuracy of anomaly identification. Prasad, Anusha, Amulya, and Devi [20] combined Isolation Forest and temporal attention-based LSTM models to detect anomalies in banking transactions, effectively capturing both structural and sequential irregularities in transaction data.

Collectively, these studies highlight the importance of adaptive, scalable, and interpretable machine learning methods for financial anomaly detection. Techniques range from graph-based and network modeling approaches to hybrid supervised-unsupervised frameworks and deep learning architectures, emphasizing the need to handle real-time, high-dimensional, and evolving financial transaction data with transparency and reliability.

III. Materials And Methods

The system introduces a robust framework for financial transaction anomaly detection, designed to identify suspicious activities through the integration of advanced machine learning and explainable techniques. The workflow begins with comprehensive data preprocessing, including cleaning, normalization, encoding, and resampling, to ensure high-quality representation of transactional behaviors. Feature engineering enhances the dataset by generating composite, temporal, and aggregated features, improving the system's ability to capture complex patterns. Multiple classification models, including Random Forest, XGBoost, Extra Trees, K-Nearest Neighbors, Voting Classifier, and Stacking Classifier, are implemented to provide robust predictions across diverse transaction types, ensuring reliable detection [21]. To enhance transparency and interpretability, Explainable Artificial Intelligence techniques such as LIME and SHAP are incorporated, allowing for both local and global explanations of the model's outputs. These mechanisms support understanding of feature importance and foster trust in automated decision-making processes [22][23]. A web-based interface developed using Flask enables secure user authentication, transaction input submission, backend processing, and real-time prediction display, improving accessibility and operational efficiency [24][25]. The integrated framework combines accuracy, interpretability, and deployability, offering a scalable solution for intelligent financial monitoring, risk assessment, and proactive anomaly detection in digital financial environments.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

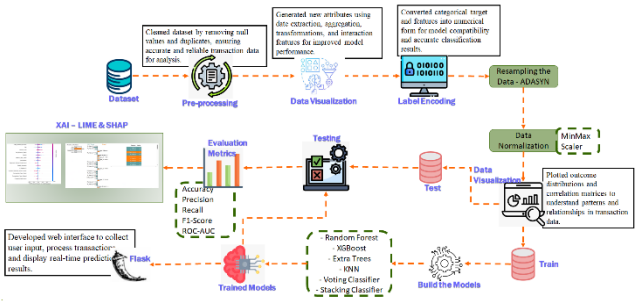

Fig. 1. System Architecture

Fig. 1 illustrates a machine learning pipeline starting with data pre-processing and feature engineering to enhance performance. After label encoding and ADASYN resampling, data is normalized using MinMaxScaler and split for training. Six models, including Stacking and Voting classifiers, are built and tested. Finally, evaluation metrics and XAI tools like SHAP ensure interpretability before Flask deployment.

A) Dataset Collection:

The dataset used for this work is obtained from a publicly available financial transaction and risk management repository, containing comprehensive transactional records suitable for anomaly detection. It includes multiple attributes that capture detailed financial operations, such as Transaction ID, Date, Account Number, Transaction Type, Amount, Currency, Counterparty, Category, Payment Method, Risk Incident, Risk Type, Incident Severity, Error Code, User ID, System Latency, Login Frequency, Failed Attempts, and IP Region. These features provide a rich representation of both normal and potentially suspicious activities, enabling in-depth analysis of transactional behaviors.

Exploring the dataset involves loading the accounting dataset into the analysis environment, examining the structure, data types, and statistical properties of each attribute. Numerical features such as Amount, System Latency, Login Frequency, and Failed Attempts are summarized with mean, median, and standard deviation, while categorical attributes like Transaction Type, Currency, Category, and Risk Type are analyzed for unique values and frequency distributions. Risk-related fields, including Risk Incident, Incident Severity, and Error Code, help highlight critical anomalies and potential threats. Fig.2 illustrates a sample snapshot of the dataset, showing the diverse attributes and their corresponding values, providing a clear view of transactional details and their potential contribution to anomaly detection analysis.

This structured dataset forms a solid foundation for preprocessing, feature engineering, and building machine learning models to identify and predict anomalous financial activities effectively.

	Transaction_ID	Date	Account_Number	Transaction_Type	Amount	Currency	Counterparty	Category	Payment_Method	Risk_Incident	Risk_Type	Incident_Severity	Error_Code	User_ID	System_Late
0	TXN00001	2024-08-18	250637	Refund	952.11	USD	Garcia-Gordon	Payroll	Cash	0	None	None	None	U001	29
1	TXN00002	2024-11-26	122794	Debit	3293.29	USD	Heath, Pena and Buchanan	Payroll	Cash	1	Error	Low	E002	U233	24
2	TXN00003	2024-04-30	152231	Debit	2849.97	USD	Pham PLC	Operations	Cash	0	None	None	None	U251	37
3	TXN00004	2024-03-23	862833	Refund	9813.53	USD	Levine, Long and Stewart	Payroll	Cash	0	None	None	None	U235	24

Fig. 2. Dataset

B) Pre-Processing:

Data preprocessing prepares the financial transaction dataset for analysis by cleaning, transforming, and structuring the data. This step ensures accuracy, removes inconsistencies, and enhances the quality of information, forming a reliable foundation for model training and anomaly detection.

Data Processing: Data processing is a crucial step to ensure that the financial transaction dataset is accurate, consistent, and suitable for modeling. Initially, null values are identified and removed to prevent errors during model training and to avoid skewed predictions caused by missing information. Duplicate records are also eliminated, ensuring each transaction is unique and preventing bias in model evaluation. During this phase, the data is inspected for outliers and inconsistencies, particularly in numerical fields such as Amount, System Latency, and Failed Attempts, which could distort model learning. Categorical attributes like Transaction Type, Currency, Category, and Payment Method are standardized to maintain consistency in naming conventions. Date and time fields are converted into a uniform format to allow temporal analysis. Additionally, irrelevant or redundant columns that do not contribute to predictive modeling are identified for removal. This structured and clean dataset forms the basis for subsequent steps such as feature engineering and encoding, ensuring models receive reliable inputs and can effectively detect anomalies in financial transactions.

Feature Engineering: Feature engineering transforms raw transactional data into meaningful attributes that enhance the model's predictive power. Date and time fields are decomposed to extract temporal features such as day, month, hour, weekday, and transaction intervals, enabling the system to identify patterns related to transaction timing. Group-based aggregation features are computed by summarizing transaction statistics per account or user, including average transaction amounts, frequency of transactions, and total risk incidents, which help capture behavioral trends. Log transformation is applied to skewed numerical features such as Amount or System Latency to normalize distributions and reduce the influence of extreme values. Interaction or composite features are created by combining two or more attributes, such as multiplying transaction amount by login frequency or failed attempts, to reveal complex relationships and hidden anomalies. Unwanted or irrelevant columns that do not contribute to classification, including identifiers that are redundant for modeling, are removed to simplify the dataset and improve computational efficiency. By generating these enriched features, the system can capture intricate patterns in transaction behavior, facilitating more accurate anomaly detection in financial activities.

Label Encoding: Label encoding converts categorical variables into numerical representations suitable for machine learning algorithms. In the financial transaction dataset, the

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

target attribute representing transaction status or anomaly class is transformed from categorical labels into numeric codes, enabling classification models to process the output efficiently. This approach ensures compatibility with algorithms that require numerical inputs, such as Random Forest, XGBoost, and Extra Trees. During encoding, each unique label is assigned a distinct integer value, preserving the relationship between categories while avoiding ambiguity. For categorical features not used as targets, one-hot encoding is applied, transforming each category into a binary column that indicates the presence or absence of a specific value. This prevents models from incorrectly assuming ordinal relationships between non-numeric categorical attributes, such as Transaction Type, Payment Method, or Currency. Proper encoding is essential for ensuring that models interpret feature values correctly, avoid misrepresentation of categorical data, and maintain predictive accuracy. Furthermore, encoding supports subsequent stages such as resampling and normalization, ensuring a fully machine-readable dataset for effective anomaly detection.

Resampling the Data: Resampling addresses the class imbalance commonly found in financial transaction datasets, where normal transactions heavily outnumber anomalous ones. Imbalanced data can cause models to be biased toward majority classes, resulting in poor detection of rare fraudulent activities. To overcome this, the Adaptive Synthetic (ADASYN) sampling technique is employed, which generates synthetic examples of minority class transactions based on their feature space distribution. ADASYN focuses on harder-to-learn instances, increasing the representation of minority transactions while maintaining data diversity and reducing overfitting. By adjusting the class distribution, the models receive a balanced training dataset, improving their ability to distinguish anomalies from normal activities. The process involves calculating the number of synthetic samples needed per minority instance and interpolating new samples along feature-space directions between the instance and its nearest neighbors. After resampling, the dataset is verified to ensure proper class balance and consistency with original feature distributions. This step is critical for enhancing model sensitivity, enabling classifiers like Random Forest, KNN, and Stacking Classifiers to reliably identify rare anomalous transactions without compromising overall accuracy.

Data Normalization: Data normalization scales numerical features to a consistent range, preventing attributes with large magnitudes from disproportionately influencing model performance. In financial transaction datasets, fields such as Amount, System Latency, Login Frequency, and Failed Attempts can have widely varying values. Min-Max scaling is applied to transform these features into a standard range, typically between 0 and 1, ensuring uniform contribution to distance-based or tree-based models. Normalization also accelerates model convergence during training by stabilizing weight updates and reducing gradient disparities. Additionally, normalized features improve interpretability of results when combined with explainable AI techniques like LIME and SHAP, allowing clearer comparison of feature importance. Both the training and testing datasets undergo identical scaling to maintain consistency and prevent data leakage. This step is particularly essential for models sensitive to feature magnitude, such as K-Nearest Neighbors, while also benefiting ensemble models by improving stability and predictive reliability. Overall, normalization ensures that all numerical attributes contribute proportionately to anomaly detection, enhancing model performance and generalization across diverse transaction patterns.

Data Visualization: Data visualization provides insight into the structure, distribution, and relationships of features within the financial transaction dataset. Initially, the distribution of classification outcomes is analyzed before and after resampling, highlighting the effects of balancing techniques like ADASYN and ensuring that minority class transactions are adequately represented. Histograms, bar plots, and density plots are used to depict numerical and categorical feature distributions, such as transaction amounts, risk incident frequency, and payment methods, enabling the identification of potential patterns or irregularities. A correlation matrix is generated to examine relationships between features, revealing dependencies that may influence model predictions. High correlations between attributes like Failed Attempts, Login Frequency, and System Latency may indicate behavioral patterns associated with anomalous transactions. Visual inspection of these relationships helps guide feature selection and engineering, ensuring that models capture meaningful interactions. Additionally, scatter plots and boxplots can highlight outliers or extreme values, providing context for preprocessing decisions like log transformations. These visualization techniques enhance understanding of transactional behaviors, support informed preprocessing and feature engineering decisions, and provide interpretable insights for anomaly detection models.

Train & Test Split: Splitting the dataset into training and testing sets is a critical step to evaluate model performance objectively. After preprocessing, feature engineering, encoding, resampling, and normalization, the dataset is divided, typically using a stratified sampling approach to maintain proportional representation of normal and anomalous transactions. The training set is used to train multiple classifiers, including Random Forest, XGBoost, Extra Trees, K-Nearest Neighbors, Voting Classifier, and Stacking Classifier, allowing models to learn patterns distinguishing normal and suspicious behaviors. The test set, which remains unseen during training, evaluates model generalization, ensuring that predictions are accurate on new data. This separation prevents overfitting and provides a reliable estimate of real-world performance. During splitting, care is taken to maintain temporal and behavioral patterns to avoid leakage, particularly for sequential or time-dependent features. The train-test strategy also facilitates hyperparameter tuning and cross-validation, improving model robustness. Ultimately, a well-defined train-test split ensures that anomaly detection models perform reliably and can effectively identify suspicious financial transactions in operational environments.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

C) Algorithms:

Random Forest: Random Forest is an ensemble technique that builds multiple decision trees on randomly selected data subsets and features. Each tree predicts independently, and majority voting determines the final output. It reduces overfitting, handles high-dimensional and nonlinear features, and manages noise effectively. In financial transaction monitoring, Random Forest identifies anomalies by analyzing patterns in amounts, timing, and user behavior. Its ability to rank feature importance enhances interpretability, allowing detection of influential risk factors and providing stable, reliable classification outcomes for suspicious transactions.

XGBoost: XGBoost is a gradient boosting algorithm that builds decision trees sequentially, correcting errors from previous trees. It uses regularization and gradient-based optimization to prevent overfitting and improve performance. In financial anomaly detection, XGBoost captures hidden patterns and subtle deviations in transaction attributes, handling missing values and nonlinear relationships efficiently. Its parallel processing and tree pruning ensure computational efficiency and predictive stability. The algorithm effectively identifies high-risk transactions while maintaining consistency across varied financial behaviors and transaction patterns.

Extra Trees: Extra Trees, or Extremely Randomized Trees, constructs multiple decision trees with random feature selection and split thresholds, increasing randomness compared to traditional ensembles. This reduces variance, prevents overfitting, and improves generalization. In financial monitoring, it efficiently models nonlinear patterns among temporal and behavioral features. Aggregating predictions from all trees ensures consistent classification performance. The algorithm performs well on large, high-dimensional datasets, making it suitable for detecting unusual financial transactions and complex anomaly patterns while maintaining computational efficiency and reliability.

K-Nearest Neighbors: K-Nearest Neighbors classifies a data point based on the majority label of its nearest neighbors using distance metrics like Euclidean distance. No explicit training is required, as predictions rely on stored instances. In financial anomaly detection, it identifies transactions that deviate from typical behavioral patterns. Proper normalization and feature selection are crucial for high-dimensional data. KNN effectively detects unusual activity by comparing feature similarity, providing an interpretable and flexible approach for recognizing anomalous transactions in transactional datasets.

Voting Classifier: Voting Classifier combines multiple model predictions through majority or weighted voting, enhancing stability and predictive strength. It integrates outputs from diverse classifiers, reducing individual model bias. In financial anomaly detection, it improves robustness by capturing varied decision boundaries and complex transaction patterns. The ensemble approach ensures more reliable identification of anomalous activities and mitigates weaknesses of individual models, providing consistent and accurate predictions for high-risk financial behavior monitoring.

Stacking Classifier: Stacking Classifier combines multiple base models and uses a meta-learner to produce final predictions. Base model outputs serve as input features for the higher-level learner, optimizing prediction combinations. In financial anomaly detection, stacking captures complex feature relationships and improves generalization across unseen transactions. It leverages multiple model perspectives, enhancing classification accuracy and detection performance. The layered approach provides strong consistency and superior ability to identify suspicious transactions in diverse financial environments.

D) Integration of XAI & Flask:

Explainable Artificial Intelligence (XAI) integrates methods like LIME and SHAP to provide interpretable insights into model predictions. XAI highlights feature contributions, supports both local and global explanations, and ensures transparency, enabling users to understand why a transaction is flagged as anomalous.

The Flask framework delivers a web-based interface for secure interaction, handling user authentication, transaction input, and backend processing. It enables real-time prediction display, seamless communication with the trained model, and a user-friendly environment, making anomaly detection accessible and interactive for monitoring financial transactions effectively.

IV. Experimental Results

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} (1)$$

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} (2)$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} (3)$$

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$F1\ Score = 2 * \frac{Recall\ \mathrm{X}\ Precision}{Recall + Precision} * 100 (1)$$

AUC-ROC Curve: The AUC-ROC Curve is a performance measurement for classification problems at various threshold settings. ROC plots the True Positive Rate against the False Positive Rate. AUC quantifies the overall ability of the model to distinguish between classes, where a higher AUC indicates better model performance.

$$AUC = \sum_{i=1}^{n-1} (FPR_{i+1} - FPR_i) \cdot \frac{TPR_{i+1} + TPR_i}{2} \quad (5)$$

Table.1 Performance Evaluation Table

ML Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
RandomForest	0.941	0.945	0.941	0.941	0.986
XGBoost	0.891	0.893	0.891	0.891	0.970
Extra Trees	0.897	0.899	0.897	0.896	0.963
KNN	0.721	0.748	0.721	0.695	0.943
Voting Classifier	0.950	0.954	0.949	0.950	0.988
Stacking Classifier	0.954	0.957	0.954	0.955	0.992

Table.1 presents the performance evaluation of all models, highlighting accuracy, precision, recall, F1-score, and ROC-AUC for comparison.

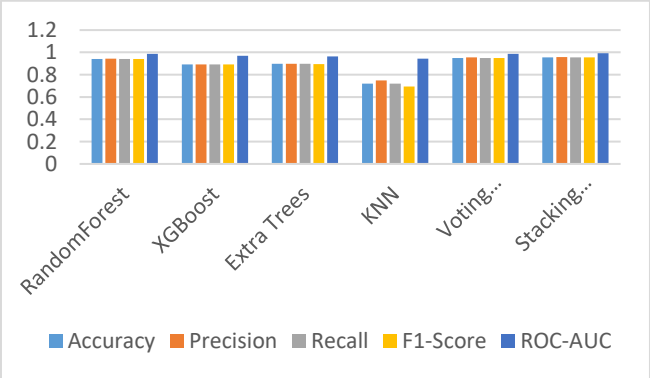

Fig. 3. Comparison Graph

Fig. 3 illustrates performance metrics across six machine learning models, showing that Stacking and RandomForest achieve the highest overall predictive accuracy.

V. Conclusion

Accurate and reliable detection of anomalous financial transactions is essential for minimizing economic losses and strengthening risk management strategies in digital financial ecosystems. The developed machine learning framework successfully integrates advanced preprocessing, feature engineering, resampling using ADASYN, and ensemble-based modeling to enhance predictive performance on imbalanced transaction data. Among the evaluated models, the Stacking Classifier demonstrated superior performance, achieving 95.4% accuracy, 95.7% precision, 95.4% recall, 95.5% F1-score, and 99.2% ROC–AUC, indicating strong generalization capability and high discriminative power for identifying risky transactions. Model transparency was ensured through Explainable Artificial Intelligence techniques, where LIME provided instance-level interpretability and SHAP delivered global feature importance insights, enabling a deeper understanding of the factors influencing anomaly predictions. These explanations improve trust, regulatory compliance, and practical usability in financial environments. Additionally, deployment through a Flask-based web interface with SQLite-backed authentication enables secure user interaction and real-time prediction, ensuring operational feasibility. The integrated architecture demonstrates robustness, interpretability, and scalability, making it suitable for real-world financial transaction monitoring and intelligent risk assessment systems.

Future enhancements can focus on incorporating deep learning architectures such as LSTM and Transformer-based models to capture complex sequential and temporal transaction patterns more effectively. Integration of real-time streaming frameworks (e.g., Apache Kafka and Spark Streaming) can enable low-latency anomaly detection in high-frequency financial environments. Advanced unsupervised and semi-supervised approaches, including Autoencoders and Isolation Forest with adaptive thresholding, may further improve detection of previously unseen fraud patterns. Expanding explainability with counterfactual explanations and model monitoring dashboards would strengthen transparency and regulatory compliance. Additionally, deploying the system in a cloud-native microservices architecture with RESTful APIs and containerization (Docker, Kubernetes) can enhance scalability, fault tolerance, and enterprise-level integration across distributed financial infrastructures.

References

[1] Chen, T., & Tsourakakis, C. (2022, August). Antibenford subgraphs: Unsupervised anomaly detection in financial networks. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 2762-2770).

[2] Pham, T., & Lee, S. (2016). Anomaly detection in bitcoin network using unsupervised learning methods. arXiv preprint arXiv:1611.03941.

[3] Ahmed, M., Mahmood, A. N., & Islam, M. R. (2016). A survey of anomaly detection techniques in financial domain. Future Generation Computer Systems, 55, 278-288.

[4] Huang, D., Mu, D., Yang, L., & Cai, X. (2018). CoDetect: Financial fraud detection with anomaly feature detection. IEEE Access, 6, 19161-19174.

[5] Rasul, I., Shaboj, S. I., Rafi, M. A., Miah, M. K., Islam, M. R., & Ahmed, A. (2024). Detecting financial fraud in real-time transactions

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

using graph neural networks and anomaly detection. Journal of Economics, Finance and Accounting Studies, 6(1), 131-142.

[6] Karnavou, E., Cascavilla, G., Marcelino, G., & Geradts, Z. (2025). I know you're a fraud: Uncovering illicit activity in a Greek bank transactions with unsupervised learning. Expert Systems with Applications, 288, 128148.

[7] Rajaprkash, S., Kumar, A., Reddy, G. S. K., Reddy, A. S., & Lokesh, B. (2025, June). Supervised and Unsupervised Learning for Fraud Detection in Banking Transactions. In 2025 International Conference on Emerging Technologies in Engineering Applications (ICETEA) (pp. 1-4). IEEE.

[8] Sizan, M. M. H. (2025). Machine learning-based unsupervised ensemble approach for detecting new money laundering typologies in transaction graphs. International Journal of Applied Mathematics, 38(2s), 351-374.

[9] Fariha, N., Khan, M. N. M., Hossain, M. I., Reza, S. A., Bortty, J. C., Sultana, K. S., ... & Begum, M. (2025). Advanced fraud detection using machine learning models: Enhancing financial transaction security. arXiv preprint arXiv:2506.10842.

[10] Mazumder, M. T. R., Shourov, M. S. H., Rasul, I., Akter, S., & Miah, M. K. (2025). Anomaly detection in financial transactions using convolutional neural networks. Journal of Economics, Finance and Accounting Studies, 7(2), 195-207.

[11] Desai, A., Kosse, A., & Sharples, J. (2025). Finding a needle in a haystack: a machine learning framework for anomaly detection in payment systems. The Journal of Finance and Data Science, 11, 100163.

[12] Oliveira, R. M. A., Sant'Anna, A. M. O., & Ferreira, P. H. (2025). Complex networks-based anomaly detection for financial transactions in anti-money laundering. Forensic Science International: Digital Investigation, 55, 302005.

[13] Emmanuel, M. (2025). Deep Learning Architectures for Real-Time Anomaly Detection in Financial Transactions. ResearchGate Publication, June.

[14] Chen, L., Jiang, H., Wang, L., Li, J., Yu, M., Shen, Y., & Du, X. (2025). Generative adversarial synthetic neighbors-based unsupervised anomaly detection. Scientific Reports, 15(1), 16.

[15] Shanaa, M., & Abdallah, S. (2025). A hybrid anomaly detection framework combining supervised and unsupervised learning for credit card fraud detection. F1000Research, 14, 664.

[16] Latif, M. N., & Kaplan, M. (2025). Unsupervised machine learning based anomaly detection in high frequency data: Evidence from Cryptocurrency Market. Pakistan Journal of Commerce and Social Sciences (PJCSS), 19(3), 407-440.

[17] Bosnyaková, B., Babič, F., Adam, T., & Biceková, A. (2025, January). Anomaly detection in blockchain network using unsupervised learning. In 2025 IEEE 23rd World Symposium on Applied Machine Intelligence and Informatics (SAMI) (pp. 000221-000224). IEEE.

[18] Zakaria, R. M., Rahman, M. M., Rahman, H., & Rafi, M. A. (2025). Detecting Financial Fraud in Real-Time Transactions Using Graph Neural Networks and Anomaly Detection Techniques. Journal of Economics, Finance and Accounting Studies, 7(6), 01-13.

[19] Davitaia, A. (2025). Artificial Intelligence and machine learning in fraud detection for digital payments. International Journal of Science and Research Archive, 15(3), 714-719.

[20] Prasad, V. S. R., Anusha, N., Amulya, K., & Devi, K. D. (2025). Anomaly Detection in Banking Transactions Using Isolation Forest and Temporal Attention-Based LSTM Deep Learning Model. Synthesis: A Multidisciplinary Research Journal, 3(1s), 14-26.

[21] Vysotska, V., Uhryn, D., Iliuk, O., Ushenko, Y., & Yatsyshyn, V. (2025). Application of machine learning for predicting fraudulent anomalies in financial transactions. In CEUR Workshop Proceedings (Vol. 4126, pp. 171-185).

[22] Paoletti, G., Giobergia, F., Giordano, D., Cagliero, L., Ronchiadin, S., Moncalvo, D., ... & Baralis, E. (2025, August). MAD: Multicriteria Anomaly Detection of Suspicious Financial Accounts from Billions of Cash Transactions. In Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2 (pp. 4751-4760).

[23] Alumona, P., Lawal, O., Ikhifa, M. O., Agbeso, D. O., Awele, O., & Olukoya, D. (2025). Fraud Detection in Financial Transactions Using Machine Learning: Insights from the PaySim Mobile Money Dataset.

[24] Herreros-Martínez, A., Magdalena-Benedicto, R., Vila-Francés, J., Serrano-López, A. J., Pérez-Díaz, S., & Martínez-Herráiz, J. J. (2025). Applied machine learning to anomaly detection in enterprise purchase processes: A hybrid approach using clustering and isolation forest. Information, 16(3), 177.

[25] Sasikala, T. S., Sivakami, S., G, P. N., & Saranya, R. (2025). TransGuard-QNet: A Hybrid Quantum Machine Learning Framework for Financial Transaction Risk Assessment and Fraud Detection With Linear Programming Optimization. Computational Economics, 1-27.