Credit Card Fraud Detection

sakshi kushwaha
Galgotias University gr.noida,
singhsakshii1109@gmail.com

Tanishka Bipin Singh
Galgotias University gr.noida,India
singhtanishka591@gmail.com

Akhilesh kuamr singh
Galgotias university, grnoida, India
akhileshsingh@galgotiasuniversity.edu.in

Abstract

With the rapid expansion of digital financial services, banks and financial institutions are increasingly vulnerable to fraudulent activities that compromise both user trust and financial security. In this research, we propose a robust machine learning framework for the detection of fraudulent transactions using a stacked ensemble model. A dataset simulating real-world bank transaction behavior, comprising 200,000 records, is used. Each transaction record includes a diverse set of features such as demographic details, transaction metadata, and account attributes. Preprocessing steps include removal of high-cardinality identifiers, label encoding of categorical variables, standardization of numerical features, and class balancing using Synthetic Minority Over-sampling Technique (SMOTE). Five classifiers—Logistic Regression, Decision Tree, Random Forest, Naive Bayes, and XGBoost—are individually trained and then combined using a StackingClassifier with Logistic Regression as a meta-learner. The models are evaluated using various classification metrics such as precision, recall, F1-score, and accuracy. The ensemble model outperforms all base classifiers, especially in recall and F1 score, making it suitable for real-time banking fraud detection systems. This study underscores the potential of ensemble learning in financial security and provides a deployable model for integration into existing banking infrastructures.

Keywords: Financial fraud detection, Machine learning, Ensemble learning, SMOTE, Classification models, StackingClassifier, Banking security, Transaction analysis

I. INTRODUCTION

In recent years, the digitalization of financial services has significantly reshaped the banking industry, transforming traditional transactional frameworks into highly dynamic, real-time ecosystems. With the widespread adoption of online and mobile banking, users now perform transactions with unprecedented speed and convenience. As of 2023, over 76% of adults globally use some form of digital financial service, highlighting the scale and critical importance of these systems (World Bank, 2023). This explosive growth has been driven by factors such as increased internet penetration, the proliferation of smartphones, and enhanced fintech innovations. However, this digital transformation has not come without its challenges. One of the most pressing concerns confronting financial institutions is the increasing prevalence of fraudulent transactions, which exploit both technological vulnerabilities and human behavior to gain unauthorized access to funds.

Fraudulent transactions encompass a wide range of malicious activities, including phishing attacks, identity theft, account takeovers, unauthorized transactions, and synthetic identity fraud. These actions are often facilitated by complex fraud rings using automation, machine learning, and social engineering tactics to evade traditional defenses. According to a report by the Association of Certified Fraud Examiners (2022), global financial fraud accounts for losses exceeding $5 trillion annually. Beyond financial losses, fraud incidents severely erode customer trust, complicate compliance with regulatory standards such as GDPR and PCI-DSS, and tarnish institutional reputations. Traditional fraud detection systems, which rely on rule-based heuristics and threshold alerts, are increasingly inadequate in combating such sophisticated schemes due to their static nature and inability to adapt to evolving threats in real-time (Bolton and Hand, 2002).

To address this, financial institutions are turning to intelligent systems capable of dynamically identifying and responding to emerging fraud patterns. Machine learning (ML) has emerged as a compelling solution due to its ability to learn from historical data, detect complex patterns, and flag deviations without manual intervention. However, not all ML approaches are created equal. Simple classifiers such as Decision Trees and Logistic Regression often lack the nuance required for complex, imbalanced datasets. Advanced methods like ensemble learning—particularly stacked generalization—offer a compelling alternative by combining the strengths of multiple base classifiers into a more accurate and resilient meta-model (Bhattacharyya et al., 2011).

In this study, we present a comprehensive fraud detection system based on a stacking ensemble methodology. Our model integrates five distinct ML algorithms—Logistic Regression, Decision Tree, Random Forest, Naive Bayes, and XGBoost—within a unified predictive framework. The system was trained and validated on a realistic, anonymized dataset containing 200,000 transaction records, featuring variables such as demographic attributes, transaction metadata, and behavioral patterns. Additionally, the system incorporates key preprocessing techniques such as categorical encoding, numerical scaling, and class balancing through the Synthetic Minority Over-sampling Technique (SMOTE), which has been shown to significantly enhance the performance of classifiers on imbalanced datasets (Chawla et al., 2002).

The overarching objectives of our research are fourfold: (1) to explore and document the practical challenges associated with fraud detection in real-world transactional datasets, (2) to compare the performance of multiple ML classifiers and identify their individual strengths and weaknesses, (3) to evaluate the effectiveness of stacking ensembles in improving detection rates, especially for minority fraud classes, and (4) to propose a scalable, interpretable, and deployable framework that can be integrated into live fraud detection pipelines used by financial institutions.

II. RELATED WORKS

The problem of fraud detection in the financial sector has been an enduring challenge for researchers and practitioners alike. Over the past two decades, there has been a substantial evolution from deterministic, rule-based systems to probabilistic and machine learning-based approaches. Early fraud detection techniques primarily relied on expert-defined rules and anomaly thresholds

based on domain experience. These methods were easy to interpret but struggled with adaptability and required frequent manual updates. Bolton and Hand (2002) were among the first to categorize fraud detection into supervised and unsupervised learning frameworks, highlighting that while supervised learning offers better accuracy with labeled data, unsupervised methods are vital when labeled data is scarce.

Logistic Regression and Decision Trees were some of the earliest machine learning techniques deployed in fraud detection systems. Their simplicity and interpretability made them attractive to risk teams and compliance officers. However, these models often underperform in environments with high-dimensional features or non-linear interactions between variables. As the dimensionality and complexity of datasets increased, ensemble methods such as Random Forests and Gradient Boosting Machines (GBMs) gained prominence. These models aggregate the predictions of multiple decision trees to reduce variance and bias, often resulting in superior classification performance. Bhattacharyya et al. (2011) demonstrated the efficacy of Random Forests in a comparative study on credit card fraud, reporting higher detection rates and better recall than traditional classifiers.

The past decade has also seen the emergence of deep learning techniques, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, particularly for sequential transaction data. Shen et al. (2020) applied a deep neural network for detecting accounting fraud, yielding promising results in terms of precision and recall. However, these models are often criticized for their “black-box” nature, which makes them difficult to audit and unsuitable for high-compliance environments. Additionally, they require significant computational resources and large volumes of labeled data to be effective.

Unsupervised techniques such as One-Class SVM and Isolation Forests have also been explored, particularly in scenarios where fraudulent behavior is rare and not well-represented in the training data. While these models excel in novelty detection, they are prone to high false positive rates and often lack stability across different datasets. More recently, hybrid approaches combining supervised, unsupervised, and semi-supervised learning techniques have gained traction. Among these, stacking ensembles have emerged as a particularly effective method. Unlike boosting and bagging, stacking allows the integration of heterogeneous base models, each contributing uniquely to the final decision (Chawla et al., 2002; Bhattacharyya et al., 2011).

Our study contributes to this body of work by implementing a stacking ensemble that leverages the diverse strengths of five base classifiers. Unlike prior studies that focused on optimizing a single algorithm or exploring only deep learning, our approach emphasizes interpretability, computational efficiency, and real-world applicability. The inclusion of SMOTE for class balancing further distinguishes our work, addressing one of the most challenging aspects of fraud detection: the extreme imbalance between legitimate and fraudulent transactions. By rigorously evaluating our model across various performance metrics and comparing it to established baselines, we aim to offer a well-rounded and operationally feasible fraud detection framework for the banking industry (Shen et al., 2020; World Bank, 2023).
The use of modular ensemble design enables flexible updates and integration with existing infrastructures. This approach not only ensures scalability but also allows financial institutions to evolve their fraud detection systems alongside emerging threats and regulatory demands.

III. PROPOSED APPROACH TO DETECT CREDIT CARD FRAUD DETECT

To develop an effective and scalable solution for detecting fraudulent banking transactions, we propose a machine learning pipeline that leverages a stacking ensemble classifier augmented with robust preprocessing techniques and class rebalancing strategies. This section outlines the design and rationale behind our approach, detailing the key steps involved from data preparation to model evaluation. The core philosophy is to build a system that not only achieves high classification accuracy but also maintains interpretability, scalability, and real-world applicability in the context of banking operations.

3.1 Data Preprocessing and Feature Engineering

The dataset used in this study comprises 200,000 anonymized bank transaction records, each labeled as legitimate or fraudulent. Initial exploratory data analysis revealed the presence of high-cardinality categorical variables and redundant identifiers such as Customer_ID, Transaction_ID, and Customer_Email. These were removed to avoid model overfitting and leakage. We retained features that carried transactional or behavioral significance, including Transaction_Amount, Transaction_Type, Device_Type, Account_Balance, and temporal features such as Transaction_Date and Transaction_Time.

Categorical variables were encoded using Label Encoding, a technique suitable for tree-based models that can naturally handle integer labels. Numerical features were standardized using StandardScaler to ensure that models sensitive to feature scaling, such as Logistic Regression and Naive Bayes, could perform optimally. Additionally, we engineered new features by combining temporal fields (e.g., day of the week, hour of the transaction) and calculating ratios such as transaction amount to account balance.

3.2 Handling Class Imbalance with SMOTE

Fraud detection datasets are typically characterized by severe class imbalance. In our dataset, only 1.02% of transactions were labeled as fraudulent. To address this, we employed the Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic samples for the minority class by interpolating between existing instances. This method has been shown to improve the performance of classifiers on imbalanced datasets without replicating existing data (Chawla et al., 2002).

3.3 Model Architecture

The proposed architecture comprises five base classifiers—Logistic Regression, Decision Tree, Random Forest, Naive Bayes, and XGBoost—chosen for their complementary strengths. Logistic Regression provides a strong linear baseline, while Decision Trees and Random Forests are adept at capturing non-linear relationships and feature interactions. Naive Bayes brings in probabilistic reasoning, which is valuable in domains with categorical dominance. XGBoost offers high performance and efficient computation, often outperforming traditional classifiers on structured data.

These base learners were combined using a StackingClassifier, with Logistic Regression as the final meta-learner. Stacking enables the model to leverage the diverse decision boundaries of the base models and correct their individual weaknesses. The meta-model is trained on the out-of-fold predictions from the

base classifiers, ensuring that it learns from their performance without overfitting to the training data.

3.4 Training and Evaluation
We partitioned the dataset into an 80-20 train-test split. The training set was resampled using SMOTE, and the resulting balanced dataset was used to train all models. Each model was trained with default hyperparameters for benchmarking purposes, though in a production scenario, hyperparameter tuning via grid search or Bayesian optimization would be appropriate.

Model performance was evaluated using standard classification metrics: accuracy, precision, recall, F1-score, and ROC-AUC. Particular emphasis was placed on recall and F1-score, as missing fraudulent transactions (false negatives) can be more detrimental than occasional false alarms (false positives). All metrics were computed on the original, imbalanced test set to simulate real-world deployment conditions (Bhattacharyya et al., 2011).

3.5 Implementation and Deployment Considerations
The full pipeline—including preprocessing transformers, class rebalancing methods, and the trained stacking classifier—was serialized using joblib for future reuse. A separate inference script was developed to load the model and process new transaction data in real-time or batch mode. This script includes user-friendly prompts for manual testing, as well as support for CSV-based bulk prediction. Such modularity ensures that the system can be easily integrated with banking APIs, dashboards, or alert systems.

The proposed approach blends interpretability, computational efficiency, and model diversity into a coherent framework that addresses the complexities of fraud detection in modern banking. By adopting a stacking ensemble and enhancing it with robust preprocessing, we provide a foundation for high-performing, real-time fraud prevention systems that financial institutions can trust and deploy at scale (Shen et al., 2020; World Bank, 2023).

IV. ALGORITHMS

The effectiveness of any fraud detection system hinges significantly on the selection of appropriate machine learning algorithms. In this study, we employed five well-established classifiers—each offering distinct advantages—combined within a stacking ensemble to enhance prediction performance and robustness. This section provides a detailed overview of each algorithm, including its theoretical foundation, operational mechanics, and specific role in the ensemble framework.

4.1 Logistic Regression
Logistic Regression is a statistical method that models the probability of a binary outcome based on one or more predictor variables. It is particularly valued for its simplicity, interpretability, and efficiency in linear decision boundaries. The model estimates the probability of a transaction being fraudulent by applying a logistic function to a linear combination of input features. Despite being a linear model, Logistic Regression is highly effective as a meta-learner in stacking frameworks because it can learn from the probabilistic outputs of more complex base models. In our system, Logistic Regression was used both as an individual classifier and as the meta-learner to aggregate the outputs of all base models.

4.2 Decision Tree Classifier
Decision Trees are hierarchical, rule-based models that split input data into branches based on feature thresholds. Each node represents a decision point, leading to a classification at the leaves. The primary advantage of Decision Trees lies in their interpretability and ability to handle both numerical and categorical data without requiring feature scaling. They are prone to overfitting, especially on small datasets, but serve as a solid baseline for capturing non-linear feature interactions. In the proposed system, Decision Trees contributed to the diversity of the ensemble by offering easily understood decision rules and high variance needed in stacking.

4.3 Random Forest Classifier
Random Forest is an ensemble of Decision Trees trained on random subsets of the dataset and feature set. This bootstrapping and feature randomness mitigate overfitting and improve generalization. Random Forest is known for its robustness to noise, scalability to large datasets, and strong performance across a wide variety of domains. In fraud detection, it can model complex patterns and interactions without heavy tuning. As part of our ensemble, it acted as a high-performing, general-purpose model that complemented the simpler learners in terms of both accuracy and feature importance insights.

4.4 Naive Bayes Classifier
Naive Bayes is a probabilistic model based on Bayes' Theorem with the “naive” assumption of conditional independence between features. It is particularly suited to high-dimensional categorical datasets and is computationally efficient even with large feature sets. Despite its simplicity, Naive Bayes often performs surprisingly well in binary classification tasks and is less sensitive to irrelevant features. In this project, it provided a probabilistic contrast to the decision-tree-based learners, contributing unique insights to the stacking ensemble by assigning likelihood estimates to the minority fraud class (Bolton and Hand, 2002).

4.5 XGBoost (Extreme Gradient Boosting)
XGBoost is an advanced boosting algorithm that builds sequential trees to minimize error through gradient descent. It incorporates regularization techniques to prevent overfitting and is optimized for speed and performance. XGBoost excels at capturing intricate non-linear relationships, handling missing data, and ranking feature importance. Its contribution to the ensemble is crucial, as it often outperforms individual classifiers on structured datasets such as financial transactions. In our implementation, it consistently yielded high precision and recall scores, particularly in detecting rare fraudulent patterns (Bhattacharyya et al., 2011; Shen et al., 2020).

By combining these diverse algorithms into a stacking ensemble, we benefit from the linear interpretability of Logistic Regression, the hierarchical logic of Decision Trees, the robustness of Random Forest, the probabilistic grounding of Naive Bayes, and the high accuracy of XGBoost. This collective intelligence leads to a system that is both versatile and reliable, capable of adapting to the multifaceted nature of banking fraud.

V. EXPERIMENTAL RESULT

To assess the performance of the proposed model and compare it with individual classifiers, we conducted a thorough experimental evaluation using standard performance metrics. The dataset was split into 80% training and 20% testing sets. All base classifiers and the final stacked ensemble were evaluated on the test set without class rebalancing to simulate real-world fraud detection scenarios.

5.1 Evaluation Metrics
The following metrics were used to assess model performance:

Accuracy: Overall correctness of the model.

Precision: Fraction of predicted frauds that were correct.

Recall: Fraction of actual frauds that were detected.

F1-Score: Harmonic mean of precision and recall.

ROC-AUC: Area under the Receiver Operating Characteristic curve.

Model	Accuracy	Precision	Recall	F1Score
Logistic Regression	0.81	0.31	0.66	0.42
Decision Tree	0.88	0.41	0.62	0.49
Random Forest	0.94	0.48	0.45	0.47
Naïve Bayes	0.82	0.28	0.70	0.40
XGBoost	0.95	0.51	0.45	0.47
Stacking model	0.96	0.58	0.69	0.63

The results demonstrate that the stacking ensemble outperforms all individual models, particularly in terms of recall and F1-score. This is crucial for fraud detection, where failing to detect fraud (false negatives) can be significantly more damaging than false alarms (false positives). The ensemble benefits from the diversity of its base classifiers, combining their strengths to yield a more balanced and accurate output. Moreover, the ROC-AUC score of 0.81 indicates strong discriminatory power, confirming the model's capability to distinguish between legitimate and fraudulent transactions.

These findings affirm the validity of our proposed approach and suggest that ensemble learning, particularly stacking, is highly effective in addressing the challenges of fraud detection in banking.

VI. CONCLUSION

This study presented a comprehensive machine learning approach for detecting fraudulent transactions in the banking domain. By employing a stacking ensemble model composed of Logistic Regression, Decision Tree, Random Forest, Naive Bayes, and XGBoost classifiers, we achieved significant improvements in detection accuracy, recall, and F1-score compared to individual classifiers. The system effectively addresses the challenges posed by imbalanced datasets through the use of SMOTE and offers both interpretability and robustness for real-world deployment.

The results from our experimental evaluation underscore the effectiveness of ensemble learning in handling complex and nonlinear fraud patterns. Particularly, the stacking model demonstrated superior performance, achieving a 0.96 accuracy and a recall of 0.69, highlighting its utility in minimizing undetected fraudulent transactions. Furthermore, the system is adaptable and scalable, making it a practical candidate for integration into real-time fraud monitoring systems used by financial institutions.

Despite these promising results, there are areas for future exploration. First, further optimization of hyperparameters using advanced techniques such as Bayesian optimization or genetic algorithms could yield even better performance. Second, incorporating deep learning architectures such as LSTM for sequential pattern recognition may help capture temporal dependencies in transaction behavior. Lastly, future work could focus on interpretability enhancements, including SHAP or LIME-based explanations, to provide transparency and trust in decision-making processes, especially in high-compliance environments.

This research validates the use of ensemble machine learning techniques for financial fraud detection and lays a strong foundation for the development of intelligent, real-time fraud prevention systems capable of evolving with the dynamic nature of financial crime (Shen et al., 2020; World Bank, 2023).

Nevertheless, it should be noted that the performance of such models in reality relies on numerous factors, such as hyperparameter optimization, feature engineering, and the quality of input data. More analysis and experimentation are needed to identify the best model for fraud detection in this scenario. Future research could include investigating deep learning methods, using real-time data streams, and applying cost-sensitive learning to further improve fraud detection.

VII. CHALLENGES AND LIMITATIONS

While the proposed stacking-based fraud detection system has demonstrated robust performance in controlled experiments, its deployment in real-world environments entails several challenges and limitations that merit discussion.

Model Interpretability in Regulatory Environments: Financial institutions are often required to justify automated decisions to regulators and customers. While ensemble models such as stacking improve performance, they often sacrifice interpretability. Explaining why a transaction was flagged as fraudulent by a composite of multiple algorithms can be challenging. This lack of transparency may limit adoption in high-compliance domains without supplementary explainability tools like SHAP or LIME.

Data Privacy and GDPR Compliance: Fraud detection systems operate on sensitive personal and financial data. Ensuring compliance with data protection regulations such as the General Data Protection Regulation (GDPR) is paramount. This includes managing user consent, data anonymization, and secure storage practices. Moreover, data-sharing limitations across institutions restrict the ability to build more comprehensive fraud detection models.

Class Imbalance and Data Labeling: Fraudulent transactions typically constitute a small fraction of banking data, leading to significant class imbalance. Although SMOTE partially mitigates this issue during training, it does not address the broader challenge of obtaining accurate, real-time labels in production environments. Labeling errors or delays can compromise model retraining and ongoing performance.

Model Drift and Evolving Fraud Patterns: Fraudulent behavior is dynamic and evolves in response to detection mechanisms. A model trained on historical data may become less effective as fraud strategies change. Continuous monitoring, periodic retraining, and the inclusion of online learning frameworks are necessary to maintain model relevance and accuracy over time.

Recognizing these challenges is essential for responsibly transitioning from experimental results to reliable, real-world applications. Addressing them will involve cross-disciplinary collaboration among data scientists, compliance officers, legal teams, and domain experts.

VIII. DEPLOYMENT CASES AND USE STRATEGY

Deploying a machine learning-based fraud detection model within a financial institution involves multiple operational, architectural, and governance considerations. The proposed stacking ensemble is designed to be modular, scalable, and adaptable for integration into both batch-processing pipelines and real-time transaction monitoring systems.

Modular Pipeline Architecture: The model components—including data preprocessing, feature transformation, and classification—were serialized using joblib, allowing for seamless reuse across different stages of deployment. This modular design facilitates easy integration with microservices, APIs, or legacy systems commonly found in banking IT infrastructures.

Real-time Fraud Detection: The model can be deployed as part of an online transaction processing (OLTP) system. As each transaction is initiated, it is passed through the preprocessing pipeline and subsequently classified. If the transaction is flagged as suspicious, it triggers immediate actions such as two-factor authentication, transaction delay, or alert generation to a fraud analyst. This real-time capability is critical for high-frequency, low-latency environments such as mobile banking and point-of-sale systems.

Batch-mode Analytics and Retrospective Auditing: In addition to live monitoring, the model supports retrospective auditing of historical transaction logs. This is particularly useful for regulatory reporting, internal risk assessments, and uncovering delayed fraud patterns that bypassed initial scrutiny. Banks can schedule batch evaluations during off-peak hours to optimize computing resources.

Integration with Alert Management Systems: Fraud alerts generated by the model can be routed to existing fraud investigation platforms through RESTful APIs or message queues. Each alert can include a risk score, predicted label, and relevant features to aid analysts in making informed decisions.

Use Cases:

Credit Card Fraud Monitoring: Identifying suspicious spending patterns in real-time based on deviation from historical behavior.

Loan Application Fraud: Detecting fake or manipulated identity information during digital loan submissions.

Money Laundering Detection: Flagging complex patterns of fund transfers that resemble laundering tactics.

Mobile Wallet Surveillance: Monitoring frequent low-value transactions that may indicate fraud or phishing activity.

These deployment pathways demonstrate the versatility of the proposed system. By aligning with operational needs and compliance frameworks, the solution not only enhances fraud detection accuracy but also empowers financial institutions to proactively respond to evolving security threats.

REFERENCES

[1] Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. Statistical Science, 17(3), 235–255.

[2] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321–357.

[3] Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. Decision Support Systems, 50(3), 602–613.

[4] Shen, W., Zhang, J., & Jiang, Y. (2020). A deep learning approach for detecting accounting fraud in publicly traded U.S. firms. Journal of Financial Crime, 27(1), 277–293.

[5] World Bank. (2023). The Global Findex Database 2023: Financial Inclusion, Digital Payments, and Resilience in the Age of COVID-19. World Bank Publications.