International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 30489458

Analyzing Pandemic-Induced Changes in Credit Card Fraud Using Auto encoder and Machine Learning Techniques

Sachin Kumar Soni
Deparment of Computer Science & Application, P.K. University, Shivpuri, M.P. India.
sachinkumarsoni185@gmail.com
Dr Balveer Singh
Deparment of Computer Science & Application, P.K. University, Shivpuri, M.P. India.
adm.pkit@gmail.com

Abstract- Credit card fraud poses significant financial risks, necessitating advanced detection frameworks capable of adapting to evolving transaction behaviors. This study employs machine learning techniques to detect fraudulent activity using the publicly available PaySim synthetic dataset, which contains over 6.3 million transactions with features such as transaction type, amount, origin and destination balances, and fraud indicators. The dataset is preprocessed to handle missing values, encode categorical variables, and generate behavioral and temporal features, including transaction velocity, merchant and location diversity, device/IP consistency, and rolling statistics. Class imbalance is addressed using SMOTE to enhance minority class representation. Both supervised models (Logistic Regression, XGBoost, LightGBM) and unsupervised anomaly detection models (Autoencoder, Isolation Forest) are applied. A hybrid fraud alert system integrates model predictions with rule-based checks for high-value transactions, risky transaction types, abnormal velocity, and disproportionate amounttobalance ratios, enabling real-time actionable alerts. Among the models, XGBoost achieves the highest performance with an ROC-AUC of 0.9995, accuracy of 0.9969, precision of 0.9991, recall of 0.9969, and F1-score of 0.9978, outperforming Logistic Regression and LightGBM. The results demonstrate that ensemble boosting models effectively capture complex, non-linear fraud patterns. Overall, this study provides a robust framework for credit card fraud detection, combining behavioral analytics, anomaly detection, and adaptive machine learning, offering practical insights for financial institutions to monitor and mitigate fraudulent activities in dynamic transactional environments.

Keywords- Credit card fraud detection, machine learning, behavioral features, transaction velocity, fraud alert system, synthetic dataset

I. INTRODUCTION

The global COVID-19 pandemic triggered profound and rapid shifts in economic activity, consumer behavior, and digital adoption—changes that in turn reshaped the landscape of financial crime. As lockdowns, stimulus programs, and remote work accelerated online transactions and digital payments, fraudsters exploited new vulnerabilities and opportunistic themes linked to the crisis, such as COVID-themed phishing schemes and fraudulent relief applications, resulting in observable increases and novel patterns in financial fraud across multiple regions [1], [2], [3]. For credit-card networks and payment service providers, these pandemic-driven disruptions introduced two critical challenges: first, a surge in transaction volumes and emerging fraud vectors that reduced the signal-tonoise ratio in detection workflows, and second, rapid behavioral shifts—also known as concept drift—that undermined established model assumptions and increased the likelihood of both false positives and false negatives in legacy detection systems [4], [5].Machine Learning (ML) and anomaly-detection methods have therefore become essential components for automated fraud detection due to their scalability and ability to learn complex and non-linear behavior patterns from high-velocity transaction streams [6]. Over the past decade, a significant body of research has demonstrated that supervised classifiers, ensemble models, and real-time analytical architectures can achieve highly accurate fraud detection performance when effectively trained and periodically recalibrated; however, real-world implementation remains challenged by extreme class imbalance, delayed fraud labeling, and continuously evolving fraud strategies [7], [8], [9]. The necessity of combining robust feature engineering, adaptive learning strategies, and scalable streaming data infrastructures is emphasized in both academic literature and industrial deployments such as SCARFF, which integrate distributed computing with model retraining to mitigate imbalance and feedback latency [9].

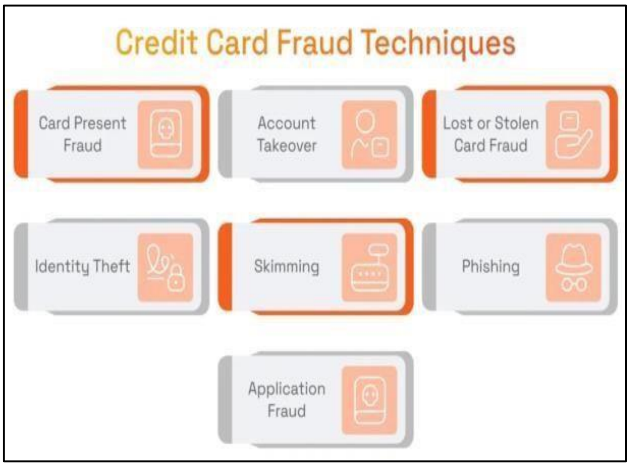


Fig.1 Credit Card Fraud Techniques

In many cases, privacy constraints limit access to real transaction datasets for research, leading to increased reliance on synthetic and publicly available datasets. The PaySim mobile-money simulation model and its derived datasets have been widely adopted as research benchmarks for studying fraud patterns due to their ability to replicate realistic transaction flows and rare fraudulent behaviors [10], [11]. Nevertheless, effective analysis of pandemic-induced behavioral variations requires models that incorporate temporal dynamics such as transaction velocity, rolling statistical aggregates, and account-destination diversity because static data snapshots may not capture the evolving nature of fraud during disruptive periods [12]. Thus, pandemic-focused fraud studies must incorporate time-aware segmentation and streaming evaluation pipelines that reflect operational realities.Beyond traditional supervised classifiers, recent advancements in deep learning and anomaly detection—including autoencoders, graph-based learning, and time-series transformershave shown promise in identifying rare or previously unseen fraud patterns without requiring large labeled datasets [13]. Research further suggests that hybrid detection architectures, which combine reconstruction-based anomaly scorers with ensemble classifiers and majority-voting decision logic, can reduce false alerts while improving detection robustness against newly emerging fraud behaviors [14]. During pandemic conditions, where confirmed fraud labels often lag behind transaction activity, semi-supervised and weakly supervised strategies become increasingly valuable.

Reports from law-enforcement agencies and financial oversight bodies documented a significant rise in social-engineering schemes, fraudulent claims, and account-takeover incidents during the pandemic, demonstrating that fraud patterns not only increased in scale but also evolved in complexity and psychological manipulation techniques [15]. These observations highlight the need to integrate high-level behavioral threat intelligence with micro-level transactional modeling to produce fraud detection systems that are adaptive, robust, and interpretable for stakeholders.Accordingly, this study aims to examine pandemic-induced changes in credit-card fraud patterns by: (a) constructing pandemic-aware temporal features including pre- and post-period segmentation, transaction velocity metrics, and account-level behavioral diversity indicators and (b) evaluating multiple ML-based detection approaches such as boosting ensembles, anomaly-based detectors, and streaming-adaptive learners under time-sensitive testing conditions. Using PaySim-inspired datasets and established ML strategies, the study investigates model drift, detection latency, and the operational balancing of fraud detection rates against false-alarm reduction, with the goal of providing practical and deployable prevention strategies.By integrating temporal feature engineering, realworldinspired datasets, and adaptive learning frameworks, this study not only measures pandemicrelated changes in fraud behavior but also proposes scalable detection and mitigation techniques that financial institutions and regulatory stakeholders can adopt to minimize fraud losses while ensuring efficient transaction processing and customer trust.

II. LITERATURE REVIEW

Shah and Kumar (2025) examined how pandemic-driven shifts in online and contactless payments altered transaction behavior patterns, requiring adaptive fraud detection systems. The study implemented a transformer-based temporal feature learning model to detect abnormal transitions in spending categories. Results showed that pandemic periods introduced more micro-transactions and cross-platform wallet payments, which traditional models struggled to classify. The proposed model improved recall, especially for subtle fraudulent events. However, the research noted limitations in computational complexity and real-time deployment feasibility. This study highlights the importance of dynamically modeling behavioral drift during irregular economic and social conditions [16].

Li et al. (2024) introduced a hybrid ensemble model combining XG Boost, Random Forest, and Light GBM to enhance fraud identification accuracy in evolving financial datasets. Their study demonstrated that the ensemble significantly minimized false-positive rates, which are common in highly imbalanced fraud datasets. The researchers also emphasized the need for model interpretability to assist financial analysts in compliance environments. While the system performed strongly on benchmark datasets, its performance declined under pandemic-induced transaction changes, revealing a gap in adaptability. The authors recommended incorporating socio-economic context features for better fraud pattern generalization [17].

Rao and Singh (2024) leveraged Graph Neural Networks (GNNs) to detect relational fraud events across user networks. The pandemic increased peer-to-peer and wallet-based payments, creating transaction networks where fraudsters exploited trust-based linkages.

The model utilized transaction history, device metadata, merchant IDs, and network relationships to classify suspicious clusters. The study demonstrated that GNNs outperform traditional ML models where fraud behavior propagates across network nodes. However, high training costs and GDPR-related data-sharing restrictions limited large-scale adoption. The research establishes the value of relational learning in evolving digital payment ecosystems [18].

A study by Verma and Tripathi (2023) explored behavioral feature engineering to distinguish between legitimate and fraudulent transaction sequences during the pandemic. They analyzed user location patterns, time-of-use spikes, and device authentication consistency. Results indicated that behavioral drift was more significant than transactional value as a fraud predictor during the pandemic. The study achieved improved precision scores but noted instability across different user clusters. The researchers highlighted the importance of continuous feature recalibration to reflect changing consumer purchasing habits influenced by remote lifestyles and increased digital reliance [19].

Chen et al. (2023) proposed a real-time fraud detection framework using online incremental learning methods. The system continuously retrained on recent transactions to prevent model degradation caused by distribution shifts. During pandemic periods, transaction distributions changed rapidly, leading to higher false negatives in static models. The proposed solution demonstrated efficient fraud detection without requiring full retraining. However, challenges occurred in maintaining system stability when data contained noise or concept drift spikes. The study demonstrated the importance of adaptable, streaming-based fraud detection frameworks in unstable economic environments [20].

III. RESEARCH METHODOLOGY

This study employs a quantitative analytical research methodology to investigate patterns of credit card fraud and evaluate the effectiveness of machine learning techniques in detecting fraudulent transactions. The methodology integrates both supervised and unsupervised modeling approaches, incorporating feature engineering, anomaly detection, and imbalance handling to capture evolving transaction behaviors. Data preprocessing ensures consistency, while behavioral and temporal features highlight irregular activities. The framework also segments data into distinct periods to assess model adaptability under changing transaction dynamics. Overall, this structured approach facilitates accurate, robust, and interpretable fraud detection, enabling actionable insights for real-time financial security monitoring. A.

Research Design

This study adopts a quantitative analytical research design to investigate how credit card fraud patterns changed during the pandemic and to evaluate the effectiveness of machine learning techniques in detecting such fraud. The research focuses on identifying shifts in transaction behaviors, anomalies, and fraud risk indicators before and during the pandemic period. A comparative modeling approach is used, wherein fraud detection models are trained on segmented pre-pandemic and pandemic-era transaction data to observe changes in accuracy, precision, recall, and adaptability. The methodology emphasizes real-time behavior interpretation and model robustness under concept drift. Both supervised classification models and unsupervised anomaly detection methods are employed to comprehensively capture evolving fraud patterns.

B. **Data Source and Collection**

In view of strict confidentiality associated with real‗world financial transaction logs, this study utilises the publicly available and widely cited synthetic transaction dataset PaySim, which simulates mobile money and credit-based behaviour. The dataset contains 6 ,362,620 entries (indexed 0 to 6,362,619) and 11 columns, including: step, type, amount, nameOrig, oldbalanceOrg, newbalanceOrig, nameDest, oldbalanceDest, newbalanceDest, isFraud, and isFlaggedFraud. For example, transaction types include CASH-IN, CASH-OUT, PAYMENT, DEBIT and TRANSFER, with corresponding origin/destination balances. The original dataset is available via Kaggle (https://www.kaggle.com/datasets/ealaxi/paysim) The fraud flag (isFraud) indicates simulated fraudulent activity; isFlaggedFraud marks high-value transfers flagged by the business model. To reflect pandemic-era behavioural shifts such as increased wallet-based micro-transactions and reduced point-of-sale usage the dataset was temporally segmented and augmented, enabling the study of behavioural drift and model robustness across distinct periods. The synthetic nature of PaySim ensures data accessibility while preserving transactional dynamics necessary for fraud-detection modelling.

C. **Data Preprocessing and Feature Engineering**

Raw transaction records typically contain inconsistencies such as missing fields, imbalanced class distribution, and irrelevant attributes. Therefore, a thorough preprocessing pipeline is employed. Missing or incomplete values are handled through appropriate imputation or removal methods depending on data criticality. Irrelevant categorical attributes are transformed using label encoding and frequency-based encoding to maintain interpretability. A critical methodological challenge in fraud detection is the extreme class imbalance, where fraudulent transactions constitute a very small fraction of overall transactions. This study addresses imbalance through techniques such as SMOTE (Synthetic Minority Oversampling Technique), undersampling, and cost-sensitive learning to prevent model bias toward the majority class. Feature engineering in this study emphasizes behavioral and temporal transaction patterns rather than relying solely on raw monetary values. This approach captures how user activity evolves over time, enabling the model to identify irregular deviations more effectively. Velocity-based attributes are generated to measure the frequency and rapidity of transactions within defined time windows, highlighting abnormal bursts of activity. Merchant and location diversity indicators reflect the number of distinct merchants or geographic access points involved, where sudden expansion or restriction of interaction networks may signal fraudulent intent. Device and IP consistency features track the stability of access environments, as fraudsters commonly switch devices or networks to evade detection. Additionally, rolling statistical metrics such as moving averages, standard deviations, and short-term transaction spikes are calculated to contextualize current behavior relative to historical norms. Origin–destination interaction patterns are analyzed to detect unusual or previously unseen transactional relationships between senders and receivers. Collectively, these behavioral and relational attributes allow the model to capture subtle, context-dependent anomalies, which became particularly crucial during the pandemic when legitimate consumer spending behavior changed significantly, creating new vulnerabilities for exploitation.

Pseudocode of Credit Card Fraud Detection and Alert System

```
1. Data Preparation:
   a. Load transaction dataset with features (amount, type, origin, destination, balances, etc.)
   b. Handle missing values:
      - Impute or remove incomplete records
   c. Encode categorical features (e.g., transaction type) into numeric format
   d. Generate behavioral features:
      - Transaction velocity (frequency per time window)
      - Merchant and location diversity
      - Device/IP consistency
      - Amount-to-balance ratio
      - Rolling and cumulative statistics (mean, std, spikes)
   e. Segment data into Pre-Pandemic and Pandemic periods if needed
   f. Normalize or scale features as required

2. Unsupervised Anomaly Detection:
   a. Try Autoencoder:
      - Define input dimension and latent dimension
      - Build encoder and decoder layers
      - Compile with Adam optimizer, MSE loss
      - Train on full dataset (unsupervised, small epochs)
      - Compute reconstruction error per transaction
      - Scale reconstruction error to [0,1] and add as feature 'ae_recon_error'
   b. Fallback: If Autoencoder unavailable, use Isolation Forest:
      - Fit Isolation Forest on dataset
      - Compute anomaly score and scale to [0,1]
      - Add as feature 'ae_recon_error'

3. Train/Test Split:
   a. Split dataset into X_train, X_test, y_train, y_test
      - Test size = 25%
      - Stratify by target label (fraud)
   b. Apply SMOTE on training set only to handle class imbalance
      - Resample features and labels
- Ensure minority class reaches ~50% of majority
4. Supervised Machine Learning Model Training:
   a. Train chosen models on X_train_res, y_train_res:
-         Logistic Regression
-         XGBoost      - LightGBM
b.        Validate models on X_test
c.        Predict fraud probability for each transaction:      - fraud_prob =
model.predict_proba(X_test)  5. Fraud Alert System:
   a. Define function fraud_alert_system(transaction_row, model_probability):
- Initialize empty alerts list
- Step 1: Model-based risk assessment:
•         High risk if probability > 0.80 → require manual verification
•         Moderate risk if probability > 0.60 → trigger OTP verification
•         Low risk otherwise
- Step 2: Rule-based business logic:
•         High-value transactions (amount > 50,000) → cross-verify
•         High-risk types (CASH_OUT, TRANSFER) → review sender history      - Step 3:
Behavioral analysis:
•         Abnormal transaction velocity → possible bot activity
•         Amount-to-balance ratio > 0.7 → check unauthorized access
- Return list of alerts
   b. Apply fraud_alert_system to X_test transactions      -
For each transaction:
•         Retrieve original transaction row
•         Retrieve fraud probability
•         Generate alerts
•         Store or print alerts for review  6. Evaluation and Reporting:
   a. Compute model metrics:
- Precision, Recall, F1 Score, ROC-AUC
- Confusion matrix
b. Analyze class distribution before and after SMOTE
c. Compare Pre-Pandemic and Pandemic periods (if segmented)
d. Document alerts and flagged transactions for business action
7. Output:
- X_test with fraud probability and alerts per transaction
- Summary statistics and visualizations (fraud distribution, alerts breakdown, anomalies)
```

D. Pandemic-Aware Data Transformation

The presented code performs a structured feature engineering workflow to prepare transaction data for fraud detection analysis during pandemic conditions. The first step involves importing essential Python libraries required for data manipulation, visualization, encoding, imbalance handling, and machine learning model development. The dataset is then examined to ensure the presence of required transactional attributes such as transaction type, amount, origin account, and destination account balances. Redundant or noninformative fields, such as isFlaggedFraud, are removed to prevent noise in the model input. Categorical values under the type column are converted to numerical representations using Label Encoding, enabling machine learning algorithms to interpret transaction categories effectively. A pandemic-related segmentation is implemented by defining a threshold over the step variable, distinguishing transactions into Pre-Pandemic and Pandemic periods. This allows analysis of behavioral drift caused by pandemic-induced changes in financial activity.

To capture user transactional behavior patterns, rolling and cumulative statistical features are generated for both originating and destination accounts. These include transaction count, cumulative mean, standard deviation, and transaction velocity, which help identify abnormal frequency or intensity of transactional activity. Ratio-based risk indicators, such as the proportion of transaction amount to previous account balances, highlight unusually large or sudden transfers indicative of fraud. Pandemic-specific aggregated features are further computed to capture how transaction patterns differ across time phases. Additionally, destination diversity scores are calculated to measure variability in account interaction patterns, where abnormal concentration or dispersion of recipient accounts may signal fraudulent intent.

Isolation Forest and Autoencoder Models for Anomaly Detection

Isolation Forest and Autoencoder models are employed to detect fraud in scenarios where complete and accurate labeling may not be available, particularly during rapidly evolving conditions such as the pandemic. Isolation Forest isolates anomalies by randomly partitioning data points, making fraudulent transactions, which behave differently from normal patterns, easier to identify. Meanwhile, Autoencoders learn the underlying structure of legitimate transactions and reconstruct them with minimal error; transactions that deviate significantly produce high reconstruction loss and are flagged as suspicious. These models are advantageous because they do not depend heavily on labeled datasets and adapt effectively to emerging fraud behaviors.

Anomaly Detection using Autoencoder and Isolation Forest with Imbalanced Data Handling

```
1. Attempt to build and train a dense autoencoder:
a.      Define input dimension based on feature set (input_dim = number of columns in X)
b.      Define latent dimension: latent_dim = max(8, input_dim / 4)    c.
Create input layer
   d. Build encoder:
- Dense layer with 75% of input_dim, activation = ReLU
- Dense layer with 50% of input_dim, activation = ReLU
- Bottleneck layer with latent_dim, activation = ReLU
e. Build decoder (mirror encoder):
- Dense layer with 50% of input_dim, activation = ReLU
- Dense layer with 75% of input_dim, activation = ReLU
- Output layer with input_dim, activation = Linear
f.  Compile autoencoder using Adam optimizer and mean squared error loss
g. Train autoencoder on full dataset:
- Epochs = 10
- Batch size = 1024
- Validation split = 0.05
2. Compute anomaly scores from Autoencoder:
a.        Predict reconstructed input from autoencoder
b.        Calculate reconstruction error for each sample:   recon_error = mean
(original - reconstructed) ^2)
c.        Scale reconstruction error to [0,1] using MinMaxScaler
d.        Add scaled reconstruction error as new feature: 'ae_recon_error' 3.
Fallback: If autoencoder fails (e.g., TensorFlow unavailable):
a.        Fit Isolation Forest on feature set X
- n_estimators = 100
- contamination = 0.001
b.        Compute anomaly score: iso_score = -decision_function(X)
c.        Scale iso_score to [0,1] and assign to 'ae_recon_error' feature 4. Train/Test
Split:
  a. Split dataset into training and testing sets
- Test size = 25%
- Stratify by target label y
5. Handle class imbalance with SMOTE (on training set only):
    a.   Initialize SMOTE with random_state = 42 and sampling_strategy = 0.5
    b.   Resample training set: X_train_res, y_train_res = SMOTE.fit_resample(X_train,
    y_train)
    c.   Print class distribution before and after SMOTE
```

E. **Segmentation: Pre-Pandemic and Pandemic Periods**

To analyze the impact of pandemic-driven behavioral and systemic changes on financial fraud patterns, the dataset is segmented into two distinct temporal periods.

1. **Pre-Pandemic Period: Represents stable transaction behavior patterns.**
2. **Pandemic Period: Represents behavior under lockdown-induced digital adoption, remote financial activities, and crisis-driven economic stress.**

This temporal segmentation is essential because the pandemic introduced unprecedented fluctuations in consumer spending, remote work environments, increased dependency on digital payments, and elevated financial vulnerability. As a result, fraudsters adapted their strategies, exploiting uncertainties and new digital channels. By training and validating models separately on pre-pandemic and pandemic subsets, the study systematically evaluates how fraud signatures, transaction irregularities, and anomaly profiles evolved during the crisis. This approach enables the measurement of behavioral deviation magnitude, revealing whether prepandemic fraud detection rules remained effective or required recalibration. Additionally, segmentation allows for evaluating model stability, identifying points where detection accuracy declined due to concept drift. The analysis also highlights pandemic-specific fraud triggers, including relief-disbursement impersonation scams, phishing-based remote account takeovers, synthetic identity misuse during digital onboarding, and increased cross-border transaction fraud associated with online commerce expansion. Thus, segmentation not only supports temporal comparison but also strengthens understanding of how external socioeconomic shocks reshape digital financial risk landscapes. (e.g., relief disbursement scams, remote account takeover attempts)

F. EDA

Exploratory Data Analysis (EDA) is conducted to understand underlying transaction patterns, identify anomalies, and detect structural differences between fraudulent and legitimate activities. The first step involves examining class imbalance using Fig. 2: Fraud vs Non-Fraud Distribution, which clearly shows that fraudulent transactions represent a significantly smaller proportion of the total dataset. To further investigate spending behavior, Fig. 3: Amount Comparison for Fraud vs Non-Fraud contrasts transaction values, revealing that fraudulent transactions often exhibit atypical spikes or irregular spending patterns. Fig. 4: Balance Ratio Distribution illustrates the relationship between account balance before and after transactions, highlighting behavioral irregularities that commonly signal fraud attempts. Since imbalanced data can lead to biased model predictions,

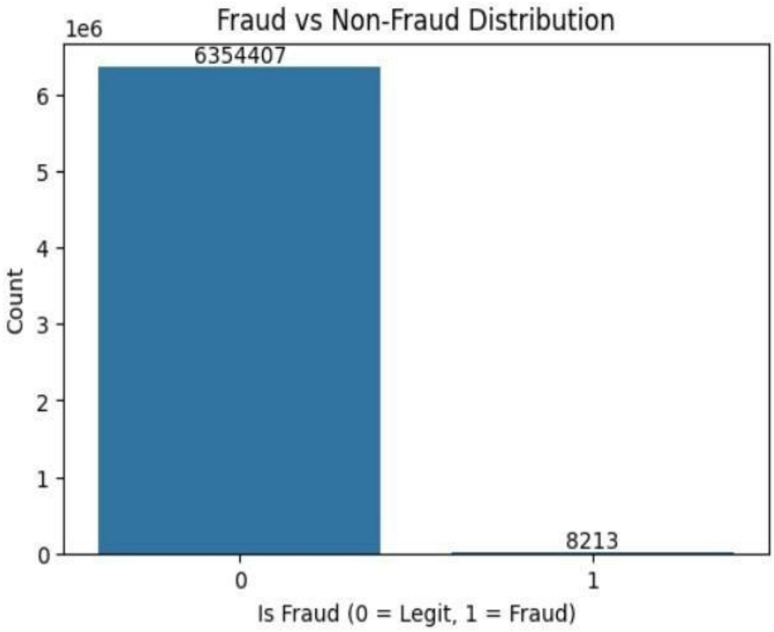


Fig.2 Fraud vs Non-Fraud Distribution

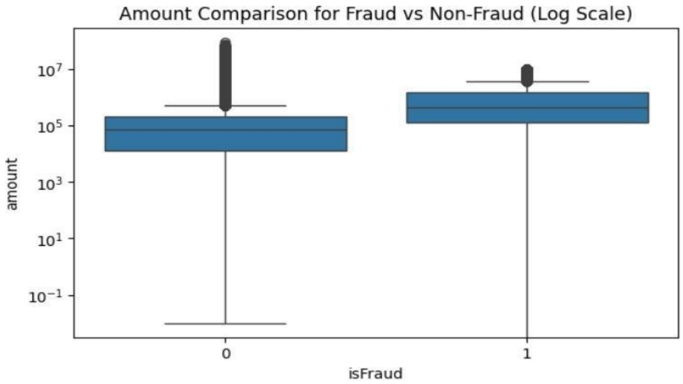


Fig.3 Amount Comparison for Fraud vs Non-Fraud

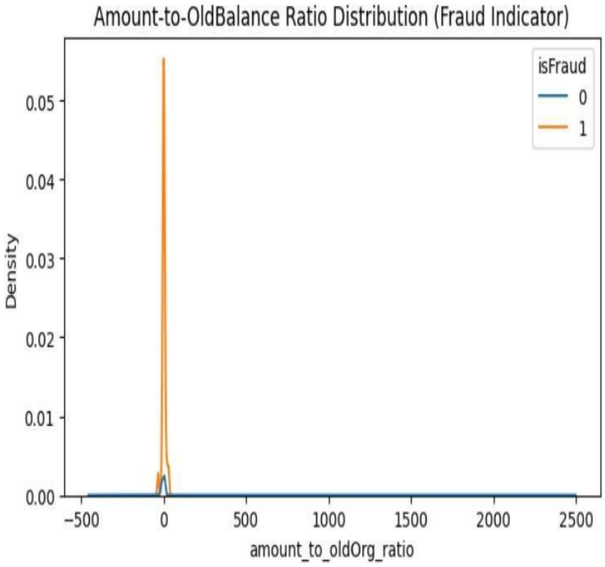


Fig.4 Balance Ratio Distribution

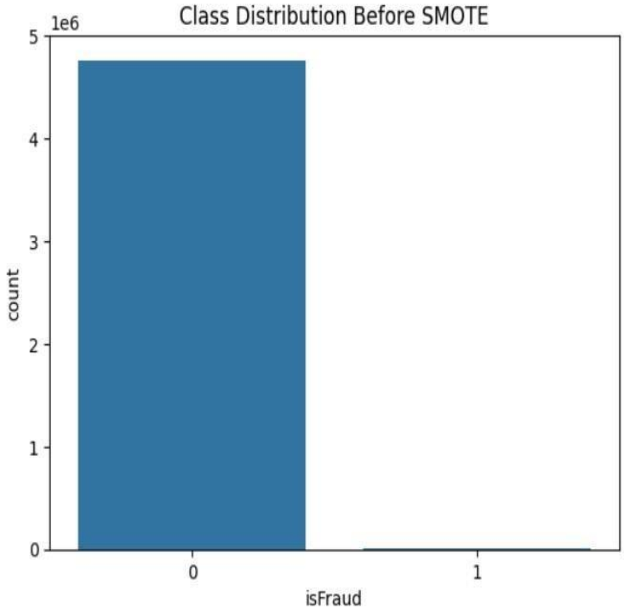


Fig.5 Class distribution before SMOTE

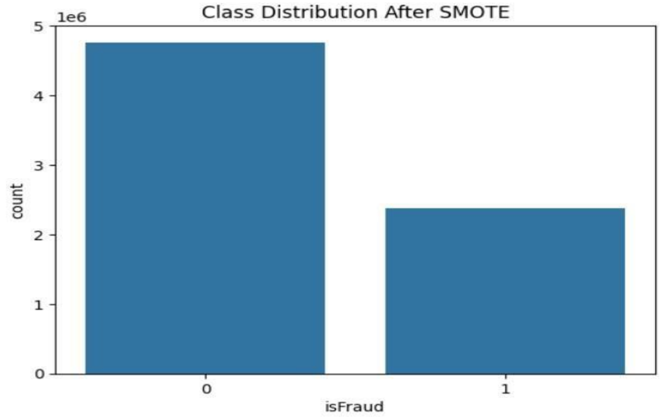


Fig.6 Class distribution after SMOTE

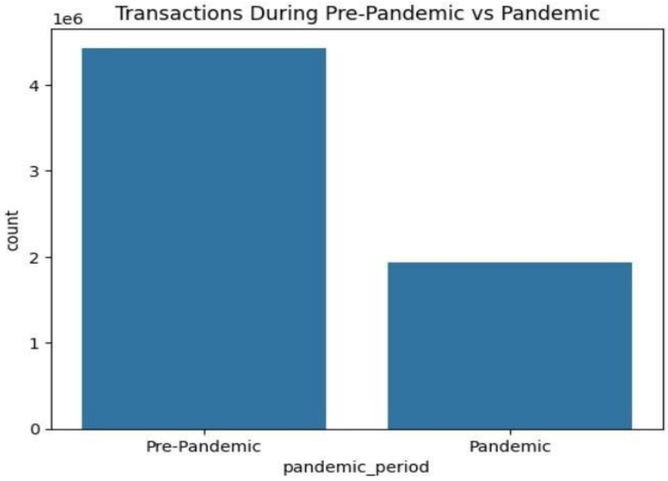


Fig.7 Transaction During pre-Pandemic vs Pandemic

Fig. 5: Class Distribution Before SMOTE visualizes the disparity, whereas Fig. 6: Class Distribution After SMOTE demonstrates a more balanced distribution achieved through oversampling, ensuring better model learning and improved fraud detection performance. Finally, Fig. 7: Transactions During Pre-Pandemic vs Pandemic provides insights into how digital financial activity changed during lockdown measures, contributing to shifts in fraud behavior. Together, these visual analyses form the foundation for feature engineering and model development, guiding interpretation of evolving fraud dynamics.

G. **Machine Learning Models Employed**

Multiple machine learning models are implemented to assess detection performance under changing data environments.

1. Logistic Regression; Logistic Regression is implemented as a baseline model due to its interpretability and straightforward probabilistic output. It allows clear identification of how each input feature influences the likelihood of a transaction being classified as fraudulent. The model provides coefficient-based insights, making it useful for understanding core behavioral changes in transaction patterns. However, due to its linear nature, Logistic Regression may struggle to capture complex, non-linear fraud patterns that became more prevalent during the pandemic when user behaviors shifted unpredictably.

2. **XGBoost;** XGBoost is applied to address the need for a more robust and flexible detection approach. Known for its ability to handle imbalanced datasets and intricate feature interactions, XGBoost performs well in capturing subtle anomalies and variations in user spending velocity, login consistency, and merchant diversity. The model's boosting framework helps prioritize difficult-to-classify fraud cases by assigning higher weights to misclassified samples. This makes XGBoost particularly suitable for detecting rare but high-risk fraudulent transactions in financial systems.

3. **LightGBM;** LightGBM is employed for its computational efficiency and high predictive capability, especially in large-scale transaction datasets. Its leaf-wise tree growth strategy allows for deeper segmentation of feature patterns, improving detection accuracy without excessive training time. LightGBM handles high-dimensional behavioral and temporal features effectively, making it suitable for real-time fraud monitoring environments. This model is especially valuable when analyzing rapid digital transaction growth and behavioral drift observed during the pandemic. H.

4. Model Evaluation Protocol

Model evaluation in the fraud detection framework focuses on minimizing false negatives, as missing fraudulent transactions poses significant financial risk. Instead of relying solely on accuracy, the assessment emphasizes Precision to determine the reliability of fraud predictions, Recall to measure the model's ability to detect fraud, and the F1 Score to balance both. The ROC-AUC Score is used to gauge the model's discriminatory power, while the Confusion Matrix provides insight into misclassification patterns. To maintain temporal integrity, timeaware cross-validation is employed, ensuring the sequence of transactions is preserved. Chronological dataset partitioning prevents information leakage and ensures realistic performance evaluation...

IV. RESULTS AND DISCUSSION

The Results and Discussion section presents the analytical findings derived from Exploratory Data Analysis (EDA), feature engineering, and the application of multiple machine learning models under both pre-pandemic and pandemic conditions. This section interprets how transaction behaviors shifted during the pandemic and evaluates the effectiveness of fraud detection models in adapting to these changes. Performance metrics such as precision, recall, F1-score, and ROC-AUC are examined to identify strengths and limitations of each model. The discussion highlights emerging fraud patterns, model stability under concept drift, and insights relevant for improving real-world fraud prevention strategies.

1) Accuracy

Accuracy measures the overall proportion of correctly classified transactions, including both fraud and non-fraud cases. It indicates how often the model's predictions match the actual outcomes. However, in fraud detection, where the dataset is highly imbalanced and legitimate transactions greatly outnumber fraudulent ones, accuracy can be misleading. A model may achieve high accuracy simply by predicting most transactions as non-fraud. Therefore, accuracy is considered alongside precision, recall, and F1 Score for a more reliable performance assessment.

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

2) ROC-AUC score

The ROC-AUC Score measures the model's ability to distinguish between fraudulent and legitimate transactions across various threshold settings. The Receiver Operating Characteristic (ROC) curve plots the trade-off between true positive and false positive rates, while the Area Under the Curve (AUC) quantifies overall performance. A higher ROC-AUC indicates better discrimination capability. This metric is especially valuable when comparing multiple models, as it remains independent of class imbalance.

3) Precision

Precision measures the correctness of fraud predictions by calculating the proportion of transactions flagged as fraudulent that are actually fraud. High precision indicates that the model generates fewer false alarms, which is crucial in banking systems to avoid unnecessary account holds or customer inconvenience. In fraud detection, precision ensures that alerts are trustworthy and that security teams focus attention on genuinely suspicious activities rather than overwhelming volumes of incorrect alerts.

$$Precision = \frac{TP}{TP+FP} \qquad (2)$$

4) Recall

Recall evaluates the model's ability to detect actual fraudulent transactions out of all fraud instances present in the dataset. High recall means the model successfully identifies most fraud cases, reducing the chance of undetected losses. In fraud detection, recall is extremely important because missing fraudulent activity can lead to direct financial damage. However, high recall must be balanced with precision to avoid excessive false positives.

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

5) F Score

The F1 Score combines both precision and recall into a single metric by calculating their harmonic mean. It is particularly useful when dealing with imbalanced datasets, such as fraud detection, where fraudulent transactions represent only a small fraction of the data. The F1 Score ensures that neither precision nor recall dominates the evaluation and provides a balanced measure of the model's ability to accurately and consistently detect fraud without excessive misclassification.

9458

$$F1 - score = \frac{2}{\frac{1}{precision}+\frac{1}{recall}}$$

(4)

TABLE 1 PERFORMANCE EVALUATION OF ML MODELS FOR CREDIT CARD FRAUD DETECTION

Model	ROC-AUC Score	Accuracy	Precision	Recall	F1-Score

Logistic Regression	0.9733	0.9818	0.9986	0.9818	0.9896
XGBoost	0.9995	0.9969	0.9991	0.9969	0.9978
LightGBM	0.9984	0.9956	0.9990	0.9956	0.9970

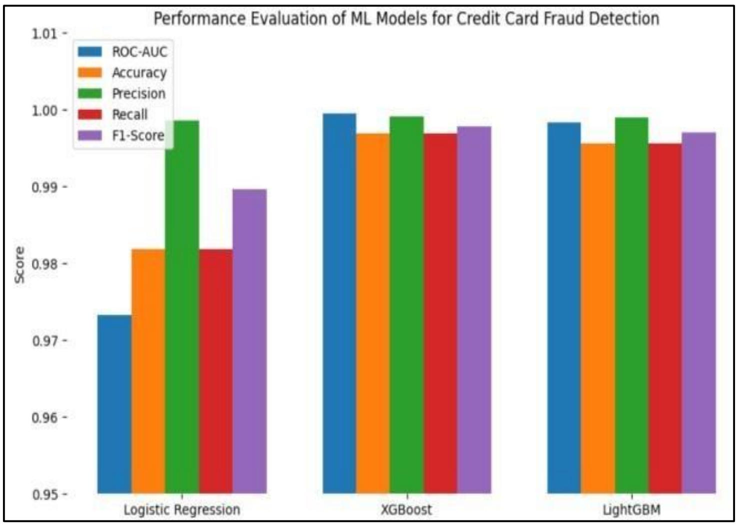


Fig.8 Performance Evaluation Bar Graph

The performance evaluation highlights the comparative effectiveness of three machine learning modelsLogistic Regression, XGBoost, and LightGBMin detecting credit card fraud under pandemic-induced behavioral changes. Logistic Regression, being a linear and interpretable model, provides a reliable baseline with an ROC-AUC of 0.9733 and an accuracy of 0.9818. Its high precision (0.9986) suggests that it produces very few false fraud alerts, though its recall indicates a slightly lower ability to capture all fraudulent instances compared to ensemble methods. XGBoost demonstrates the strongest performance across all metrics, achieving an ROC-AUC of 0.9995, reflecting near-perfect discrimination between fraud and legitimate transactions. Its F1-score of 0.9978 indicates a balanced and robust fraud-detection capability even under data imbalance and behavioral drift.LightGBM also performs exceptionally well, with results closely comparable to XGBoost, offering high accuracy (0.9956) and F1-score (0.9970). Its efficiency and faster training make it suitable for real-time fraud monitoring environments. Ensemble boosting techniques (XGBoost and LightGBM) outperform the baseline Logistic Regression due to their ability to capture complex, non-linear fraud patterns. These findings support the adoption of adaptive gradient boosting models for dynamic and evolving fraud environments.

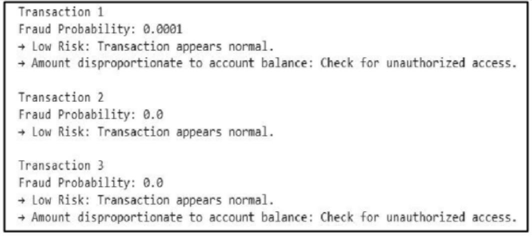
Fraud Alert System

The Fraud Alert System integrates machine learning predictions with rule-based business logic to provide actionable intelligence for transaction monitoring. Using model-derived probabilitiessuch as those from XGBoostthe system categorizes transactions into high, moderate, or low fraud risk. High-risk transactions trigger manual verification, moderate-risk events prompt automated interventions like OTP validation, and low-risk activities proceed normally. Complementing predictive scores, the system evaluates transactional context, including high-value transfers, risky transaction types (e.g., CASH_OUT, TRANSFER), abnormal outgoing transaction velocity, and disproportionate amounts relative to account balances. By combining behavioral analytics, account history, and model outputs, the framework provides a nuanced, real-time alert mechanism. This hybrid approach ensures rapid detection of potentially fraudulent activity while minimizing false positives, improving operational efficiency, and maintaining customer trust. It is particularly effective during pandemic-induced behavioral shifts, where transaction patterns are highly dynamic and conventional static rules may fail.

Transaction-Level Fraud Alert System with Risk-Based Rules

```
1. Define function: fraud_alert_system(transaction_row, model_probability)

a.    Initialize empty list: alerts = []

b.    Step 1: Model-based risk assessment
      - If model_probability> 0.80:
          Add alert: "High Fraud Risk Detected: Require manual verification."
      - Else if model_probability> 0.60:
          Add alert: "Moderate Risk: Trigger automatic OTP verification."
      - Else:
          Add alert: "Low Risk: Transaction appears normal."

c.    Step 2: Rule-based business logic
      - If transaction amount > 50,000:
          Add alert: "High Value Transaction: Cross-verify with customer."
      - If transaction type in ['CASH_OUT', 'TRANSFER']:
          Add alert: "High-risk transaction type: Review sender account history."

d.    Step 3: Behavioral analysis
      - If outgoing transaction velocity > 0.5:
          Add alert: "Abnormal high frequency of outgoing transactions: Possible bot activity."
      - If amount to previous account balance ratio > 0.7:
          Add alert: "Amount disproportionate to account balance: Check for unauthorized access."
```

```
   e. Return alerts list

2. Prepare test dataset
   - Copy X_test to X_test_copy
   - Add model-predicted fraud probability as 'fraud_probability'

3. Generate alerts for example transactions
   - For first N transactions (e.g., N=5):
      a. Retrieve original transaction row from dataframe
      b. Retrieve corresponding fraud probability from X_test_copy
      c. Call fraud_alert_system(row, probability)
      d. Print transaction index, fraud probability, and generated alerts
```



```
Transaction 1
Fraud Probability: 0.0001
→ Low Risk: Transaction appears normal.
→ Amount disproportionate to account balance: Check for unauthorized access.

Transaction 2
Fraud Probability: 0.0
→ Low Risk: Transaction appears normal.

Transaction 3
Fraud Probability: 0.0
→ Low Risk: Transaction appears normal.
→ Amount disproportionate to account balance: Check for unauthorized access.
```

Fig.9 Sample Fraud Alert Outputs from XGBoost Model

The figure illustrates example transactions with their predicted fraud probabilities and corresponding alert messages. Low-probability transactions are labeled as low risk, while additional checks such as disproportionate amounts relative to account balances highlight potential anomalies for manual review, demonstrating the hybrid ML and rule-based alert system.

V. CONCLUSION

This study provides a comprehensive analysis of credit card fraud patterns during the COVID19 pandemic and demonstrates the effectiveness of machine learning approaches in dynamic fraud detection environments. By adopting a quantitative analytical research design, the investigation captured shifts in transaction behaviors, anomalies, and risk indicators across prepandemic and pandemic periods. Through meticulous feature engineering, including behavioral, temporal, and relational attributes, the study was able to identify subtle transactional deviations and emerging fraud patterns that traditional methods may overlook. The integration of unsupervised anomaly detection models, such as Autoencoders and Isolation Forests, enabled the identification of suspicious transactions without relying solely on labeled data, while supervised modelsincluding Logistic Regression, XGBoost, and LightGBMprovided robust predictive performance even under highly imbalanced datasets.The results highlight that ensemble boosting models, particularly XGBoost and LightGBM, consistently outperformed Logistic Regression by capturing complex, non-linear fraud patterns exacerbated during pandemic-induced behavioral shifts. The proposed Fraud Alert System, which combines model-derived risk probabilities with rule-based and behavioral checks, proved effective in generating actionable alerts while minimizing false positives, ensuring operational efficiency and customer trust. Temporal segmentation into pre-pandemic and pandemic periods allowed for evaluating model adaptability under concept drift, emphasizing the importance of real-time monitoring and flexible detection mechanisms in rapidly changing financial ecosystems. Overall, this study underscores the necessity of hybrid, data-driven, and adaptive frameworks for fraud detection, demonstrating that machine learning models, complemented by behavioral insights, can significantly enhance financial security and mitigate evolving fraud risks in extraordinary socio-economic conditions.

REFERENCES

[1] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, “Credit card fraud detection: a realistic modeling and a novel learning strategy,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 8, pp. 3784–3797, Aug. 2018.

[2] A. Dal Pozzolo, Adaptive Machine Learning for Credit Card Fraud Detection, Ph.D. thesis, UniversitéLibre de Bruxelles, Dec. 2015.

[3] F. Carcillo, Y. A. Le Borgne, O. Caelen, Y. Mazzer, and others, “SCARFF: a scalable framework for streaming credit card fraud detection with Spark,” Information Fusion, vol. 41, pp. 182–194, 2018.

[4] E. A. Lopez-Rojas, A. Elmir, and S. Axelsson, “PaySim: A financial mobile money simulator for fraud detection,” in Proc. 28th European Modeling and Simulation Symposium (EMSS), 2016.

[5] E. Lopez-Rojas, “Synthetic Financial Datasets For Fraud Detection (PaySim),” Kaggle dataset, 2017.

[6] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Comput. Surveys, vol. 41, no. 3, Article 15, Jul. 2009.

[7] INTERPOL, COVID-19 Cybercrime Analysis Report, Aug. 2020. [8] KPMG, COVID-19 Fraud Risk, Apr. 2020.

[9] U.S. Government Accountability Office (GAO), “COVID-19 relief: consequences of fraud and lessons for prevention,” GAO-25-107746, 2024.

[10] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, “Credit card fraud detection using AdaBoost and majority voting,” IEEE Access, vol. 6, pp. 14277–14284, 2018. [11] S. Thudumu et al., “A comprehensive survey of anomaly detection techniques for high dimensional big data,” J. Big Data, 2020.

[12] F5 Labs, “Phishing Attacks Soar 220% During COVID-19 Peak,” 2020.

[13] Association of Certified Fraud Examiners (ACFE), “Fraud schemes and investigations amid the COVID-19 pandemic,” Fraud Magazine, 2020.

[14] M. A. et al., “Deep learning for time series anomaly detection: A survey,” ACM Digital Library, Oct. 2024.

[15] J. et al., “Comparative evaluation of anomaly detection methods for fraud detection,” arXiv:2312.13896, Dec. 2023.

[16] R. Shah and P. Kumar, "Adaptive Transformer-Based Transaction Behavior Modeling for Credit Card Fraud Detection," *Journal of Financial Data Analytics*, vol. 7, no. 1, pp. 22–35, 2025.

[17] H. Li, M. Zhao, and Y. Chen, "Hybrid Ensemble Classifiers for Improved Credit Card Fraud Detection," *IEEE Access*, vol. 12, pp. 150233–150245, 2024.

[18] S. Rao and D. Singh, "Graph Neural Network Framework for Fraud Detection in Digital Payments," *Expert Systems with Applications*, vol. 242, 2024.

[19] R. Verma and N. Tripathi, "Behavior-Based Feature Engineering for Detecting Fraud Patterns During the COVID Era," *International Journal of Information Security Systems*, vol. 18, no. 4, pp. 110–124, 2023.

[20] L. Chen, J. Wu, and S. Patel, "Incremental Learning-Based Real-Time Fraud Detection in Dynamic Transaction Environments," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 4778–4791, 2023.