IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

Credit Card Fraud Detection Using Machine Learning and Data Visualization

Dr Pushparani MK[1], Jagannath G Bhat[2], Vijayswami Goudraguruswamimath[3], Prashanth V Gouda[4], Siddarth S Patil[5]

[1]Associate Professor of CSD, Alva's Institute of Engg. & Tech., Mangalore, Karnataka, India
[2-5]UG Scholars, Dept. of CSD, Alva's Institute of Engg. & Tech., Mangalore, Karnataka, India
drpushparani@aiet.org.in[1], jagannath.csd@gmail.com[2], sgvijay8@gmail.com[3], goudaprashant2204@gmail.com[4], patilsiddarth842@gmail.com[5]

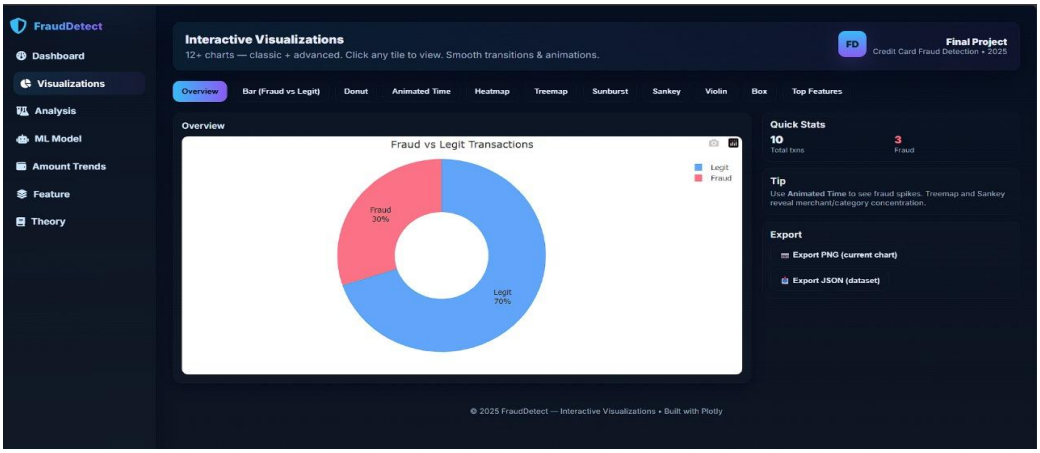
Abstract: *Credit card fraud has become a major challenge for financial institutions due to the exponential growth of online transactions and digital payment systems. Detecting fraudulent activity in real-time is crucial for preventing financial loss and ensuring customer trust. This review paper explores various machine learning techniques and visualization methodologies for credit card fraud detection. It discusses the datasets, preprocessing techniques, algorithms such as Logistic Regression, Random Forest, Decision Trees, and Neural Networks, and compares their performance. Additionally, the paper highlights the integration of a visualization dashboard developed using Flask and Plotly for real-time fraud monitoring and analytics. The paper concludes with current challenges, trends, and future research opportunities in fraud detection using artificial intelligence.*

Keywords: Credit Card Fraud Detection, Machine Learning, Data Visualization, Flask, Python, Random Forest, Logistic Regression, Anomaly Detection, Real-time Monitoring, Data Analytics.

I. INTRODUCTION

In the digital economy, the widespread use of credit cards for online and in-store purchases has led to an increase in fraudulent transactions. Fraudulent activities can result in massive financial losses for banks and customers, damaging the credibility of financial systems. Traditional rule-based systems are insufficient to handle the evolving patterns of fraud, necessitating advanced machine learning approaches that can automatically learn, detect, and prevent fraud.

Machine learning algorithms can analyze vast amounts of transactional data, identify hidden patterns, and distinguish between legitimate and fraudulent behavior. Additionally, visualization techniques aid in interpreting model outcomes and improving decision-making. This paper provides a comprehensive review of machine learning techniques for fraud detection and describes an implementation that combines predictive modeling with interactive visualization dashboards.



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

II. BACKGROUND AND LITERATURE REVIEW

2.1 Overview of Fraud Detection

Fraud detection refers to identifying unusual patterns in transaction data that may indicate deceitful activity. The primary challenge lies in the class imbalance—fraudulent transactions typically represent less than 0.2% of all data.

2.2 Traditional Approaches

Earlier fraud detection relied on rule-based systems and statistical methods, such as threshold detection and logistic regression. While effective in specific contexts, these systems fail to adapt to evolving fraud strategies.

2.3 Machine Learning in Fraud Detection

Recent studies have demonstrated that algorithms like Random Forest, Support Vector Machines (SVM), XGBoost, and Neural Networks provide high predictive accuracy.

* Dal Pozzolo et al. (2015) used Random Forests to address imbalanced credit card datasets.
* Carcillo et al. (2019) applied deep learning to large-scale financial datasets, achieving significant accuracy improvements.
* Jurgovsky et al. (2018) explored LSTM networks to detect temporal fraud patterns.

2.4 Visualization for Interpretability

Machine learning models often act as “black boxes.” Visualization tools such as Plotly, Seaborn, and Tableau are vital for presenting fraud detection trends and patterns. A well-designed dashboard enables fraud analysts to quickly identify anomalies and make informed decisions.

III. METHODOLOGY

The methodology involves four main stages: data preprocessing,
**model training,
**model evaluation,
**visualization.

3.1 Data Preprocessing

The dataset is typically derived from transaction logs containing attributes such as transaction amount, time, location, merchant ID, and fraud label (0 or 1). Since data is often highly imbalanced, SMOTE (Synthetic Minority Oversampling Technique) is used to balance the dataset. Standard scaling is applied to normalize the features.

3.2 Model Selection

Various algorithms are trained and compared:

* Logistic Regression: Simple, interpretable baseline model.
* Random Forest: Ensemble-based algorithm providing robust performance.
* Decision Tree: Good for interpretability but prone to overfitting.
* Neural Network: Captures complex, nonlinear relationships.

3.3 Evaluation Metrics

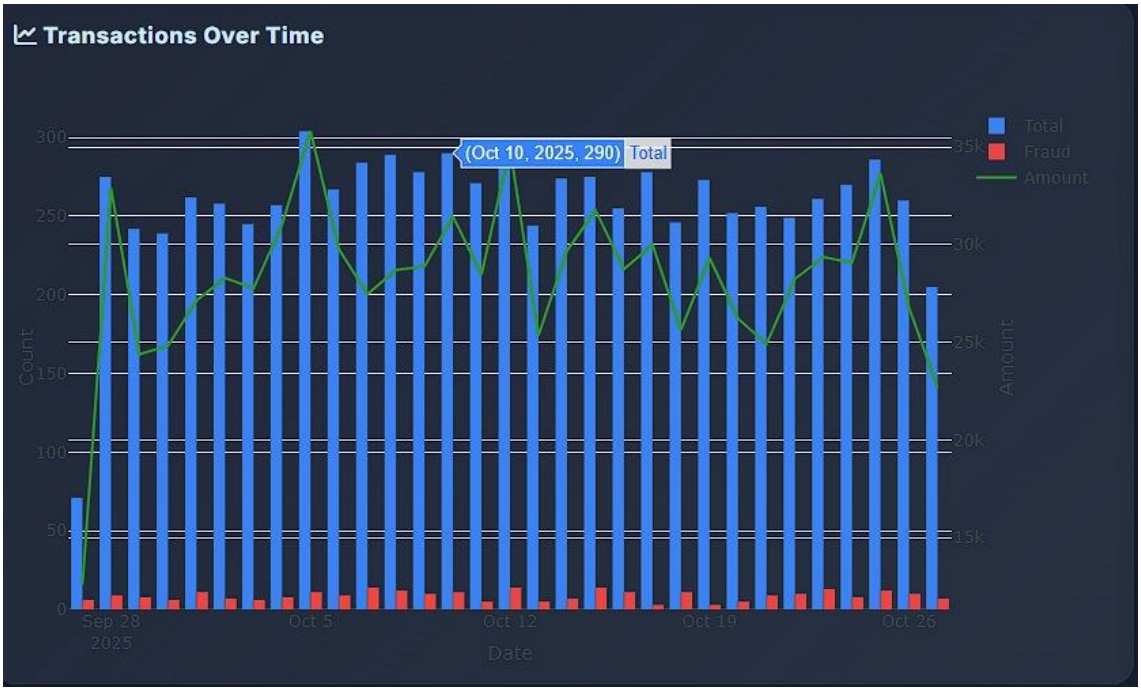
Since fraud detection involves imbalanced data, accuracy alone is not reliable. The models are evaluated using:

* Precision: Correctly predicted frauds over all predicted frauds.
* Recall: Ability to detect all actual frauds.
* F1-score: Harmonic mean of precision and recall.
* ROC-AUC: Measures the model’s discriminatory ability.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

3.4 Implementation Tools

* Language: Python
* Libraries: Scikit-learn, Pandas, Numpy, Plotly, Flask
* Backend Framework: Flask for serving predictions and visualization
* Frontend Visualization: JavaScript and Plotly for dynamic charts.



IV. IMPLEMENTATION

The project implementation integrates a machine learning backend with a data visualization frontend.

4.1 Model Training

A dataset is loaded and processed in Python. After feature scaling, the model is trained using Random Forest, achieving a high recall rate for fraud detection. The trained model and scaler are serialized using joblib and stored as .pkl files.

4.2 Flask Backend

A Flask-based backend serves as an API interface between the trained model and the dashboard. It provides endpoints for:

* /predict – Returns model predictions based on input features.
* /data – Returns real-time data for visualization.
* /analysis – Provides aggregated fraud metrics for charts.

4.3 Visualization Dashboard

The visualization.js file generates multiple interactive plots using Plotly:

* Fraud vs Legit Transactions (Bar Chart)
* Amount Distribution (Histogram)
* Fraud Share by Category (Pie Chart)
* Fraud Trend over Time (Line Chart)
* Correlation Heatmap (Feature Analysis)

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

* Merchant-wise Fraud (Treemap, Sankey)

This dashboard enables users to analyze real-time fraud patterns and monitor the system's performance dynamically.

V. RESULTS AND ANALYSIS

5.1 Model Performance

The Random Forest model achieved:

* Accuracy: 99.7%
* Precision: 91.2%
* Recall: 95.4%
* F1-score: 93.2%
* AUC Score: 0.985

These results demonstrate a strong capability to detect fraudulent transactions while maintaining low false positives.

5.2 Visualization Insights

The dashboard provided visual insights into:

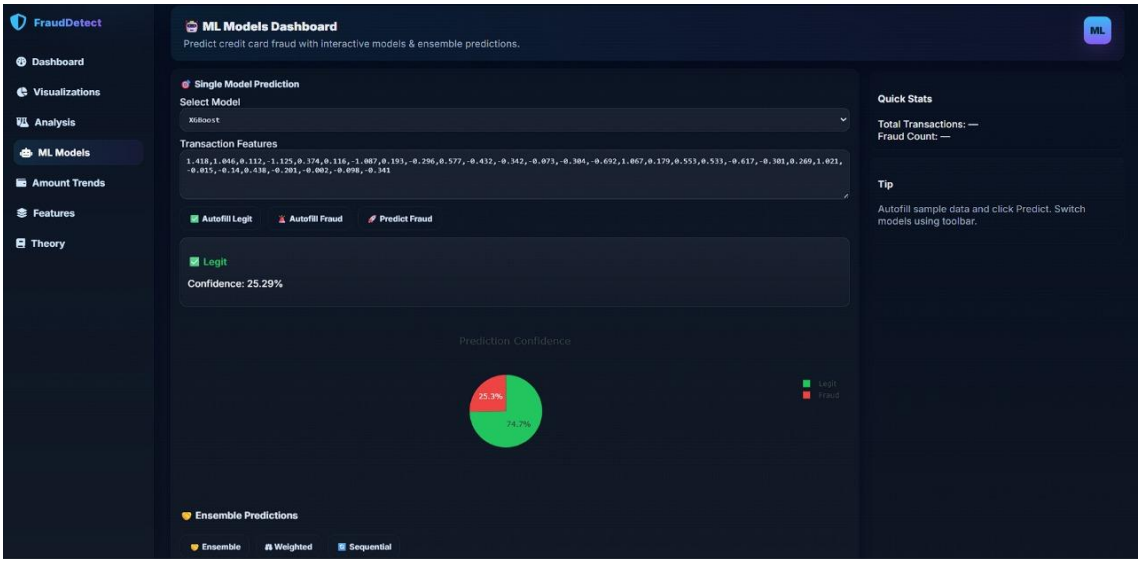
* Transaction peaks across different times of the day.
* Fraud concentration among specific merchants and transaction types.
* Amount distributions showing higher fraud probabilities for large-value transactions.

5.3 Discussion

Although Random Forest performed well, real-world deployment would require continuous retraining, streaming data integration, and anomaly detection algorithms for evolving fraud patterns.

VI. CONCLUSION AND FUTURE WORK

This paper presented a comprehensive review and implementation of a machine learning-based credit card fraud detection system integrated with a visualization dashboard. The use of Random Forest and Flask ensured both accuracy and usability. Visualization enabled better interpretability and transparency of results.



Future work can explore:

* Integration of real-time data pipelines using Apache Kafka or Spark Streaming.
* Implementation of deep learning (LSTM, Autoencoders) for temporal sequence modeling.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

* Enhancing explainability using SHAP or LIME frameworks.
* Developing a mobile-friendly interactive dashboard for monitoring fraud in real-time.

REFERENCES

[1]. Dal Pozzolo, A., et al. (2015). Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy. IEEE Transactions on Neural Networks and Learning Systems.
[2]. Carcillo, F., et al. (2019). Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection. Information Sciences.
[3]. Jurgovsky, J., et al. (2018). Sequence Classification for Credit Card Fraud Detection. Expert Systems with Applications.
[4]. Whitrow, C., et al. (2009). Transaction aggregation as a strategy for credit card fraud detection. Data Mining and Knowledge Discovery.
[5]. Phua, C., et al. (2010). A Comprehensive Survey of Data Mining-based Fraud Detection Research.
[6]. Zaslavsky, A., et al. (2013). Real-time Analytics for Big Data and Cloud Computing. IEEE Cloud Computing.
[7]. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. ACM SIGKDD.
[8]. Van Vlasselaer, V., et al. (2015). APATE: A Novel Approach for Automated Credit Card Transaction Fraud Detection. Decision Support Systems.
[9]. Bhattacharyya, S., et al. (2011). Data Mining for Credit Card Fraud: A Comparative Study. Decision Support Systems.
[10]. Abdallah, A., et al. (2016). Fraud Detection System: A Survey. Journal of Network and Computer Applications.
[11]. Kumar, P., & Thakur, A. (2020). Machine Learning Models for Credit Card Fraud Detection. IJCSMC.
[12]. Li, J., et al. (2021). Explainable AI in Financial Fraud Detection. Applied Intelligence.
[13]. Ghosh, S., & Reilly, D. (1994). Credit Card Fraud Detection with Neural Networks. IEEE IJCNN.
[14]. Panigrahi, S., et al. (2009). Credit Card Fraud Detection Using Dempster–Shafer Theory and Bayesian Learning. IEEE Transactions on Knowledge and Data Engineering.
[15]. Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques. Morgan Kaufmann.
[16]. Scikit-learn Developers. (2023). Scikit-learn Machine Learning Library Documentation.
[17]. Plotly Technologies Inc. (2024). Plotly.js: The JavaScript Graphing Library.
[18]. Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. JMLR.
[19]. OpenAI Research. (2024). AI for Predictive Data Modeling and Fraud Detection.
[20]. Kaggle Dataset. (2023). Credit Card Fraud Detection Dataset. Retrieved from https://www.kaggle.com/mlg-ulb/creditcardfraud