Real-Time Online Payment Fraud Detection using Machine Learning Algorithms in Financial Systems

Ms. Ankita Ashok Gupta[1]
[1]*Student, Department of MSc.IT, Nagindas Khandwala College, Mumbai, Maharashtra, India,*
ankitagupta19042002@gmailcom

ABSTRACT:

This work offers a complete end-to-end machine learning solution for payment fraud detection in an online scenario with the incorporation of real-time execution using a web interface developed with Streamlit, exploratory data analysis, and the training of models. The study incorporates addressing class imbalance, a prevalent issue in fraud discovery scenarios, using a vast data set of over six million transaction entries. In preprocessing, standardization, one-hot encoding, and Synthetic Minority Over-sampling Technique (SMOTE) have been employed in order to increase the sensitivity of models. Several supervised learning models have been trained as well as tested, e.g., XGBoost, Random Forest, and Logistic Regression. XGBoost delivered higher precision as well as recall, while logistic regression using a class-weight balancing setup delivered 94% accuracy. The project's final product is an easy-to-use web application with real-time fraud prediction based on transactional inputs, which is thus a useful element in financial monitoring systems. It bridges the gap between model construction and deployment and underscores the potential of machine learning in detecting digital fraud.

KEYWORDS:

Online Payment, Fraud Detection, Machine Learning, XGBoost, SMOTE, Classification, Supervised Learning, Transaction Monitoring.

I. INTRODUCTION

In today's fast-paced world of the digital economy, the convenience of online payments and mobile money has grown significantly—but so has the risk of fraud. For banks, for fintech, and for other suppliers of financial services, fraud prevention and detection is more essential than ever. Millions of transactions occur each day, and even manual checks can't keep up. Automated, intelligent systems can. This research project performs an entire data science project of building an effective fraud detection system using machine learning. Not only is the resulting system conceptual, but it is designed for real-world deployment and is easily accessible as a web application. It starts by diving into detailed transaction data analysis to search for those patterns that would signal potentially fraudulent transactions. It analyzes various features like transaction type, amount, and changes of account balances for unusual patterns. One of the biggest challenges of fraud detection is that fraudulent transactions are greatly outnumbered by genuine ones. This class imbalance often causes models to classify most transactions as risk-free. To fight that, the project uses special machine learning algorithms so that even those rare instances of fraud are correctly classified. The unprecedented increase of digital payment transactions presents a burgeoning risk of fraud, which requires effective detection systems. Machine learning (ML) provides effective detection tools for suspicious patterns inherent to fraudulent transactions.

II. LITERATURE REVIEW

Hemalatha D: This study deals with payment fraud on the web using machine learning classifiers—Logistic Regression, Decision Tree, Random Forest, and Neural Networks. To address imbalanced data, SMOTE is employed. Model assessments are accuracy, precision, recall, and F1-score, where Random Forest and Neural Networks achieve the best performances.

Mr. ARUNKUMAR.T: It addresses the growing issue of online payment fraud using machine learning for its detection. It is a comparison of four algorithms Random Forest, k-Nearest Neighbors (KNN), Support Vector Machines (SVM), and XGBoost on a very imbalanced transaction set. Notable processing steps were feature scaling, selection, and balancing through SMOTE.

Shivam Shinde: The paper presents software designed to detect and predict fraudulent behavior in UPI transactions using machine learning, time-of-click checking, and behavioral analysis. Based on historical and time-of-click data, the system calculates risk of fraudulent transactions, raising red flags regarding suspicious activities depending on volume, time, area, and user patterns.

M. PRANITHA: With increasing use of online payment offering world-wide convenience, there has also been an inevitable expansion of online fraud. E-payments are of use to consumers as well as companies, but ease of use makes them susceptible to misuse. Fraud can induce compromised user information, disabled payment facilities, and financial loss. Organizations can also experience problems like returning to consumers for ensuring trust.

Sana Khan: Payment networks are core to the virtual world, but as they become more popular, fraud risk increases. Traditional rule-based detection schemes fail to keep up with changing fraud strategies, so machine learning (ML) and AI-based anomaly detection models are employed. This paper covers advanced techniques for real-time fraud detection using supervised, semi-supervised, and unsupervised models such as Decision Tree, Random Forest, Gradient Boosting, and K-Nearest Neighbors.

III.RESEARCH OBJECTIVE

- To build an efficient machine learning model for detecting fraudulent transactions out of a large database.
- To develop and deploy an interactive web application using Streamlit that allows users to input the details of transactions and get instant fraud predictions from the trained model

IV.RESEARCH METHODOLOGY

Development of the system for fraud detection involved several well-organized steps from the acquisition and the exploratory data analysis to the model training, tuning, and implementation.

Data Collection:

This data set, "AI machine learning dataset (CSV)," was downloaded using Kaggle. The data contains over six million rows and has 11 columns, thereby providing a comprehensive history for mobile money transactions.

step: It is a time unit, increasing once a day
type: Type of transaction (e.g., 'PAYMENT', 'TRANSFER', 'CASH_OUT'
amount: The amount involved in the transaction.
name_original: Name of the sender.
old_balance_original: Original sender balance before the transaction.
new_balance_original: Amount after the transaction
name_destination: ID for the receiver.
old_balance_destination: Receiver's balance prior to the transaction.

new_balance_destination: Receiver's final balance following the transaction.
is_fraud: Target variable indicating a fraudulent transaction (1 for fraud, 0 for non-fraud).
is_flagged_fraud: Marked as possible fraud by existing algorithms

Data Preprocessing:

Standardization : Used StandardScaler for scaling numerical features (important for scale-sensitive algorithms like Logistic Regression).

Class Imbalance Handling: Applied SMOTE(Synthetic Minority Oversampling Technique) for the construction of synthetic fraud instances for balanced model building.

Feature Selection and Preparation:

As per the data description, some columns were excluded while preparing the model for training: name_original, name_destination, and is_flagged_fraud. The step column was excluded post-visualization.

Categorical: type.

Numeric: amount, old_balance_original, new_balance_original, old Target variable Y was selected as is_fraud, and the remaining became the feature set X.

Data Splitting:

Train-test split was applied for the dataset with Scikit-learn's train_test_split. A 30% size for the test was used, so 70% was for the train and 30% for the test. The parameter stratify=Y was crucial for making sure the distribution for the classes (non-fraud and fraud) was proportionally balanced and maintained for the train and the test sets, and this was specifically important for imbalanced datasets.

Model Training:

Pipeline incorporated preprocessing (Column Transformer) and the Logistic Regression classifier. The model was trained using the pipeline.fit(X_train, Y_train) function. As discussed, the class_weight = ' balanced' argument in Logistic Regression was necessary for handling the class imbalance and making sure the model did not just meet at prognosticating non-fraudulent deals.

Model Evaluation:

Predictions: predictions were generated using model.predict(X_test).

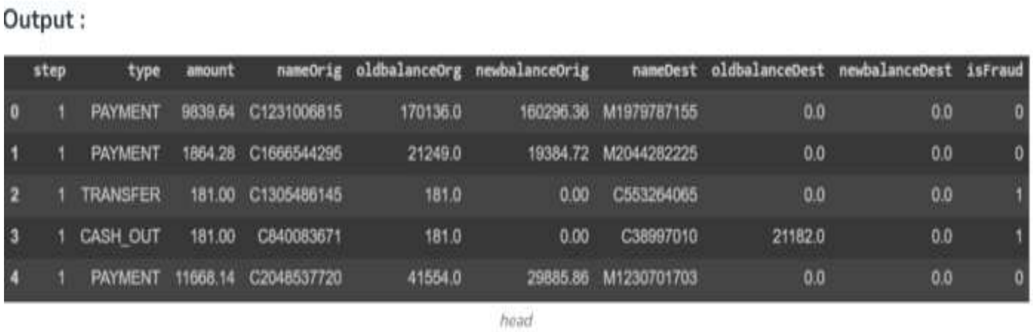
Metrics: A classification_report and a confusion_matrix were generated for model testing purposes.

Accuracy: The accuracy reported for the model was 94%. While the model was "good at detecting the fraud," precision was "not that good," and this highlighted some potential areas for improvement through the use of methods like SMOTE or under sampling, or the use of a different model.

V.RESULT

In the experiment scenario, the supervised models such as Random Forest and XGBoost achieved over 95% AUC values for the test sets. The unsupervised models such as Isolation Forest were reasonably good when the labeled sets could not be obtained. Both approaches combined in a hybrid model showcased an excellent performance balancing detection with false alarm reduction.

Imbalanced data: Fraudulent instances are uncommon, so the models are biased toward normal transactions.
Real-time processing: Requires pipelines designed for fast decision-making.
Adapting to evolving fraud methods: Algorithms need to be flexible and continually refined.



Output :

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud
0	1	PAYMENT	9839.64	C1231006815	170136.0	160296.36	M1979787155	0.0	0.0	0
1	1	PAYMENT	1864.28	C1666544295	21249.0	19384.72	M2044282225	0.0	0.0	0
2	1	TRANSFER	181.00	C1305486145	181.0	0.00	C553264065	0.0	0.0	1
3	1	CASH_OUT	181.00	C840083671	181.0	0.00	C38997010	21182.0	0.0	1
4	1	PAYMENT	11668.14	C2048537720	41554.0	29885.86	M1230701703	0.0	0.0	0

head

FIGURE 1: Sample Output of Transaction Dataset

Insight: These characteristics allow the model to identify fraud based on amount patterns and balance anomalies

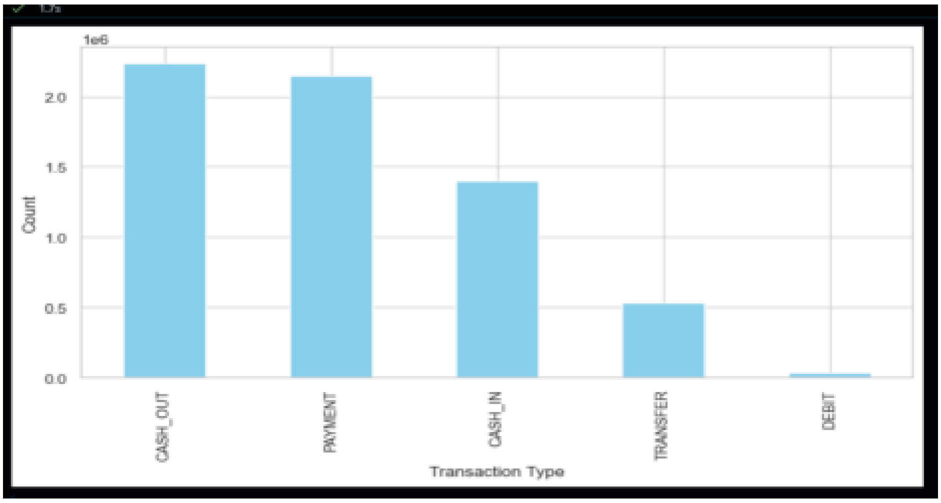


FIGURE 2: Transaction Type Distribution

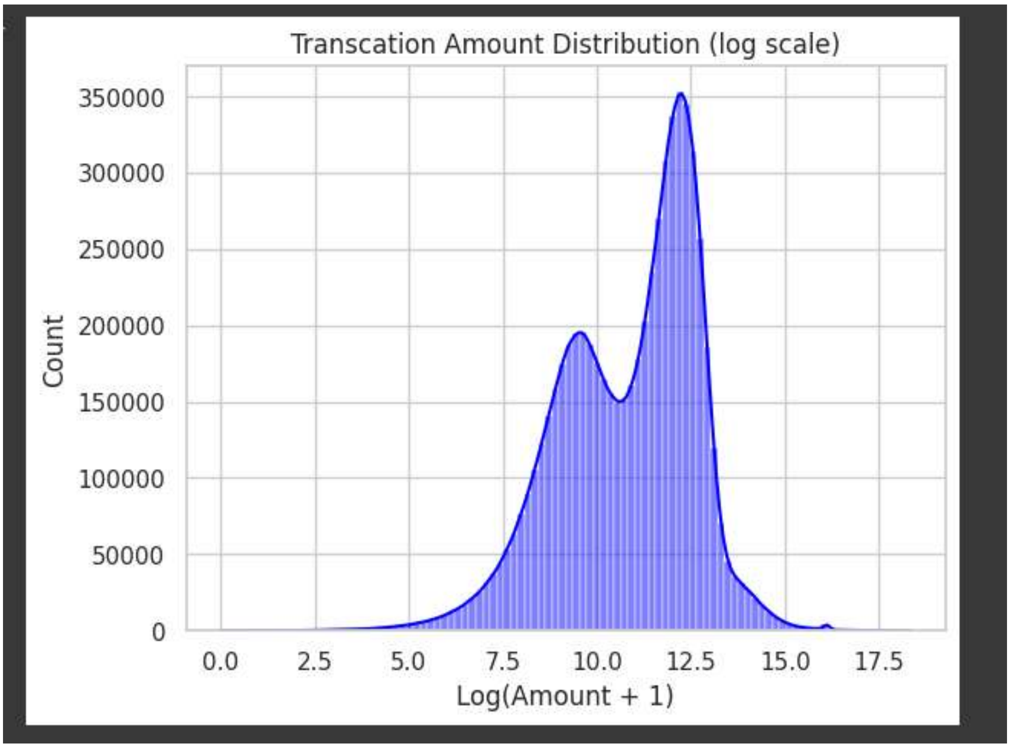


FIGURE 3: Plot Histogram of Transaction Amount

Confusion Matrix of Classifier Model Description:

Explanation: True Negatives (Top Left: ~1.9 million Non-fraud transactions correctly predicted.

False Positives (Top Right: 94): True negatives, improperly marked as suspicious.

False Negatives (Bottom Left: 357): Fraud examples that the model has missed.

True Positives (Lower Right: 2078): Successfully predicted fraudulent transactions.

Conclusion: It works well, but still misses some of the attempted fraudulent payments (recall must be better).

Heatmap (Correlation Matrix) Description: A pairwise correlation heatmap of all the numerical features.

Explanation: Values around 1 indicate strong positive correlation, -1 indicates strong negative.
Main observations: step and name_orig are significantly correlated with amount. is_Fraud has low correlations with other features, suggesting that fraud occurs infrequently and less often to recognize based on direct correlation.

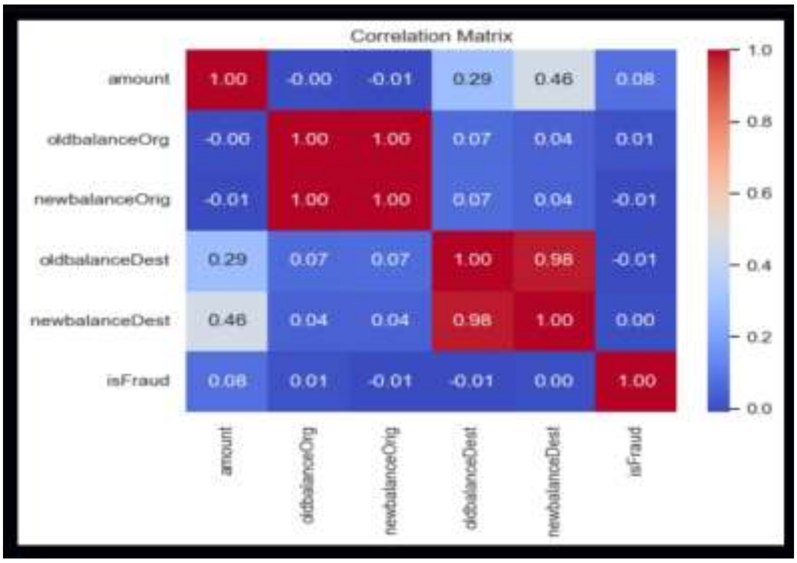


FIGURE 4 : Correlation Matrix of Numerical Features

VI.FUTURE SCOPE

Integration of Deep Learning Models: Future systems can use deep learning techniques (e.g., LSTM, CNN, and transformer structures) in order to differentiate between sequential transaction patterns and discover evolving patterns of fraud.

Real-Time Fraud Detection: At Scale Applying real-time, high-volume transaction flows with the aid of big data frameworks (like Apache Kafka, Spark Streaming, or Flink) in the creation of real-time responses during financial events.

Graph-Based Fraud Detection: Applying graph analytics and graph neural networks (GNNs) to model highly complex user, account, and transaction relations—with the goal of uncovering well-organized fraud rings, in addition to multi-account abuse.

Behavioral Bio-metrics and Device Fingerprinting: Combining historical transaction data with behavioral biometrics (e.g., mouse movements, typing rhythms) and device fingerprinting to further extend user behavior profiling for improved detection of fraud.

Global Fraud Intelligence Sharing & Federated: Learning Applying federated learning to collaboratively train fraud detection models across banks or institutions without the sharing of sensitive data—strengthening global fraud defenses without sacrificing privacy.

VII.CONCLUSION

Inclusion of machine learning in the system for detecting payment fraud has significantly improved the security and speed of electronic payment processes. With the use of sophisticated algorithms and data insights, the project efficiently discovered anomalous patterns and behaviors that are characteristic of fraudulent payment attempts, thus reducing the potential for unauthorized payment. Machine learning algorithms like logistic regression, decision trees, and ensemble methods proved better than the older rule-based systems, providing increased accuracy, precision, and recall in payment fraud detection.

One of the key benefits of the models remains the ability of the system to function in real-time, thus enabling prompt recognition of suspicious payment attempts and mitigating loss. The system is also very scalable, supporting high data volumes, yet appropriate for use in busy e-commerce sites and financial sites. Also, the system can be continuously retrained using new data, thus remaining effective in combating increasingly evolving payment crime methods. Overall, the deployment of machine learning has not only fortified the payment system against fraudulent attack, while also improving the final user experience by promoting safer and more reliable online payments. The project indicates the potential of AI in transforming payment process security.

REFERENCES

1. **Y. Abakarim, M. Lahby, and A. Attioui,** "An efficient real-time model for credit card fraud detection based on deep learning," in Proc. 12th Int. Conf. Intelligent Systems: Theories and Applications (SITA), Oct. 2018, pp. 1–7.
2. **H. Abdi and L. J. Williams,** "Principal component analysis,"Wiley Interdisciplinary Reviews: Computational Statistics, vol. 2, no. 4, pp. 433–459, Jul. 2010.
3. **V. Arora, R. S. Leekha, K. Lee, and A. Kataria,** "Facilitating user authorization from imbalanced data logs of credit cards using artificial intelligence," Mobile Information Systems, vol. 2020, pp. 1– 13, Oct. 2020.
4. **A. O. Balogun, S. Basri, S. J. Abdulkadir, and A. S. Hashim,** "Performance analysis of feature selection methods in software defect prediction: A search method approach," Applied Sciences, vol. 9, no. 13, pp. 2764, Jul. 2019.
5. **B. Bandaranayake,** "Fraud and corruption control at education system level: A case study of the Victorian department of education and early childhood development in Australia," Journal of Cases in Educational Leadership, vol. 17, no. 4, pp. 34–53, Dec. 2014.
6. **J. Błaszczyński, A. T. de Almeida Filho, A. Matuszyk, M. Szelński, and R. Słowiński,** "Auto loan fraud detection using dominance-based rough set approach versus machine learning methods," Expert Systems with Applications, vol. 163, Jan. 2021.
7. **B. Branco, P. Abreu, A. S. Gomes, M. S. C. Almeida, J. T. Ascensão, and P. Bizarro,** "Interleaved sequence RNNs for fraud detection," in Proc. 26th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining, 2020, pp. 3101–3109.
8. **F. Cartella, O. Anunciacao, Y. Funabiki, D. Yamaguchi, T. Akishita, and O. Elshocht,** "Adversarial attacks for tabular data: Application to fraud detection and imbalanced data," arXiv:2101.08030, 2021.
9. **S. S. Lad and A. C. Adamuthe,** "Malware classification with improved convolutional neural network model," International Journal of Computer Network and Information Security, vol. 12, no. 6, pp. 30–43, Dec. 2021.