Online - Fraud Deduction

Abstract

Online financial transactions have become increasingly common, making fraud detection a critical challenge for financial institutions. This project focuses on developing a machine learning-based system to detect fraudulent transactions using a real-world dataset containing over 6.3 million records. The dataset exhibits a significant class imbalance, with fraudulent cases comprising less than 0.2% of the total transactions.

To address this, extensive data preprocessing was conducted, including normalization, categorical encoding, and feature selection. Multiple classification algorithms were implemented and compared, including Logistic Regression, Decision Tree, Random Forest, XGBoost, SVM, K-Nearest Neighbors, AdaBoost, Gradient Boosting, and Naive Bayes. The models were evaluated using accuracy, precision, recall, and F1-score, with a particular focus on the minority fraud class.

Top Performing Algorithms (Based on Precision, Recall, and F1-score for Fraud Class — isFraud = 1)

Algorithm	Accuracy	Precision (fraud)	Recall (fraud)	F1-score (fraud)
Random Forest	99.97%	0.98	0.79	0.87
XGBoost	99.96%	0.89	0.80	0.84
Gradient Boosting	99.95%	0.93	0.69	0.79
Decision Tree	99.97%	0.90	0.87	0.88
SVM	99.93%	1.00	0.43	0.60
KNN	99.91%	0.97	0.37	0.54
Logistic Regression	99.98%	0.84	0.09	0.16
Naive Bayes	99.97%	0.00	0.00	0.00
AdaBoost	99.98%	1.00	0.11	0.20

About Data set

The online fraud detection dataset consists of over 6.3 million transaction records, simulating the behavior of users performing financial transactions within an online banking system. Each record captures a single transaction and includes various attributes that describe both the transaction itself and the financial state of the involved accounts before and after the transaction. These attributes include the type of transaction (e.g., PAYMENT, TRANSFER, CASH_OUT), the transaction amount, and account identifiers for the sender (nameOrig) and receiver (nameDest). Additionally, it provides the original and new balances of both accounts involved (oldbalanceOrg, newbalanceOrig, oldbalanceDest, newbalanceDest).

A key characteristic of the dataset is its high class imbalance—fraudulent transactions (isFraud = 1) account for a very small percentage of the total transactions, making fraud detection particularly challenging. The dataset also includes a binary field called isFlaggedFraud, which indicates whether a transaction was flagged as suspicious by an automated rule-based system. However, this field is rarely active and does not play a central role in model training.

The dataset is complete and free of missing values, and its structure makes it well-suited for supervised machine learning tasks. It reflects real-world complexities, such as transactions with zero balances, large money transfers, and patterns in fraud that tend to occur more frequently in certain transaction types like TRANSFER and CASH_OUT. These nuances make the dataset ideal for building and evaluating fraud detection models that prioritize precision and recall over simple accuracy due to the imbalanced nature of the target variable.

Conclusion

In this project, we successfully built and evaluated multiple machine learning models to detect fraudulent transactions in an online financial dataset containing over 6 million records. Given the highly imbalanced nature of the dataset—where fraudulent transactions represent a small fraction of the total—we focused on optimizing not just overall accuracy, but also **precision, recall, and F1-score for the fraud class**, which are critical for real-world fraud detection systems.

We implemented and compared a variety of algorithms including Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, XGBoost, AdaBoost, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Naive Bayes. Among all these, the **Random Forest Classifier** consistently delivered the best performance with:

- **Accuracy**: 99.97%
- **Precision (Fraud)**: 0.98
- **Recall (Fraud)**: 0.79
- **F1-score (Fraud)**: 0.87

These results demonstrate that Random Forest not only generalizes well across unseen data but also maintains a strong balance between minimizing false negatives (missed frauds) and false positives (incorrect fraud alerts).

The project also highlights the importance of:

- **Data preprocessing** (normalization, encoding),
- **Class imbalance handling**, and
- **Model evaluation using appropriate metrics beyond accuracy**, especially in critical domains like fraud detection.