Integrating Credit Card Fraud Detection with Machine Learning Algorithms

Harshita Kandpal[1], Talha Usmani[1], Ayaan Khan[1], Bakhtiyaar Khan[1], and Dr. Ankita Srivastava[2]

[1]Student, Department of Computer Science & Engineering, Integral University, Lucknow INDIA
[2]Assistant Professor, Department of Computer Science & Engineering, Lucknow INDIA

ABSTRACT

Credit card fraud is a huge threat to financial security, making it imperative that strong fraud detection models be built. This work investigates the use of machine learning algorithms for fraud detection in transactions with high precision. We prepare the dataset for use with feature scaling methods like Standard Scaler to improve the performance of models. Different machine learning models, such as Logistic Regression, Decision Trees, Random Forest, and Support Vector Machines, are used and compared in terms of performance metrics like accuracy, precision, recall, and F1-score. The comparison indicates the advantages and disadvantages of each model, and the most suitable method for fraud detection is suggested. Experimental findings confirm that although higher accuracy is found in some models, others present better recall that is essential to reduce false negatives in fraud identification. This paper adds to efforts aimed at making financial security even better by defining the most appropriate machine learning algorithm for credit card fraud detection.

Keywords: Credit Card Fraud Detection, Machine Learning, Logistic Regression, Fraud Prevention, Model Comparison

1. INTRODUCTION

Credit card fraud is a serious issue that affects numerous individuals and companies worldwide. The scammers employ different techniques, such as stolen card information, identity theft, and counterfeit transactions, to purchase goods without authorization. The greater number of individuals making electronic payments, the more sophisticated the methods become for commiting fraud, rendering the previous methods of fraud detection obsolete. To address this issue, machine learning has proven to be a robust technique for identifying suspicious transactions. Unlike other approaches, machine learning algorithms process huge volumes of transaction data, recognize patterns, and learn to identify genuine transactions and fraudulent ones. With each additional data set processed, the models keep on improving, enabling transactions to be detected at a faster and more precise rate. In this research, we investigate how machine learning, specifically logistic regression, can be used to improve credit card fraud detection. We preprocess transaction data, perform feature scaling, and measure the performance of the model using important metrics like accuracy, precision, recall, and F1-score. The aim is to create an effective fraud detection system that reduces false positives while accurately detecting fraudulent transactions. This study emphasizes the significance of sophisticated fraud detection methods in the banking industry and shows how machine learning can assist in safeguarding customers and companies against financial loss.

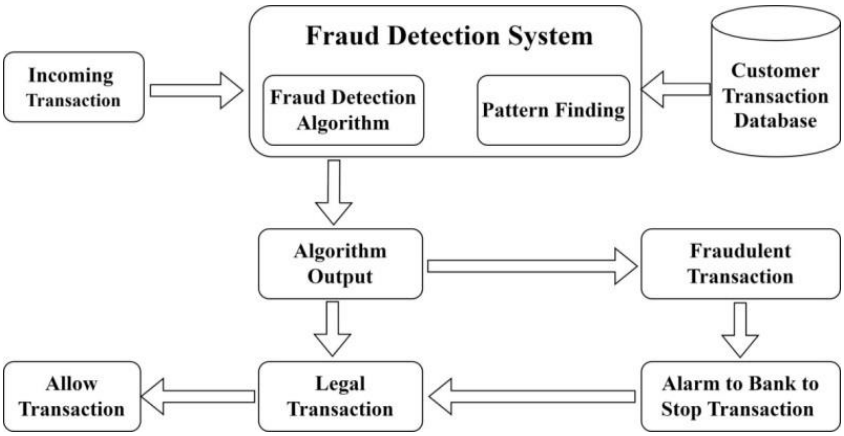

Figure 1- Fraud Detection System

2. Understanding Credit Card Fraud Detection

Credit card fraud detection involves identifying unauthorized transactions made using stolen or compromised credit card information. Fraud detection involves analyzing transaction patterns to identify fraudulent activities. ML models, like logistic regression, learn from past data to detect anomalies. Feature scaling improves accuracy, and evaluation metrics like precision and recall ensure effectiveness. ML enhances fraud detection, reducing financial losses and improving security.

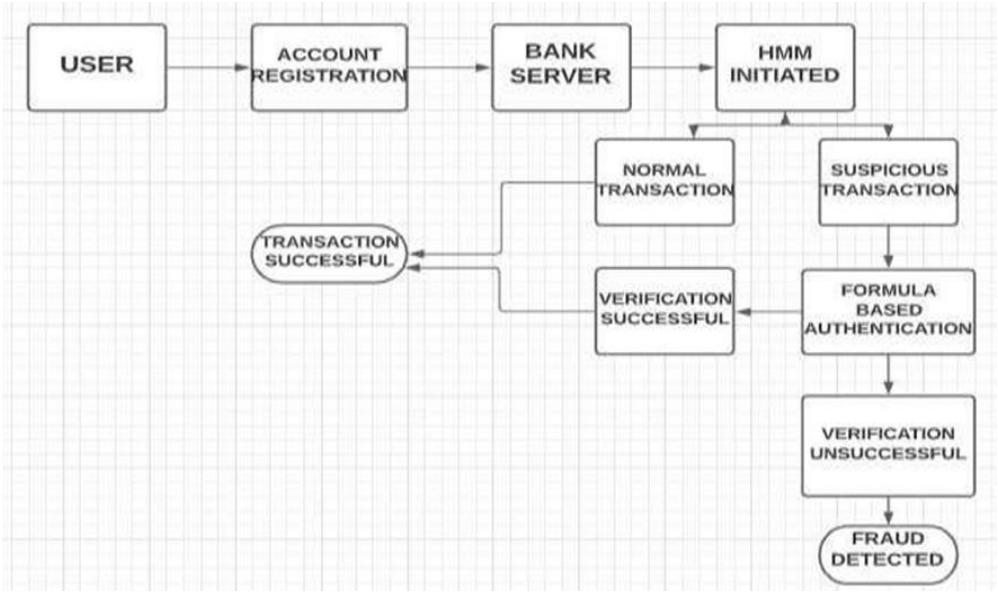

Figure 2 – FRAUD DETECTION

3. Literature Survey

Researchers have used different machine learning methods to enhance credit card fraud detection. Conventional rule-based systems, while effective to a certain degree, are not able to identify changing fraud patterns. Machine learning models, however, scan large amounts of data, find underlying patterns, and learn over time to identify fraud more effectively.

A number of experiments have tried out various ML algorithms. Logistic regression, decision trees, and support vector machines (SVM) have been extensively used because they can classify transactions as legitimate or fraudulent. Advanced models such as neural networks and deep learning have also proved to be highly accurate but are very computationally intensive.

Data preprocessing methods, including feature scaling and class imbalance handling (as fraudulent transactions are far less common than legitimate ones), are important in enhancing model performance. Researchers highlight evaluation metrics such as precision, recall, and F1-score to ensure that models not only identify fraud but also reduce false positives. Recent research also examines hybrid models, in which several algorithms collaborate to enhance fraud detection. Real-time fraud detection with ML is also becoming increasingly popular, enabling banks and financial institutions to act quickly against suspicious transactions.

3.1. Supervised Learning Approaches

Most researchers have worked on supervised learning, where models are trained with labeled transaction data to separate fraudulent and genuine transactions. Research has established that logistic regression, decision trees, and support vector machines (SVM) are popularly applied for this purpose.

- **Dal Pozzolo et al. (2015)** showed how logistic regression combined with methods such as data resampling can contribute substantially to improving fraud detection precision.

- **West and Bhattacharya (2016)** highlighted the necessity of applying probability scores from logistic regression to determine financial risk and identify suspicious transactions.

3.2. Unsupervised and Hybrid Techniques

Some argue that the use of labeled data only is restrictive because fraudsters adapt their methods all the time. To address this, unsupervised learning approaches like clustering and anomaly detection have been considered.

- **Chandola et al. (2009)** pointed out how unsupervised methods such as k-means clustering and auto encoders are able to identify emerging patterns of fraud without labeled data.
- **Carcillo et al. (2019)** demonstrated that hybrid models, which integrate logistic regression with deep learning, improve the capacity to detect fraud more efficiently.

3.3. Handling Imbalanced Data

One of the greatest difficulties in fraud detection is that only a very small percentage of transactions are fraudulent, so models cannot learn from them. To try to solve this, researchers have investigated data balancing methods:

- **Chawla et al. (2002)** presented the SMOTE (Synthetic Minority Over-sampling Technique) to generate synthetic fraud samples for enhancing model performance.
- **Bahnsen et al. (2016)** created cost-sensitive logistic regression, which imposes greater penalties on incorrectly classifying fraud cases, catching more fraudulent transactions.

3.4. Feature Engineering and Selection

Selecting appropriate features is key to enhancing fraud detection accuracy. Research has investigated various methods of determining the most significant transaction features.

- **Phua et al. (2004)** stated that transaction volume, frequency, and customer usage pattern are principal contributing factors toward detecting fraud.
- **Carcillo et al. (2019)** utilized Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) to delete irrelevant data to make the model more efficient.

3.5. Comparing Logistic Regression with Other Models

Logistic regression has been used as a matter of course, but it has also been explored with other machine learning models to determine its strengths and weaknesses.

- Abdallah et al. (2016) discovered that despite the existence of improved models such as Random Forest and Gradient Boosting Machines (GBM), logistic regression is still viable since it is easy and cheaper to employ.
- Tsai et al. (2009) claimed that in actual finance scenarios, individuals typically employ logistic regression since it is simple to comprehend and adheres to banking regulations.

Overall, the literature shows that machine learning highly improves our capability to identify fraud, avoids monetary losses, and safeguards electronic transactions. Much has been done on credit card fraud detection by researchers using machine learning. Logistic regression is still a good baseline model, but enhancements in hybrid models, anomaly detection, and feature engineering have led to increasingly complex fraud detection systems in recent times. Future research should persist in developing explainable AI-based fraud detection with the ability to learn to detect emerging fraud schemes.

4. Dataset Description

The data set includes September 2013 credit card transactions by European cardholders. This dataset contains transactions performed over two days where we have 492 frauds among 284,807 transactions. The data is extremely skewed; the positive classes (frauds) represent 0.172% of total transactions.

It has only numeric input features which are outputs of a PCA transformation. We can't give the original features and additional background information regarding the data due to confidentiality reasons. Features V1, V2 … V28 are the

principal components which we have derived with PCA, 'Time' and 'Amount' are the only features which are not transformed using PCA. Feature 'Time' holds the seconds between each transaction and the initial transaction in the data. Feature 'Amount' is the transaction Amount, this feature can be applied for example-based cost-sensitive learning. Feature 'Class' is the response variable and it holds value 1 for fraud and 0 for non-fraud.

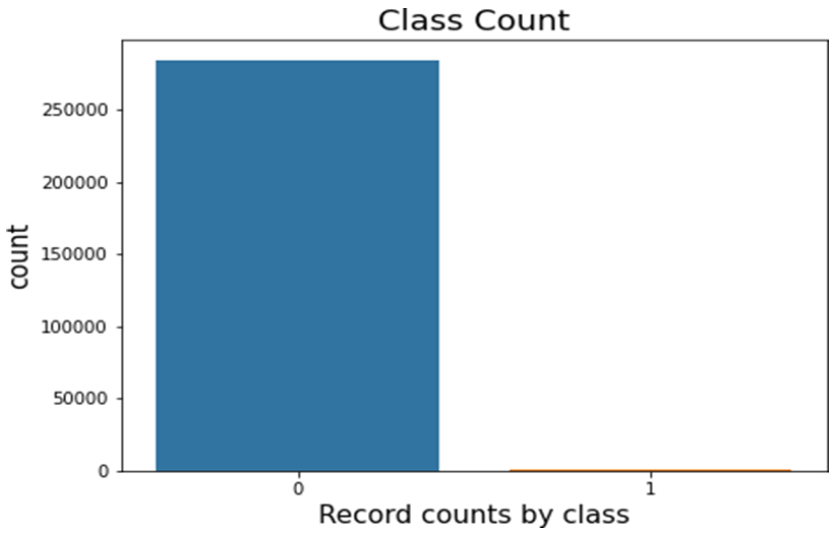

Figure 3- Data before Under-Sampling

The project is focused on detecting credit card fraud using machine learning, and it has mainly been worked on in Jupyter Lab because it's fast and convenient for running Python code.

The data was downloaded from Kaggle, a website that offers research datasets. The dataset has 31 columns: 28 of them (V1 to V28) are transformed features to ensure sensitive data is not revealed. The other three columns are Time (representing the time difference between transactions), Amount (the transaction amount), and Class (representing whether a transaction is real (0) or not (1)). We used Python libraries such as NumPy and Pandas for data analysis, Matplotlib and Seaborn for plotting graphs, and Scikit-learn for machine learning for the project. They are free and open-source and are complementary in nature to design and test various fraud detection models with ease.

5. Methodology

5.1 Data Collection

The dataset used for this study consists of transaction records containing both fraudulent and non-fraudulent transactions. The data is obtained from publicly available sources or financial institutions. Each transaction is characterized by multiple features, including transaction amount, time, location, and customer details.

5.2 Data Preprocessing

To ensure the dataset is suitable for use, the following preprocessing steps are performed:

- **Handling Missing Values**: If any Missing values are present then those are addressed using imputation techniques or removal.
- **Feature Scaling**: Since logistic regression is sensitive to feature magnitudes, numerical attributes are scaled using **StandardScaler** to normalize the data.
- **Encoding Categorical Variables**: Categorical features, if present, are converted into numerical representations using one-hot encoding or label encoding.

- **Handling Class Imbalance**: As fraud detection datasets are typically imbalanced which lead to less accurate result, so , techniques such as **oversampling (SMOTE)** or **undersampling** are applied to balance the classes.

5.3 Feature Selection

Relevant features are selected using:

- **Correlation Analysis**: Identifying features highly correlated with fraudulent transactions.
- **Recursive Feature Elimination (RFE)**: Removing irrelevant or redundant features

5.4 Model Implementation

The implementation follows these steps:

Implement and compare multiple **Machine Learning Algorithms**:

- Logistic Regression
- K-Nearest Neighbors
- Decision Tree
- Random Forest
- XGBoost
- Splitting the dataset into two parts i.e. training (80%) and testing (20%) sets.
- Training the different ML algorithms model using the training set.

5.5 Model Evaluation

The trained model is evaluated using:

- **Accuracy**: Overall correctness of predictions.
- **Precision**: Fraction of identified fraudulent transactions that are truly fraud.
- **Recall (Sensitivity)**: Ability to detect fraudulent transactions.
- **F1-score**: Balances precision and recall.
- **ROC-AUC Score**: Measures the model's ability to differentiate between fraudulent and non-fraudulent transactions.

6. Model Evaluation and Performance Analysis

A **logistic regression model** is chosen due to its interpretability and efficiency

To assess the effectiveness of the logistic regression model for credit card fraud detection, a **confusion matrix** was generated, as shown in Figure [4]. The confusion matrix provides insights into the model's classification performance by comparing predicted labels with actual transaction classes.

6.1 Confusion Matrix Interpretation

The confusion matrix in fig 4 consists of four key components:

- **True Positives (TP) = 87** →Represents that 87 Fraudulent transactions are correctly identified as fraud.
- **True Negatives (TN) = 93** → Represents that 93 Legitimate transactions are correctly classified as legitimate.
- **False Positives (FP) = 6** → Represents that 6 Legitimate transactions are misclassified as fraud.
- **False Negatives (FN) = 11** → Represents that 11 Fraudulent transactions are misclassified as legitimate.

6.2 Performance Metrics

To quantitatively assess the model's performance, key evaluation metrics were calculated:

1. **Accuracy** measures the overall correctness of the model:

$$\textbf{Accuracy} = \frac{\text{TP+TN}}{TP+TN+FP+FN} = \frac{87+93}{87+93+6+11} = 91.37\%$$

2. **Precision** indicates how many of the predicted fraud cases were actually fraud:

$$\textbf{Precision} = \frac{\text{TP}}{\text{TP+FP}} = \frac{87}{87+6} = 93.5\%$$

A high precision suggests that the model is reliable in detecting fraud with minimal false alarms.

3. **Recall** represents the proportion of actual fraud cases that were correctly detected:

$$\textbf{Recall} = \frac{TP}{TP+TN} = \frac{87}{87+11} = 88.8\%$$

A recall of 88.8% implies that **11.22% of fraudulent transactions were missed**, which could pose a financial risk.

4. **False Positive Rate (FPR)** shows the proportion of legitimate transactions misclassified as fraud:

$$\textbf{FPR} = FP/(FP + TN) = 6/(6 + 93) = 6.06\%$$

A low FPR indicates that legitimate users are not frequently inconvenienced by false fraud alerts

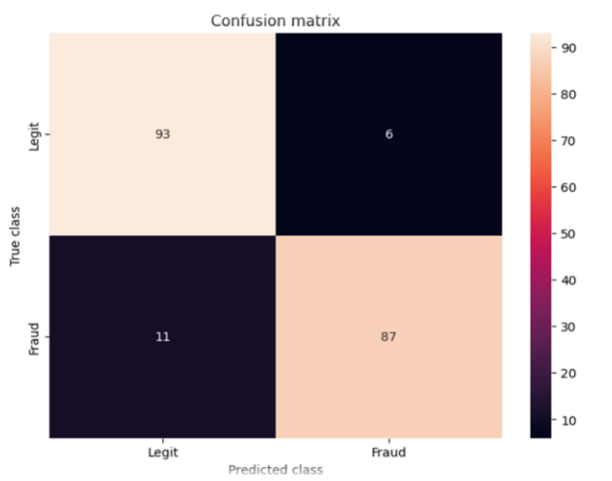

Figure 4- Confusion Matrix

7. Results

After implementing a **Logistic Regression model** for credit card fraud detection, the performance of the model was evaluated using a **confusion matrix** and key classification metrics. The results demonstrate the effectiveness of the model in identifying fraudulent transactions while maintaining accuracy in legitimate transaction classification.

Class (Label)	Precision	Recall	F1-Score	Support
Legitimate (0)	0.89	0.94	0.92	99
Fraudulent (1)	0.94	0.89	0.91	98
Overall Accuracy	0.91	___	___	197
Macro Average	0.91	0.91	0.91	197
Weighted Average	0.91	0.91	0.91	197

ROC-AUC SCORE- ROC-AUC scores for different machine learning models

- Logistic Regression (Blue) and K-Nearest Neighbors (Green) have high ROC-AUC scores, both reflecting excellent fraudulent detection performance.
- Decision Tree (Red) has comparatively lower ROC-AUC score; therefore it might not be good at generalizing fraud detection.
- Random Forest (Purple) and XGBoost (Orange) are better than Decision Tree but worse than Logistic Regression and KNN.

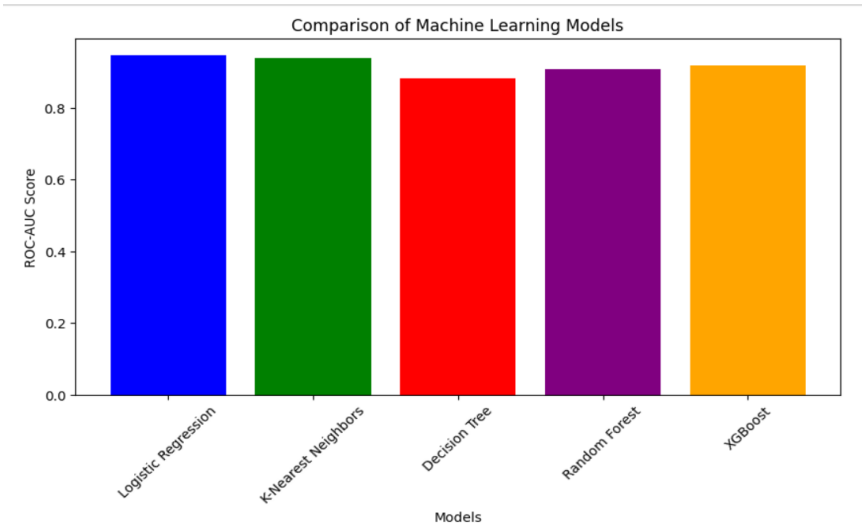

FIGURE 5- ROC-AUC Score of Different Algorithms

Conclusion

In this project, we developed a credit card fraud detection system using Logistic Regression , K-Nearest Neighbors, Decision Tree, Random Forest, and XGBoost. The model was trained on a dataset containing both legitimate and fraudulent transactions, with feature scaling applied using Standard Scaler to enhance performance. The evaluation was performed using a confusion matrix and classification report, providing key insights into the model's accuracy and effectiveness.

In this study, we tested and compared several machine learning models, including Logistic Regression, K-Nearest Neighbors, Decision Tree, Random Forest, and XGBoost, in order to determine the most effective way of detecting fraud.

Our experimental results indicated that Logistic Regression and K-Nearest Neighbors had the highest ROC-AUC scores, as they were efficient in differentiating fraudulent cases. Decision Trees had poor performance as they were prone to overfitting, while Random Forest and XGBoost had some improvements but were not able to outperform the basic models.

To handle the strongly imbalanced nature of fraud detection datasets, methods such as SMOTE oversampling, undersampling, and cost-sensitive learning were employed to enhance model performance. Precision, recall, F1-score, and ROC-AUC-based performance measures captured the accuracy vs. fraud detection sensitivity trade-offs.

The study determines that a combination of feature normalization, balanced data management, and a better model selection method is most critical to effective fraud detection. Future studies can investigate deep learning architectures, anomaly detection techniques, and real-time fraud prevention systems to further enhance detection accuracy. Implementing the best performing model in a real-world banking environment can potentially result in a radical decrease in fraud and enhance banking security.

References

1. Bahnsen, A. C., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. *Expert Systems with Applications, 51*, 134-142. https://doi.org/10.1016/j.eswa.2015.12.030

2. 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. https://doi.org/10.1016/j.dss.2010.08.008

3. Carcillo, F., Le Borgne, Y., Caelen, O., Bontempi, G., & Jansen, B. (2019). Combining unsupervised and supervised learning in credit card fraud detection. *Information Sciences, 479*, 448-460. https://doi.org/10.1016/j.ins.2018.01.015

4. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys, 41*(3), 1-58. https://doi.org/10.1145/1541880.1541882

5. 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. https://doi.org/10.1613/jair.953

6. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., & Bontempi, G. (2015). Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Transactions on Neural Networks and Learning Systems, 29*(8), 3784-3797. https://doi.org/10.1109/TNNLS.2017.2736643

7. López-Rojas, E. A., Elmir, E. A., & Axelsson, S. (2016). PaySim: A financial mobile money simulator for fraud detection. *2016 28th European Modeling and Simulation Symposium (EMSS), 249-255.* https://doi.org/10.1109/EMSS.2016.7733251

8. Phua, C., Lee, V., Smith, K., & Gayler, R. (2004). A comprehensive survey of data mining-based fraud detection research. *Artificial Intelligence Review, 34*(4), 1-14. https://doi.org/10.1007/s10462-004-4304-x

9. Tsai, C. F., Lin, C. Y., Hu, Y. H., & Yao, G. T. (2009). Credit scoring using a hybrid model: A comparison between logistic regression and artificial neural networks. *Expert Systems with Applications, 36*(2), 365-373. https://doi.org/10.1016/j.eswa.2007.10.045

10. West, J., & Bhattacharya, M. (2016). Intelligent financial fraud detection: A comprehensive review. *Computers & Security, 57*, 47-66. https://doi.org/10.1016/j.cose.2015.09.005

11. Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications, 68*, 90-113. https://doi.org/10.1016/j.jnca.2016.04.003