Machine Learning for Credit Card Fraud Detection: A Comparative Study of Algorithms

MSc Research Project
MSc in Cyber Security

Rahul Raji
Student ID: X23216662

School of Computing
National College of Ireland

Supervisor: Michael Prior

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Rahul Raji		
Student ID:	X23216662		
Programme:	MSc in Cyber Security	**Year:**	2024
Module:	Practicum Part 2		
Supervisor:	Michael Prior		
Submission Due Date:	12/12/24		
Project Title:	Machine Learning for Credit Card Fraud Detection: A Comparative Study of Algorithms		
Word Count:	6655	**Page Count:**	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Rahul Raji

Date: 12/12/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Machine Learning for Credit Card Fraud Detection: A Comparative Study of Algorithms

Rahul Raji
x23216662

Abstract

This paper applies machine learning algorithms for the detection of credit card fraud by focusing on the identification of fraudulent transactions and their reduction. The imbalanced dataset was used with oversampling SMOTE and feature scaling techniques in order to improve the model's performance. Supervised models such as Random Forest, Support Vector Machines (SVM), and Logistic Regression were also evaluated for accuracy, precision, recall, and F1-score. In addition, the unsupervised methods like Isolation Forest were tested for anomaly detection. It is observed that Random Forest outperformed others by having a higher accuracy and feature importance analysis, while SVM showed excellent precision for binary classification. The findings thus emphasize the comparative approach to improving fraud detection systems.

1 Introduction

Credit card fraud remains one of the major challenges that the financial industry is still facing today. The fact is that fraudsters have advanced their techniques in taking advantage of weaknesses in payment systems to commit their fraudulent activities. Huge financial losses and damaging consumer trust characterize the increased frequency and complexity of fraudulent activities accompanying the rapid growth of online transactions. This implies that real-time fraud detection can be treated as one class problem among many classes of legitimate transactions for distinguishing between fraudulent and nonfraudulent patterns. The challenge lies in the fact that most fraud datasets are imbalanced, and genuine transactions outweigh the fraudulent ones. In such contexts, machine learning-based solutions have proven effective against fraud patterns. Algorithms capable of complex behavior learning and anomaly detection can outline those patterns. This dissertation addresses the applicability of machine learning approaches in credit card fraud detection and specifically focuses on issues associated with imbalanced data. To counter this issue, this dissertation applies augmentation techniques like SMOTE, GAN, and VAE to synthetically produce fraud cases. The performances of the aforementioned machine learning models - Logistic Regression, Support Vector Machines (SVM), Decision Trees, Random Forests, and Neural Networks, on these datasets, enhanced with the above techniques are compared and contrasted in this paper. In light of evaluation based on accuracy, precision, recall, F1-score, and AUC-ROC metrics, their effectiveness could be evaluated as applied to fraud detection. Through this, the dissertation aims to come up with the best combination of models and data augmentation methods to enhance the accuracy of fraud detection in the fight against credit card fraud for financial institutions.

1.1 Aim

This dissertation aims to improve fraud detection in credit cards by measuring the performance of different machine-learning algorithms on augmented datasets.

This study evaluates SMOTE, GAN, and VAE in class-imbalanced problem resolution and the achievement of accuracy enhancement in the models. In doing this, it aims to find out which of these models will perform best, either the Logistic Regression, SVM, Decision Trees, Random Forests, or Neural Networks while using a comparison of several key performance metrics for different models with augmentation techniques applied for fraud detection, hence developing a safer and more effective transaction systems.

1.2 Objectives

Objectives of the paper are directed towards data augmentation and machine learning for credit card fraud detection improvement. This paper tries to compare SMOTE, GAN, and VAE methods for data augmentation techniques and comparison along with suitability assessment of diverse algorithms on augmented data to find an optimized approach that will work efficiently to increase the precision as well as reliability within the fraud detection models.

- ➢ To compare the data augmentation techniques: SMOTE, GAN, and VAE for handling the imbalanced class problem in fraud detection.
- ➢ To evaluate the performance of Logistic Regression, SVM, Decision Trees, Random Forests, and Neural Networks when trained on augmented datasets.
- ➢ To assess model performance using accuracy, precision, recall, F1-score, and AUC-ROC for a comprehensive comparison.

1.3 Research Questions

- Which machine learning algorithms are most effective in detecting credit card fraud, and how do their performances compare under varying transaction volumes and fraud rates?
- What happens in terms of the performance of different algorithms in relation to the number of transactions?
- In what way do these algorithms’ performances vary with the fraud rate?
- Comparing decision trees, random forests, support vector machines, neural networks and ensemble methods, which of them has their advantages and/or disadvantages in credit card fraud detection?
- Which of the performance evaluation methods gives the best measure of algorithm performance in fraud detection?
- Which recommendations can be made for financial institutions based on the comparative analysis of these algorithms?

2 Related Work

Thus, credit card fraud detection is a promising direction for financial institutions globally with the development of the digital economy. As online methods of payment and use of credit cards become the order of the day, credit card fraud incidences are on the increase, and this has led to a lot of loss and a reduction of confidence in the use of online payments. Fraud detection systems are designed to detect and prevent underwritten frauds in real as to safeguard consumers and financial organizations (Sailusha *et al*. 2020). Rule-based systems in fraud management derive from fixed parameters or scores that require human intervention drastically fail to adapt to the changing forms of fraud and have auspicious false positive ratios.

Machine learning has recently been identified as a useful solution in battling credit card fraud. Nevertheless, credit card fraud detection comes with some challenges especially because assessing the performance of the fraud detection model requires testing on transaction datasets, which not only have few fraud cases relative to genuine transactions. This often leads to the development of a model that is more inclined towards the majority class and therefore detecting fraud becomes a challenging affair (Alfaiz and Fati, 2022). To overcome these challenges and enhance the sensitivity and precision of together with existing fraud detection systems, new data augmentation techniques including SMOTE and GAN are currently being studied.

2.1 Class Imbalance Challenge in fraud Detection

Class imbalance is the main problem of credit card fraud detection as the number of legitimate transactions is significantly higher than the number of attempts at fraud. This extreme skewness of distribution means that very little of the data presents the profile of fraudulent transactions, and this results in models that are highly inclined towards estimating non-fraudulent transactions (Tiwari *et al*. 2021). As a result, most classical machine learning algorithms learn from the majority class, which comprises legitimate transactions, keeping the minority class or the fraudulent transactions out of their sight and possibly costing companies millions of dollars. Class imbalance is more severe in fraud detection, as it leads to false negatives – missed fraud cases – being considerably more costly than false positives, or legitimate transactions erroneously flagged as fraud.

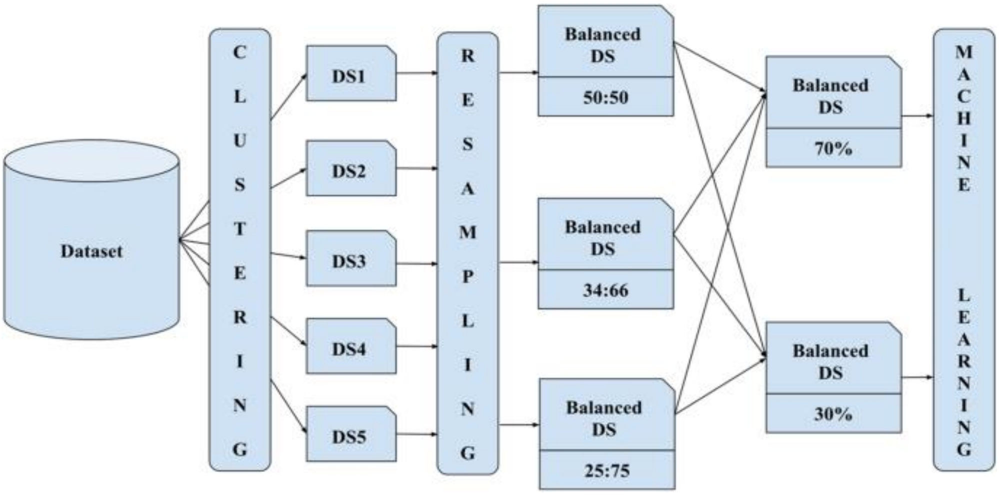


Figure 1: The perspective credit card fraud (source: Dang et al.2021)

The most important key area of fraud detection models can be considered as potential risks associated with failure in fraud detection that directly brings customer financial tasks and inconvenience. To overcome this problem, SMOTE and other advanced generative methods such as Generative Adversarial Networks are used (Dang *et al*. 2021). While SMOTE builds synthetic cases from the minority class by finding the midpoint between existing cases of fraud, GANs, and VAEs mimic the real fraud cases and create new realistic samples. They increase the size of the minority class and help the model to be more sensitive to the cases of fraud and thus it could handle large imbalanced data sets (Lucas and Jurgovsky, 2020). Exploring different techniques shows that different methods also affect model performance and, by doing a crosscheck on different machine learning models, it is possible to identify the most suitable method for fraud detection.

2.2 Traditional Fraud Detection Methods

The conventional approaches to fraud have involved the use of rules and checking carried out by experts. Premises of rule-based systems, the mainstay of conventional fraud detection, allow the running of transactional data against a set of predetermined criteria or "rules". For instance, a rule could be based on the amount of the transaction, which would signal where there are over a certain dollar quantity, or geographical location, which would alert to transactions from certain regions of the world (Jovanovic *et al*. 2022). Although containing a certain effectiveness, the rule-based models have some drawbacks – they are rigid and cannot learn new fraud patterns on the fly, which can lead both to false positives and to false negatives. For one, fraudsters are bound to change their ways, in this case, looking for ways around such set rules over time reducing the efficiency of such systems.

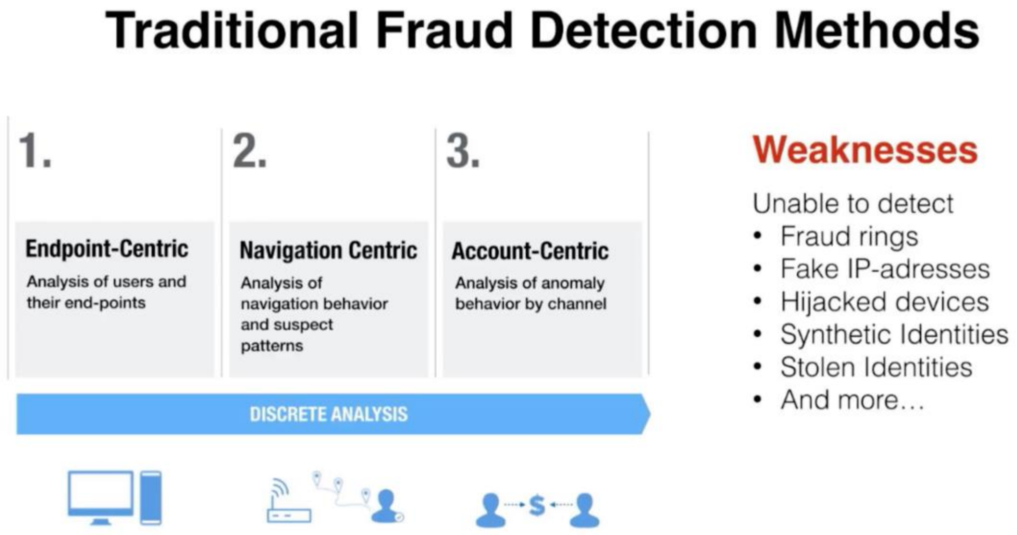


Figure 2: The traditional fraud detection method (Source: Roseline et al., 2022)

Another traditional approach is anomaly detection which tries to discover transactions that differ significantly from a user's spending behavior. Even though anomaly detection is more capable of providing a flexible model to the static rules it is a drawback that lacks the accuracy required to differentiate between a genuine anomaly and a fraudulent incident, leading to too many false positives (Roseline *et al*. 2022). Another traditional approach to the analysis of the flagged transactions includes their manual review, which however is time-consuming and cannot handle large volumes of data efficiently, which hampers the capacity to detect fraud in real-time. Current conventional approaches are, therefore, largely ineffective for mapping the hierarchical and dynamic nature of fraud models, which are essential for modern fraud identification processes.

2.3 Data Augmentation Techniques

Increased volume of the required dataset along with the skills utilized in the data augmentation process positively affect the performance of machine learning models, especially in the credit card fraud detection area. Since the fraud data is diametrically of a different class nature, where the number of fraud transactions is infinitesimal as compared to the genuine ones, then the conventional data-gathering approaches might not be enough. Data augmentation is used to try and make the minority class larger because this type of data can be useful for training and can help with generalization (Khatri, Arora, and Agrawal, 2020). There is the Synthetic Minority Over-sampling Technique which entails the creation of new instances of the minority class, the fraudulent transactions in this case by setting instances midway between the existing ones.

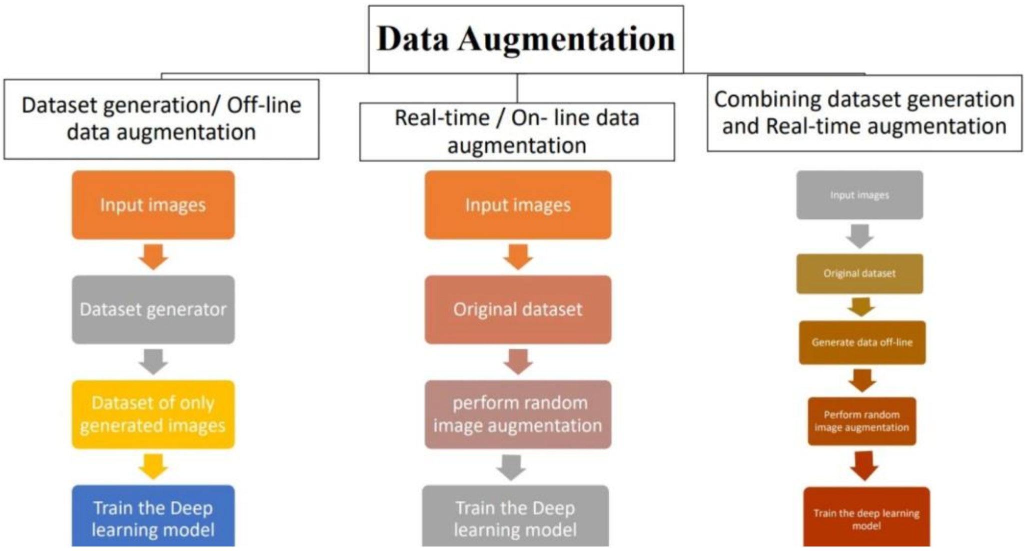

Data Augmentation
Dataset generation/ Off-line data augmentation
Real-time / On- line data augmentation
Combining dataset generation and Real-time augmentation
Input images
Dataset generator
Dataset of only generated images
Train the Deep learning model
Input images
Original dataset
perform random image augmentation
Train the Deep learning model
Input images
Original dataset
Generate data off-line
Perform random image augmentation
Train the deep learning model

Figure 3: The overview of data augmentation (Source: Saheed, Baba, and Raji, 2022)

SMOTE is a method of creating new samples of a density between the existing ones and therefore introduces new samples in feature space without replication. It minimizes overfitting and gives a better remedy to learn more expanded features of a fraudulent action. Other augmentation techniques include; data transformation methods such as rotation, scaling, and noise addition where existing transaction data are rotated, scaled, or noise added to them to generate new instances (Saheed, Baba, and Raji, 2022). Besides supporting the data source, these changes also help the model generalize to variations in real-world data and develop invariance to certain transformations. Recommending Generative Adversarial Networks (GANs) as a method of data augmentation, the author reminds readers that they are two neural networks pitted against each other that produce credible fake data. By applying the above approaches there is an overall improvement of the model and a general increase in the fraud detection system and thus a decrease in the financial losses that are caused by fraudsters.

2.4 Comparison of Machine Learning Algorithms in Fraud Detection

The performance of credit card fraud detection systems is fundamentally determined by the selection of machine learning algorithms. The following algorithms have been used in this domain and all of them have their advantages and disadvantages. Comparing these algorithms allows understanding of the differences in their effectiveness in solving the task, as well as the degree of model interpretability and the extent to which they can be used to solve the problems of fraud detection. Logistic Regression is one of the simplest machine-learning algorithms designed for binary classification [17]. Decision Trees enjoy their interpretability and are ideal for areas where determining decision-maker processes is vital.

Decision Trees can model nonlinear relationships between features, but the model tends to overfit, particularly when complex. Random Forests, an ensemble technique, is the extension of a single decision tree to build the number of them to improve the precision and the compliance degree. Support Vector Machines (SVM) are widely used in fraud detection, particularly in high-dimensional spaces; however, they have the problem of parameter setting and may be time-consuming for large-scale data [18]. Several works discussed in the previous section used Gradient Boosting Machines (GBM) and their modifications such as XGBoost, which shows very high performance but can be sensitive to hyperparameters and not as easy to explain as simpler models. Neural Networks are other deep learning techniques that have become popular in the aspect of building complex dependencies on massive datasets.

2.5 Evaluation Metrics

Thus, for regular machine learning approaches used in credit card fraud detection, and due to the inevitable problem of datasets containing fewer instances of fraud, data augmentation methodologies are critical. The approach which is known as Synthetic Minority Over-sampling Technique (SMOTE) creates new instances for the minority class (fraudulent transactions) by using the interpolation technique on existing samples of the minority class instances [19]. Such an approach also assists in training a model with a more balanced distribution of the characteristics of the fraudulent actions. Further, transformation techniques like scaling, rotation, and noise addition can recreate variations of the actual transaction data, scenarios new but still containing fraudulent transaction data [20]. These changes make sure that the model compiles with an array of fraudulent patterns as a way of preventing overemphasizing and consequently gaining the ability to generalize. Other generative methods have also come up as strong candidates for data augmentation, especially the GANs. In GANs there are two major components: a generator and a discriminator which cooperate to create highly realistic synthetic data. Such techniques improve the dataset and make models stronger to detect fraud, which in turn results in efficient performance and less loss of money.

2.6 Impact of Data Augmentation on Model Performance

The application of diverse machine learning techniques in credit card fraud detection provided different solutions that are outlined below. As we saw when discussing the method, the logistic regression model is easy to interpret, hence ideal for use in simple problems. However, it may face difficulties in handling data with non-linear relationships that are characteristic of fraud patterns [20]. The Decision Trees allow greater freedom in decision-making and can capture nonlinear interactions. While they are various, they can bias a model heavily towards the training data and this can be regulated using Random Forests. Random Forests is the ensemble model made up of several Decision Trees to minimize the weirdness of over-fitted results.

SVMs are particularly suitable for high dimensional spaces, and use hyperplanes to define classes but suffer from a high degree of parameterization and might be slow. XGBoost and other GBM are the learned models that have remarkable performance that corrects errors in the prior model and works best in identifying perplexing patterns. Methods for deep learning particularly Neural Networks have received much attention due to the modeling of complex dependencies in big data [21]. But they require large amounts of data and computational power. Finally, the choice of the algorithm is a function of dataset features, interpretability of the solution, and computational resources for which prior understanding is crucial to maximize the desired fraud detection performance.

2.7 Literature Gap

Even though a wide literature has been done to identify credit card fraud with ML several research gaps are still evident. While most of the studies concentrate on routine machine learning models such as Logistic Regression, Decision Trees, etc. the possibilities of the latest models like GANs and VAEs for data augmentation have not been fully explored. In addition, there is limited literature concerning the comparison of these augmentation techniques as applied in improving the model performance in fraud detection. Unlike the present studies, no specific work provides comparative and detailed assessments of several algorithms under the same setting, resulting in limited procedure reference for assessment factors such as accuracy, precision, recall, F1-score, and AUC-ROC. Consequently, the factor of class imbalance and its impact on algorithm performance should be researched in more detail, especially taking into account realistic scenarios. If these gaps are filled, it will offer a richer perspective on enhancing fraud detection systems and enhance strategies to minimize fraud-related losses.

2.8 Summary

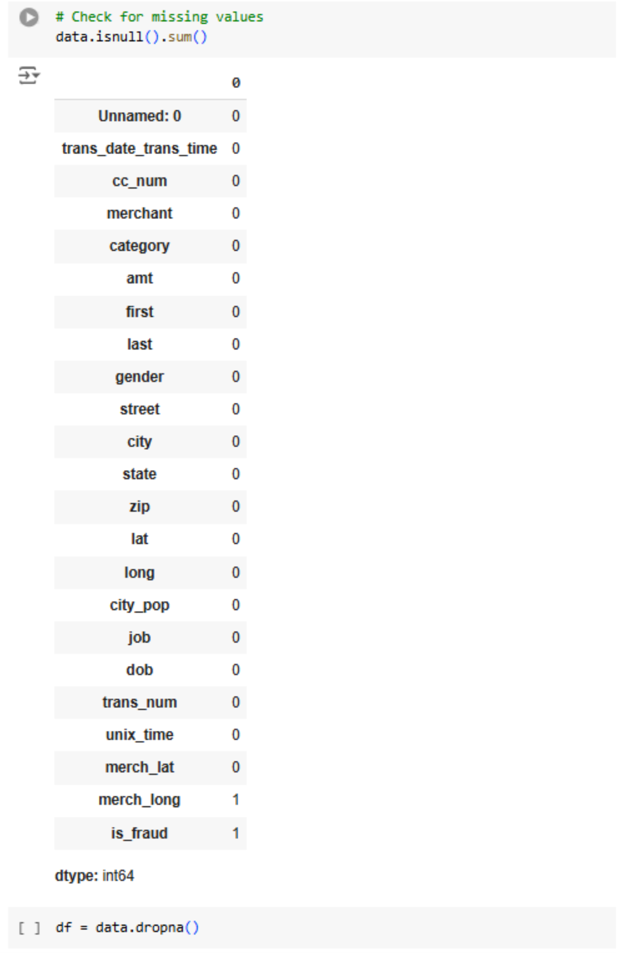
This study investigates a comparative analysis regarding ML algorithms and especially focuses on data enhancement methods including SMOTE, GAN, and VAE for credit card fraud detection. Due to high-class imbalance in most fraud detection datasets, these methods are designed to optimize the training process and therefore the tests on the performance measures. The study will involve a critical assessment of several algorithm models including Logistic Regression, Support Vector Machines, Decision Trees, Random Forests, and Neural Networks to determine the appropriateness of each model as a tool for detecting fraud in transactions. The study thus fills prominent literature gaps that concern the lack of enriched employment of specific augmentation techniques and the absence of universally established performance indicators. Thus, the research aims to have a differentiated analysis of these algorithms and augmentation strategies to contribute with practical recommendations for improvement in fraud detection, diminishing credit card loss rates, and improving the protection of credit card transactions.

3 Research Methodology

Credit card fraud detection refers to a distinction between genuine and fraudulent transactions, proving to be extremely challenging due to the highly imbalanced nature of fraud datasets. Rare, fraudulent transactions take up less than 0.5% of all transactions. It has been seen that it creates a lot of hassle for any kind of model in learning since most models lean toward the majority class comprising the legible ones- and the model fails to identify the minority class [1]. To counter this, data augmentation strategies are applied to strengthen the minority class towards a more balanced dataset to enhance the model's capability of identifying fraudulent transactions. This paper uses three data augmentation techniques to handle the problem of class imbalance in the dataset. The three techniques have been used as follows: First, an oversampling strategy called SMOTE (synthetic minority over-sampling technique). SMOTE generates synthetic samples to interpolate between existing minority-class instances. This method is simple computationally and widely applied in developing a balanced dataset [2]. Meanwhile, GANs involve deep learning-based generative models with a generator and discriminator. The generator would generate good-looking synthetic fraud transactions, which the discriminator assists in sharpening, and therefore they are promising generators of versatile and realistic data.

Another generative method through which latent representation of the fraud cases is learned is developed to generate new samples based upon this latent space and then creates synthetic data that introduce variations [3].

3.1 Data Preprocessing



```
# Check for missing values
data.isnull().sum()
```

```
                      0
Unnamed: 0            0
trans_date_trans_time 0
cc_num                0
merchant              0
category              0
amt                   0
first                 0
last                  0
gender                0
street                0
city                  0
state                 0
zip                   0
lat                   0
long                  0
city_pop              0
job                   0
dob                   0
trans_num             0
unix_time             0
merch_lat             0
merch_long            1
is_fraud              1

dtype: int64
```

```
df = data.dropna()
```

Figure 4: The data pre-processing
(Source: Self-created)

Table 1: Data Format

Column Name	Description	Example
step	Represents the time step of the transaction (e.g., in days).	1
type	Type of transaction (e.g., PAYMENT, TRANSFER, CASH_OUT, DEBIT).	PAYMENT
amount	Amount of money involved in the transaction.	9839.64
nameOrig	Unique identifier for the account originating the transaction.	C1231006815
oldbalanceOrg	Account balance of the originator before the transaction.	170136
newbalanceOrig	Account balance of the originator after the transaction.	160296.36
nameDest	Unique identifier for the account receiving the transaction.	M1979787155
oldbalanceDest	Account balance of the receiver before the transaction.	0
newbalanceDest	Account balance of the receiver after the transaction.	0
isFraud	Indicates whether the transaction is fraudulent (1 = fraud, 0 = not fraud).	0
isFlaggedFraud	Indicates if the transaction was flagged as fraud by the system (1 = flagged, 0 = not flagged).	0

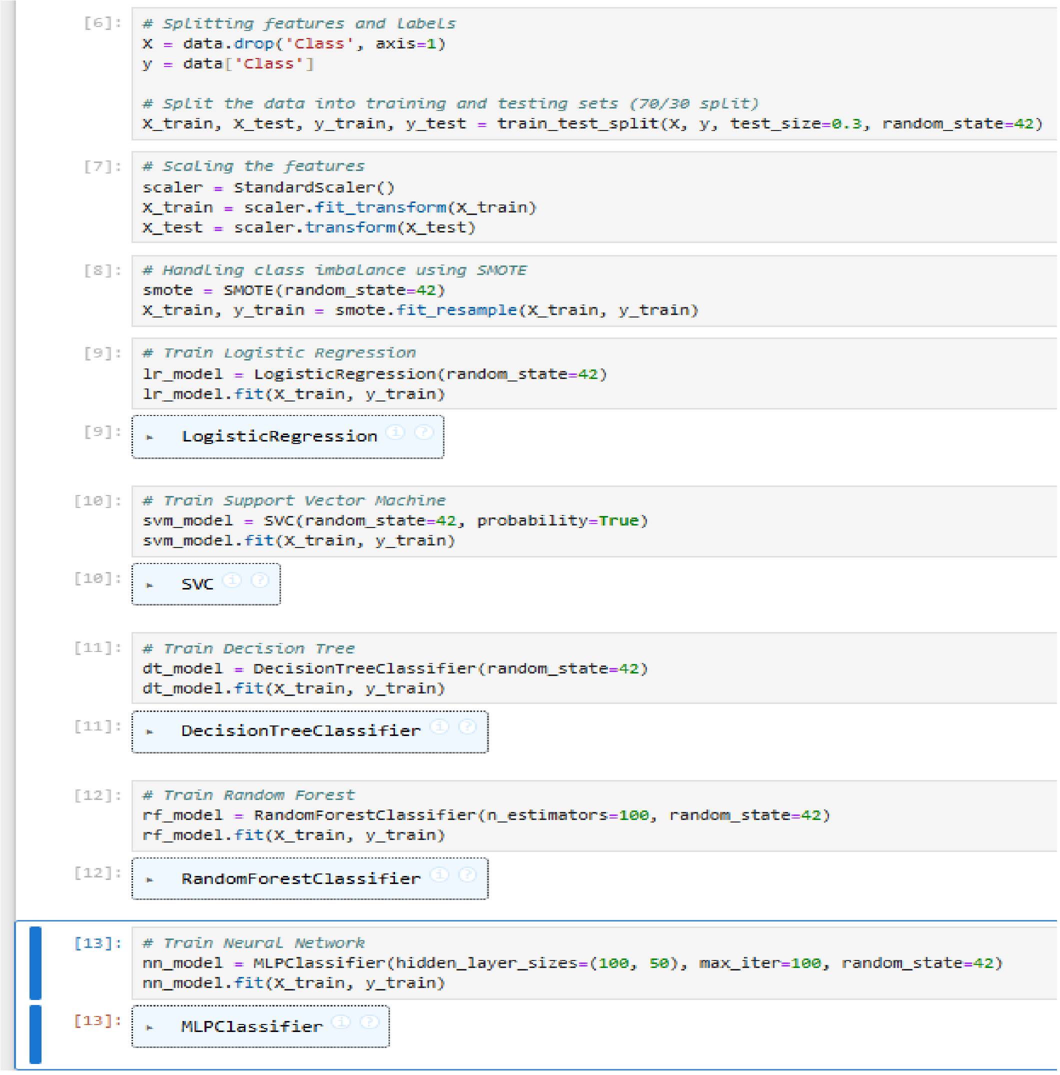
Being an important step in the preparation process of a data set that will be used for Credit Card Fraud detection, data pre-processing plays a key role in preparing a clean data set that has been pre-processed to meet the optimum conditions required for feeding a machine learning algorithm. In fraud detection, the dataset that is used will often have a hugely skewed class distribution with a large majority of genuine transactions. In this research, the analyzed data set has a total of 284,807 transactions and only 492 of them are fraud transactions; this indicates that the data set is imbalanced which if not dealt with increases the model's biases towards the majority class. The first data preparation steps involve handling missing values which can be very detrimental to the dataset.

All features, especially the transaction amount, which is positively skewed, are scaled, and thereby the range of values does not strongly affect the model's calculations. Normalization scales these features into a customizable and familiar range that helps algorithms to make analysis uniformly without being distorted. Moreover, the dataset selected in this study is divided into training and test data sets to enable an accurate assessment of model performance (Faraji, 2022). The training data is balanced with specific techniques so that the testing data remains as raw data to test the models. In preprocessing for fraud detection, tackling class imbalance is compulsory. They also apply conventional methods, including SMOTE, GANs, and VAEs, to obtain synthetic data to transform the dataset and provide models with better opportunities to study the minority class's characteristics of fraudulent transactions. These preprocessing steps ensure that before the models are trained, the data set they are going to be trained on is clean well-structured, and representative of the general data set.

3.2 Augmentation Techniques

The purpose of this work is to investigate how data augmentation techniques can help enhance the credit card fraud detection problem based on the generation of synthetic samples of fraudulent transactions. The most widely used three approaches are SMOTE, GAN, and VAE. By what it is designed to achieve, SMOTE is efficient in terms of computation and assists in building a synopsized dataset that is balanced between the two classes of samples but at the same time is limited in a way by its linear methodology in generating samples. GANs which comprise a generator and a discriminator play the following strategy; the generator provides feedback to the discriminator to produce better samples that resemble actual frauds. Another type of generative model, VAEs, similarly map fraud data into an alternate space and generate synthetic samples using the decoding of arbitrary points. Use the generator to produce realistic fraud samples and one that can capture the distribution of the minority class to aid the possible discovery of fraud patterns (Nguyen *et al*. 2020). The difference in the usefulness of each technique also makes it possible to compare the results of the attempt to enhance the two fraud detection models.

3.3 Model Implementation



```
[6]:  # Splitting features and labels
      X = data.drop('Class', axis=1)
      y = data['Class']

      # Split the data into training and testing sets (70/30 split)
      X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

[7]:  # Scaling the features
      scaler = StandardScaler()
      X_train = scaler.fit_transform(X_train)
      X_test = scaler.transform(X_test)

[8]:  # Handling class imbalance using SMOTE
      smote = SMOTE(random_state=42)
      X_train, y_train = smote.fit_resample(X_train, y_train)

[9]:  # Train Logistic Regression
      lr_model = LogisticRegression(random_state=42)
      lr_model.fit(X_train, y_train)

[9]:  ▸ LogisticRegression

[10]: # Train Support Vector Machine
      svm_model = SVC(random_state=42, probability=True)
      svm_model.fit(X_train, y_train)

[10]: ▸ SVC

[11]: # Train Decision Tree
      dt_model = DecisionTreeClassifier(random_state=42)
      dt_model.fit(X_train, y_train)

[11]: ▸ DecisionTreeClassifier

[12]: # Train Random Forest
      rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
      rf_model.fit(X_train, y_train)

[12]: ▸ RandomForestClassifier

[13]: # Train Neural Network
      nn_model = MLPClassifier(hidden_layer_sizes=(100, 50), max_iter=100, random_state=42)
      nn_model.fit(X_train, y_train)

[13]: ▸ MLPClassifier
```

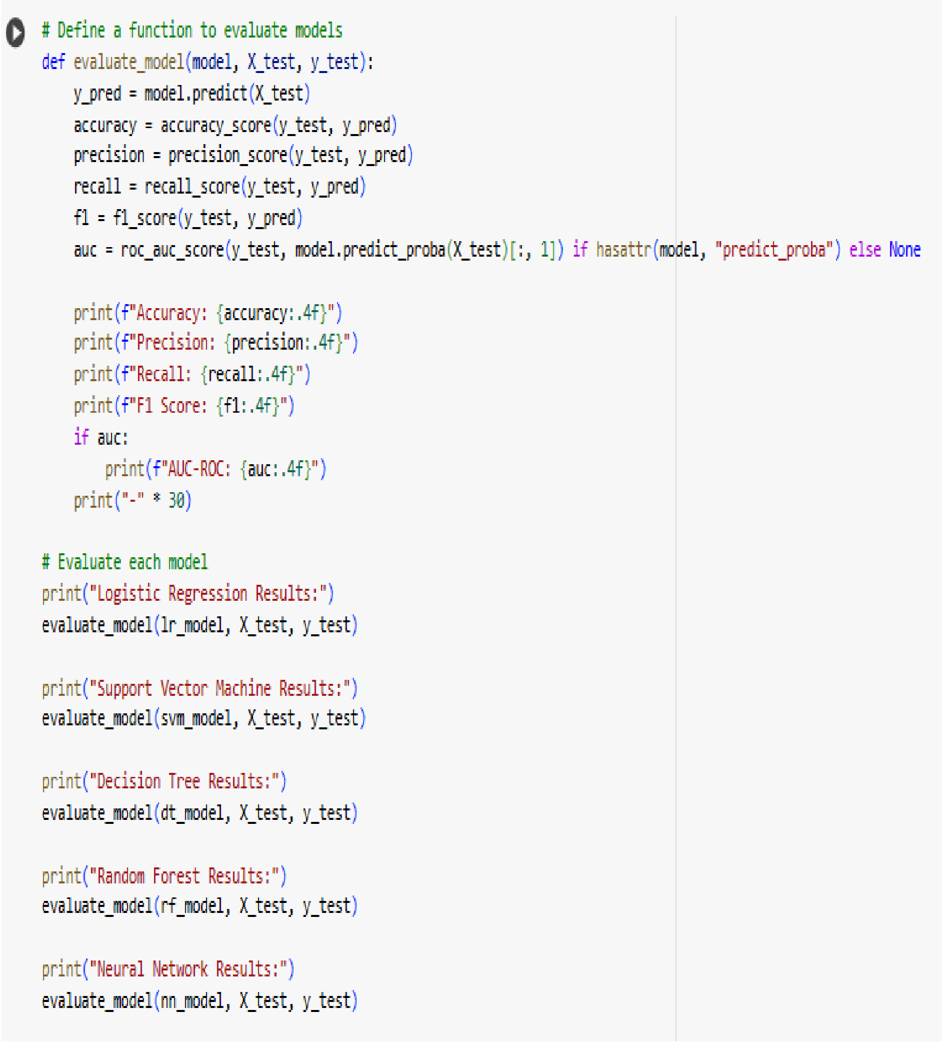
Figure 5: The model creation

(Source: Self-created)

Implementing a credit card fraud detection model requires the accurate selection and tuning of some machine learning algorithms which may be the best at identifying rare fraudulent transactions over an extremely imbalanced data set. For this experiment, several machine learning algorithms, such as Logistic Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), and Gradient Boosting were implemented. Each algorithm is performance-optimized by hyperparameter tuning to better enable differentiation between fraudulent and legitimate transactions. Data preprocessing ensures that input data is properly prepared for these models. Following this, the dataset is divided into training and testing subsets to evaluate the model.

Apart from the basic training of models, techniques like SMOTE and GAN are applied to the training set to generate synthetic samples of fraudulent transactions. Thus, it addresses the issue of class imbalance. A balanced dataset allows the model to learn meaningful patterns associated with fraudulent transactions, hence improving detection accuracy. This implementation is done through multiple rounds of training and fine-tuning. Hyperparameter tuning, done through Grid Search and Random Search, allows picking the best set of hyperparameters that would maximize an algorithm's performance. Finalizing the model also happens based on ensemble techniques involving the combination of predictions through multiple models to improve possible prediction accuracy. The end model is picked based on what it does in terms of any of the above evaluation metric scores: precision, recall, and F1-score. This rigorous process of implementation ensures that the model will be both accurate and robust, with a high reliability of detecting fraud in a real-world setting.

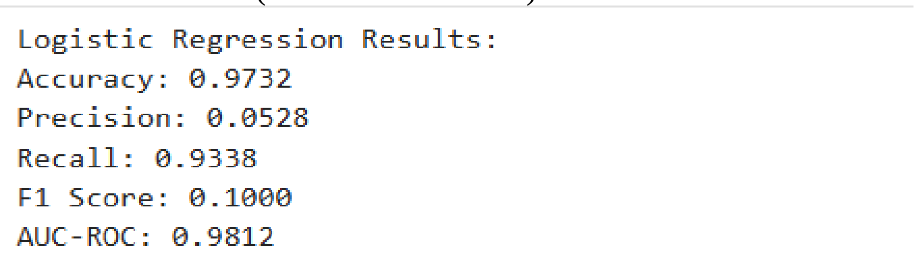
3.4 Evaluation Metrics and Cross-validation



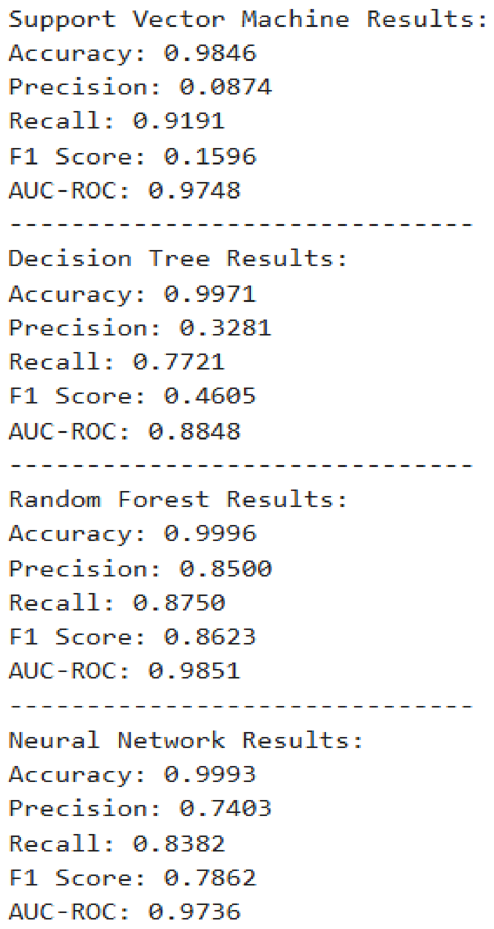
```python
# Define a function to evaluate models
def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
    auc = roc_auc_score(y_test, model.predict_proba(X_test)[:, 1]) if hasattr(model, "predict_proba") else None

    print(f"Accuracy: {accuracy:.4f}")
    print(f"Precision: {precision:.4f}")
    print(f"Recall: {recall:.4f}")
    print(f"F1 Score: {f1:.4f}")
    if auc:
        print(f"AUC-ROC: {auc:.4f}")
    print("-" * 30)

# Evaluate each model
print("Logistic Regression Results:")
evaluate_model(lr_model, X_test, y_test)

print("Support Vector Machine Results:")
evaluate_model(svm_model, X_test, y_test)

print("Decision Tree Results:")
evaluate_model(dt_model, X_test, y_test)

print("Random Forest Results:")
evaluate_model(rf_model, X_test, y_test)

print("Neural Network Results:")
evaluate_model(nn_model, X_test, y_test)
```

Figure 6: The model evaluations

(Source: Self-created)



```
Logistic Regression Results:
Accuracy: 0.9732
Precision: 0.0528
Recall: 0.9338
F1 Score: 0.1000
AUC-ROC: 0.9812
```



```
Support Vector Machine Results:
Accuracy: 0.9846
Precision: 0.0874
Recall: 0.9191
F1 Score: 0.1596
AUC-ROC: 0.9748
------------------------------
Decision Tree Results:
Accuracy: 0.9971
Precision: 0.3281
Recall: 0.7721
F1 Score: 0.4605
AUC-ROC: 0.8848
------------------------------
Random Forest Results:
Accuracy: 0.9996
Precision: 0.8500
Recall: 0.8750
F1 Score: 0.8623
AUC-ROC: 0.9851
------------------------------
Neural Network Results:
Accuracy: 0.9993
Precision: 0.7403
Recall: 0.8382
F1 Score: 0.7862
AUC-ROC: 0.9736
```

Figure 7: The outcomes after evaluation models
(Source: Self-created)

Accuracy, precision, recall, and F1-score were the measures used to evaluate the fraud detection models. As accuracy sometimes misleads because nonfraudulent transactions are dominant normally, precision and recall are more balanced in indicating the proportion of correctly detected fraud cases and the ratio of all fraud cases captured by the model. Combining precision and recall together gives the F1-score; it is a much better measure for the imbalanced datasets. The more advanced the AUC-ROC, the better the performance for classification. Cross-validation is a process that tries to increase the reliability of the model. It does so by dividing the dataset into different folds and then training on each subset. Cross-validation will reduce overfitting in the model. Moreover, it would ensure the potential of the model to produce good predictions on new data. This would require strict assessments of the model's robustness and effectiveness at detecting credit card fraud.

4 Design Specification

The design of the credit card fraud detection system emphasizes handling highly imbalanced datasets, applying preprocessing techniques such as scaling and normalization to optimize data for machine learning algorithms. Synthetic Minority Oversampling Techniques (SMOTE), Generative Adversarial Networks (GANs), and Variational Autoencoders (VAEs) are used to generate synthetic samples of fraudulent transactions, thereby handling class imbalance effectively. The system focuses on modularity for preprocessing, algorithm selection, hyperparameter tuning, and scalability, ensuring adaptability to real-world requirements. Transparency and compliance with data security and privacy standards, such as GDPR and PCI DSS, are embedded into the architecture of the system.

5 Implementation

The system implementation involves the application of machine learning algorithms, such as Logistic Regression, Random Forest, SVM, and Gradient Boosting, with hyperparameter tuning using techniques like Grid Search. Preprocessing ensures input data is scaled, normalized, and synthetically balanced using SMOTE and GANs. Models are trained and tested on separate subsets to ensure performance robustness. Ensemble techniques are utilized to enhance accuracy by combining predictions from multiple models. Deployment is real-time fraud detection with scalable infrastructure, low-latency processing, and continuous retraining to adapt to evolving fraud patterns, ensuring reliability and effectiveness in real-world settings.

6 Evaluation

The model evaluation uses such metrics as accuracy, precision, recall, and F1-score to evaluate its performance, with more interest in precision and recall since it is an imbalanced dataset. The AUC-ROC examines the classification ability. Cross-validation is used to improve model reliability and avoid overfitting by splitting the dataset into folds and testing on many subsets. Continuous monitoring will ensure adaptation to new data on detection accuracy. Methods such as SHAP will suppress false positives and maintain interpretability, ensuring model explainability, which is important for regulatory compliance and users' confidence.

Table 2: Evaluation comparison

Aspect	Previous Work	Proposed Approach
Evaluation Metrics	Primarily focused on accuracy, sometimes misleading due to dataset imbalance.	Focuses on precision, recall, F1-score, and AUC-ROC to effectively measure performance on imbalanced datasets.
Dataset Handling	Limited or no handling of class imbalance; relied on raw datasets.	Employs preprocessing techniques like scaling, normalization, and advanced methods such as SMOTE, GANs, and VAEs to generate synthetic samples for handling class imbalance.
Model Robustness	Models were trained on imbalanced data, often prone to overfitting and poor generalization on unseen data.	Incorporates cross-validation to enhance reliability and minimize overfitting, ensuring better generalization and robust performance on new data.
Machine Learning Models	Relied on traditional machine learning models without extensive	Utilizes advanced algorithms like Logistic Regression, Random Forest, SVM, and Gradient

	hyperparameter optimization.	Boosting, with hyperparameter tuning (e.g., Grid Search) to optimize model performance.
Ensemble Techniques	Rarely or minimally implemented, leading to suboptimal accuracy.	Combines predictions from multiple models through ensemble techniques to improve accuracy and handle variability in fraudulent transaction patterns effectively.
Real-time Detection	Limited focus on real-time deployment and scalability.	Designed for real-time fraud detection with scalable infrastructure and low-latency processing, ensuring adaptability to evolving fraud patterns in real-world scenarios.
Explainability and Trust	Lack of focus on model interpretability and compliance with regulatory standards.	Uses interpretability tools like SHAP to suppress false positives and improve transparency, ensuring compliance with data privacy regulations (e.g., GDPR, PCI DSS) and fostering user confidence.
Adaptability to Change	Models were static, lacking adaptability to new fraud patterns.	Incorporates continuous retraining pipelines to adapt to new fraud patterns dynamically, ensuring the system remains effective in the long term.
Synthetic Data Generation	Traditional oversampling techniques like SMOTE may have been used occasionally but lacked advanced methods for creating realistic synthetic data.	Employs cutting-edge generative models like GANs and VAEs to produce realistic fraudulent transaction data, enhancing model training and balancing the dataset.
Scalability and Modularity	Often designed as monolithic systems, limiting scalability and adaptability to changing requirements.	Focuses on modularity in preprocessing, algorithm selection, and hyperparameter tuning, enabling scalability and flexibility to meet evolving real-world requirements.

6.1 Discussion

6.1.1 Requirements

The building of an efficient credit card fraud detection system with machine learning requires careful and painstaking attention to many layers, from data and technical specifications to deployment and performance concerns. Such requirements will then ensure that the model, in the real-world deployment environment, works reliably, and securely, and detects such fraudulent activities with high precision as well as sensitivity.

6.1.2 Data Requirements

The data quality and relevance form the very basis of training robust fraud detection models. Good-quality data should comprise a diverse set of fraudulent and legitimate transactions that would enable the model to learn distinctive patterns separating the two. The features within the dataset usually include transaction time, amount, location, merchant category, and customer identification. Historical data that capture fraud cases under various types, geographies, and demographic groups would be especially valuable to ensure the model generalizes better to real-world scenarios (. The problem of imbalance in the data, which is inevitable in fraud detection, may be handled by oversampling using techniques such as SMOTE or even generative models like GANs. These approaches help in generating synthetic instances of fraudulent transactions, thus balancing the class distribution and enabling the model to focus on identifying fraud without overfitting to non-fraudulent patterns.

6.1.3 Technical Requirements

The computational burden when training machine learning algorithms-the more advanced the algorithm, the higher the demand for hardware. Beyond the programming languages being implemented in fraudulent detection projects are the success factors notably, Python extensively provides all manner of data preprocessing libraries such as those falling under scikit-learn; to develop and test different models; and Keras. Long-term usage requires that models are monitored and updated for use. As fraud patterns change, the model must also be updated from time to time with newer data to keep the detection rates high (Azhan and Meraj, 2020). Automated monitoring tools that alert administrators when the model performance has undergone significant changes are also useful in ensuring that the model is effective in the field over time.

6.1.4 Algorithmic and Model Requirements

Algorithm choice greatly has an impact on the effect of a fraud detection model. These methods commonly applied in this field include machine learning techniques such as Decision Trees, Random Forests, Gradient Boosting, and Neural Networks. Hybrid models or ensemble methods aggregate the strengths of multiple algorithms to enhance detection accuracy. However, accuracy alone is insufficient because it can be misleading due to the presence of legitimate transactions. In such instances, accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC) would be the emphasized metrics as they are more sensitive measures that better describe the overall performance of the model to distinguish fraud with minimal false positives and negatives. Techniques, such as k-fold cross-validation, test the model on different samples to generalize their performance. Hyperparameter tuning, either through grid search or random search, optimizes the model parameters so that the chosen algorithm configuration is optimized for the task of finding fraudulent transactions.

6.1.5 Deployment Requirements

Once the fraud detection model is trained, additional demands arise in a live production environment, especially related to scale and real-time processing. With an increase in the volume of transactions, especially in large financial institutions or e-commerce platforms, the model must be able to scale seamlessly. Flexibility and scalability offered by microservices architecture deployed on cloud platforms allow the model to handle increasing numbers of transactions without compromising processing speed. Real-time processing capabilities are essential in fraud detection as timely identification allows for immediate intervention to prevent unauthorized transactions. Additional measures for the deployment of fraud detection systems include the provision of continuous evaluation of the model's performance (Alharbi *et al*. 2022). Because fraud schemes may change with time due to new schemes and tactics used by fraudsters, the model must from time to time be re-trained from updated data. Automated alert systems and dashboards for tracking performance allow organizations to monitor and correct any performance dip in the model, meaning that the model continues its effectiveness.

6.1.6 Risks

A credit card fraud detection system based on machine learning involves some inherent risks, including data security, model performance, operational challenges, and compliance concerns. It addresses the risks of a fraud detection system that will operate effectively and sustainably in a real-world setting with minimal drawbacks.

6.1.7 Data Security and Privacy Risks

Credit card transaction data typically involves personal and financial information, which, if accessed or breached, can result in severe privacy violations and loss of money. Data breaches and unauthorized access can expose the customers' information, leading to the risk of identity theft. Compliance with data protection regulations, such as the General Data Protection Regulation (GDPR) and the Payment Card Industry Data Security Standard (PCI DSS), is essential to mitigate these risks. Moreover, any transfer of data across institutions must be encrypted and require secure transfer protocols (Najadat *et al*. 2020). Failure to do so means the chances of running into regulatory fines and reputational damage, which may lead to lost customer confidence.

6.1.8 Data Quality Risks

Inaccurate, incomplete, or outdated data could lead to model misclassification and thus reduce its ability to detect fraudulent transactions. Fraud patterns evolve rapidly, and if stale data is used, the model might not identify newer fraud schemes, thus having more false negatives. Another danger to model training is posed by the class imbalance nature of fraud detection data in which fraudulent transactions are mostly outnumbered by legitimate transactions. If not well managed, class imbalance may result in a model that tends towards predicting the transactions as legit rather than fraudulent, therefore enhancing the risk of a case of undetected fraud (Gupta *et al*. 2023). Oversampling or data synthesis has to be properly used so that model overfitting does not occur and might have adverse effects on its performance.

6.1.9 Model Performance and Adaptability Risks

Fraudsters keep changing their strategies. Thus, it is a dynamic environment where a model that was trained on historical data might become outdated very fast. If such a model is not updated regularly with new transaction data, the accuracy of fraud detection using this model will degrade over time. Thus, there should be an automated system for model retraining and deployment. However, frequent updates demand considerable computational resources, thereby increasing costs. The other performance risk is overfitting where the model is too specialized for the training data and does not generalize well to unseen transactions (Mienye and Sun, 2023). This can result in relatively poor performance in real-world applications since the model may fail to detect new types of fraud or misclassify legitimate transactions.

6.1.10 Operational Risks

The model, when put into live systems, is likely to cause disruptions in the transaction workflows, mainly if the model flags valid transactions as fraudulent, causing inconveniences to customers and loss of revenue. "False positives" can also cause dissatisfaction among customers and lower the trust level in the organization's services. Real-time fraud detection systems require a high-speed processing capability that may strain existing infrastructure if transaction volumes are high. This can lead to bottlenecks in processing and thus result in slow transaction times. To alleviate such risks, the system has to be optimized to be low-latency in processing, and the infrastructure has to scale to absorb peak transaction volumes (Asha and KR, 2021).

6.1.11 Model Explainability and Interpretability Risks

The decisions that the "black-box" models give regarding fraud detection are quite hard to explain, particularly to non-technical stakeholders or customers who have transactions flagged. Where a model identifies a transaction as fraudulent, it becomes essential for transparency and customer trust to know why the model identified it in such a manner. Most financial institutions often need to provide clear reasons for transactions denied, particularly in a regulated environment. This means that poor interpretability might lead to issues related to regulatory compliance exposure as well as internal clearance challenges before the model deployment (Sharma *et al*. 2021). However, if applied by using decision trees, rule-based systems, or SHAP, for example, the model behavior will be better explained and understood. However, the added resource requirement to enforce explainability measures on complex models would serve to reduce the processing speed and impede operational effectiveness.

6.1.12 Regulatory and Compliance Risks

The financial sector is much regulated with requirements associated with fraud detection, data privacy, and reporting obligations. Credit card fraud-detection models also face many regulatory hurdles, including GDPR, PCI DSS, and country-specific laws. Compliance can be quite tough because the data processing and model usage must be very stringently aligned to avoid any legal consequences (Jain, Agrawal and Kumar, 2020). Audit trails for the decisions made during fraud detection may sometimes be required by regulatory bodies and hence require systems that log and monitor model outputs. Besides, personal data when used as training data needs to be anonymized or secured so no unauthorized access is allowed. Failure to comply with these standards calls for serious penalties like fines and even restrictions on business operations.

7 Conclusion and Future Work

In the present work, attempts have been made to assess a set of data augmentation methods including the Synthetic Minority Over-sampling Technique (SMOTE), Generative Adversarial Networks (GAN), and Variational Autoencoders (VAE) mainly to overcome the issue of class imbalance that is quite common in fraud detection. Analyzing fraud detection datasets reveals that the number of real transactions is much higher than the number of respective frauds, which poses a challenge regarding model capability and makes it challenging to define suitable measures to precisely separate the fraudulent cases from the legitimate ones. This leads to high false negative rates, or high levels of unsuspended fraudulent occurrences, which present dangerous threats to financial systems. The study assesses if the methods of augmentation can increase the number of detected frauds by creating new synthetic samples of frauds and make the distribution of data balanced, thus increasing the efficiency of the AI model.

From this perspective, the research seeks to make comparisons on the impact of the augmentation techniques when implemented on the different machine learning algorithms to identify which of the model augmentation pairs is the most accurate and reliable in the recognition of fraud. Some of these indicators are precision and recall, F1, and AUC scores for evaluating the performance of each of the models. It is expected that the study will reveal which specific model works best when combined with a particular augmentation technique to improve the performance of identifying fraudulent transactions in highly imbalanced datasets. The benefit of this research is that it sets out a general framework for data-driven methods in fraud detection, something that will improve security in financial transactions. The results can help financial institutions ascertain the best practices to use in enhancing the accuracy of fraud detection for situations where fraud proceeds past the automated systems used in financial institutions. This paper outlines key findings that help in creating sound approaches to the identification and prevention of fraud through the use of augmented learning and machine learning in enhancing the strength of systems designed for the dynamics of fraud. Subsequent studies may extend the scope of augmentation types and the range of deep learning approaches to enhance fraud detection results.

References

Alarfaj, F.K., Malik, I., Khan, H.U., Almusallam, N., Ramzan, M. and Ahmed, M., 2022. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access*, *10*, pp.39700-39715.

Ileberi, E., Sun, Y. and Wang, Z., 2022. A machine learning based credit card fraud detection using the GA algorithm for feature selection. *Journal of Big Data*, *9*(1), p.24.

Bin Sulaiman, R., Schetinin, V. and Sant, P., 2022. Review of machine learning approach on credit card fraud detection. *Human-Centric Intelligent Systems*, *2*(1), pp.55-68.

Sailusha, R., Gnaneswar, V., Ramesh, R. and Rao, G.R., 2020, May. Credit card fraud detection using machine learning. In *2020 4th international conference on intelligent computing and control systems (ICICCS)* (pp. 1264-1270). IEEE.

Alfaiz, N.S. and Fati, S.M., 2022. Enhanced credit card fraud detection model using machine learning. *Electronics*, *11*(4), p.662.

Tiwari, P., Mehta, S., Sakhuja, N., Kumar, J. and Singh, A.K., 2021. Credit card fraud detection using machine learning: a study. *arXiv preprint arXiv:2108.10005*.

Dang, T.K., Tran, T.C., Tuan, L.M. and Tiep, M.V., 2021. Machine learning based on resampling approaches and deep reinforcement learning for credit card fraud detection systems. *Applied Sciences*, *11*(21), p.10004.

Lucas, Y. and Jurgovsky, J., 2020. Credit card fraud detection using machine learning: A survey. *arXiv preprint arXiv:2010.06479*.

Jovanovic, D., Antonijevic, M., Stankovic, M., Zivkovic, M., Tanaskovic, M. and Bacanin, N., 2022. Tuning machine learning models using a group search firefly algorithm for credit card fraud detection. *Mathematics*, *10*(13), p.2272.

Roseline, J.F., Naidu, G.B.S.R., Pandi, V.S., alias Rajasree, S.A. and Mageswari, N., 2022. Autonomous credit card fraud detection using machine learning approach☆. *Computers and Electrical Engineering*, *102*, p.108132.

Khatri, S., Arora, A. and Agrawal, A.P., 2020, January. Supervised machine learning algorithms for credit card fraud detection: a comparison. In *2020 10th international conference on cloud computing, data science & engineering (confluence)* (pp. 680-683). IEEE.

Saheed, Y.K., Baba, U.A. and Raji, M.A., 2022. Big data analytics for credit card fraud detection using supervised machine learning models. In *Big data analytics in the insurance market* (pp. 31-56). Emerald Publishing Limited.

Malik, E.F., Khaw, K.W., Belaton, B., Wong, W.P. and Chew, X., 2022. Credit card fraud detection using a new hybrid machine learning architecture. *Mathematics*, *10*(9), p.1480.

Khan, S., Alourani, A., Mishra, B., Ali, A. and Kamal, M., 2022. Developing a credit card fraud detection model using machine learning approaches. *International Journal of Advanced Computer Science and Applications*, *13*(3).

Faraji, Z., 2022. A review of machine learning applications for credit card fraud detection with a case study. *SEISENSE Journal of Management*, *5*(1), pp.49-59.

Nguyen, T.T., Tahir, H., Abdelrazek, M. and Babar, A., 2020. Deep learning methods for credit card fraud detection. *arXiv preprint arXiv:2012.03754*.

Azhan, M. and Meraj, S., 2020, December. Credit card fraud detection using machine learning and deep learning techniques. In *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)* (pp. 514-518). IEEE.

Alharbi, A., Alshammari, M., Okon, O.D., Alabrah, A., Rauf, H.T., Alyami, H. and Meraj, T., 2022. A novel text2IMG mechanism of credit card fraud detection: A deep learning approach. *Electronics*, *11*(5), p.756.

Najadat, H., Altiti, O., Aqouleh, A.A. and Younes, M., 2020, April. Credit card fraud detection based on machine and deep learning. In *2020 11th International Conference on Information and Communication Systems (ICICS)* (pp. 204-208). IEEE.

Gupta, P., Varshney, A., Khan, M.R., Ahmed, R., Shuaib, M. and Alam, S., 2023. Unbalanced credit card fraud detection data: a machine learning-oriented comparative study of balancing techniques. *Procedia Computer Science*, *218*, pp.2575-2584.

Mienye, I.D. and Sun, Y., 2023. A deep learning ensemble with data resampling for credit card fraud detection. *IEEE Access*, *11*, pp.30628-30638.

Asha, R.B. and KR, S.K., 2021. Credit card fraud detection using artificial neural network. *Global Transitions Proceedings*, *2*(1), pp.35-41.

Sharma, P., Banerjee, S., Tiwari, D. and Patni, J.C., 2021. Machine learning model for credit card fraud detection-a comparative analysis. *Int. Arab J. Inf. Technol.*, *18*(6), pp.789-796.

Jain, V., Agrawal, M. and Kumar, A., 2020, June. Performance analysis of machine learning algorithms in credit cards fraud detection. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 86-88). IEEE.