Online fraud prediction using Machine Learning Models

MSc Research Project
MSc in Data Analytics

Ajay Krishnaa Ayyakutty Ramesh
Student ID: x21193649

School of Computing
National College of Ireland

Supervisor: Dr. Rejwanul Haque

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……Ajay Krishnaa Ayyakutty Ramesh ………………………………………………

Student ID: ………x21193649…………………………………………………………………………..……

Programme: ………MSc in Data Analytics……………………… **Year:** ……2023………………..

Module: …………Research Project…………………………………………………………….………

Supervisor: …………Dr.Rejwanul Haque……………………………………………….………

Submission Due Date: ………………18-09-2023………………………………………………………….………

Project Title: …Online fraud detection using Machine learning Models…….………

Word Count: ………6746…… **Page Count**……25…………………..……..

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: ……………………… Ajay Krishnaa Ayyakutty Ramesh……………………

Date: ……………14-08-2023……………………

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):	

Online fraud prediction using Machine Learning Models

Ajay Krishnaa Ayyakutty Ramesh
Student ID: x21193649

Abstract

This study introduces an innovative approach to detecting online bank fraud by combining machine learning with ensemble approaches. Online bank fraud could be a major issue for the financial sector since it causes considerable costs and harms buyers confidence within the banking sector. The objective of this research project is to make a fraud detection system that can dependably separate between honest to goodness and fraudulent transactions, decreasing the monetary hazard for both business and their customers. The ponder begins with a profound jump into the current writing on the subject of online bank fraud detection and the numerous distinctive machine learning strategies that have been created to combat it. Information on the design requirements, such as data pretreatment, model selection, and the creation of the ensemble technique, are provided next. The primary results of this study show that the ensemble approach is better to the more common machine learning ensemble methods. The suggested approach improves upon existing methods in terms of detection accuracy, precision, recall, and F1-score. To improve fraud detection effectiveness, the study highlights the value of mixing model predictions. These results have substantial ramifications for the financial sector since the suggested approach may improve security and inspire confidence in digital exchanges.

1 Introduction

1.1 Background

Online banking has emerged as an essential component of today's sophisticated monetary systems, thanks to the meteoric rise in internet use and the steady march of technical progress explained by Wang et.al (2020). The ease and availability provided by internet banking have fundamentally altered the manner in which people and organisations undertake financial transactions. On the other hand, this transition to digital has also resulted in an alarming surge in the number of instances of fraud committed online. The use of advanced strategies and methods by cybercriminals to attack weaknesses in online banking systems is resulting in significant financial losses and undermining customer faith in digital financial services explained by Yaacoub et.al (2020).

When it comes to combating the ever-evolving nature of online fraud, the traditional techniques of fraud detection, which are mostly focused on rule-based systems, have been shown to be insufficient explained by Shen et.al (2021). These systems often fail to react to the dynamic behaviour of fraudsters, which leads to high false-positive rates and, ultimately, dissatisfied customers as well as operational inefficiencies for financial institutions. In order to solve these difficulties, an increasing focus is being placed on the investigation of innovative technology solutions that might improve online fraud detection. The ability of machine learning algorithms to recognise patterns and irregularities makes them an excellent choice for applications that deal with fraud detection.

Machine learning algorithms have shown significant promise in a variety of different fields explained by Mohammed et.al (2018). Machine learning models may detect fraudulent activity with a better level of accuracy and productivity than traditional methods can achieve by using massive amounts of past transaction data and continually learning from fresh data. By utilizing the capabilities of machine learning algorithms, this study intends to form a commitment to the current endeavors that are being made to progress the identification of online extortion. This extend points to make strides the security and dependability of online banking by building up a solid fraud detection system. As a result, it'll ideally protect consumers and companies from the threats that are related with online fraud.

1.2 Importance of the Study

It is impossible to overestimate the significance of spotting fraudulent activity on the internet. The likelihood of becoming a victim of fraud increases continuously as more financial dealings are moved online. The rule-based approaches that have traditionally been utilized for fraud detection have been appeared to be lacking for keeping up with the progressively complex procedures utilized by fraudsters. As a result, there's an critical need to explore and put into hone arrangements that are more cutting-edge and intelligent and are based on machine learning algorithms explained by Priya et.al (2021). This project intends to contribute to the improvement of productive fraud detection systems that can protect the money related resources of clients and increment certainty in online banking by conducting research in this zone or domain.

1.3 Aim

The point of this study is to look at and plan an successful system for recognizing fraudulent bank transactions by utilizing machine learning techniques. The purpose of the project is to urge an understanding of how machine learning strategies may be utilized to identify and anticipate fraudulent activity in real time.

1.4 Objectives

The particular goals of this research are as follows:

1. To survey the existing literature on online fraud discovery strategies and machine learning algorithms
2. To explore various machine learning algorithms suitable for fraud detection and analyze their effectiveness.
3. Design and implement a prototype online fraud detection system.
4. To evaluate the performance of the proposed system using real-world datasets.
5. To assess the practical implications and challenges of deploying the fraud detection system in the context of online banking.

1.5 Research Questions

This study will address the following research questions:

1. Which machine learning algorithms are most effective in detecting online fraud?

1.6 Rationale

The purpose of this project is to create a commitment to the expanding body of knowledge within the field of detecting fraudulent activity on the online banking with the interaction of machine learning. This study has the potential to assist financial institutions and undertakings within the assurance of their customers from fraudulent activities and the consequent development of customer trust in online banking services through the advancement of an productive fraud detection system.

1.7 Structure

The report is broken down into seven parts: background, previous studies, research methodology, design specifications, implementation, evaluation, and summary and suggestions for further study. The Introduction sets the stage for the rest of the paper, discussing such topics as the need of multi-factor authentication for safe online banking and how online fraud may be detected using machine learning techniques. Online fraud detection and multi-factor authentication are the focus of the research evaluated in the works cited below. Data gathering, dataset selection, and machine learning algorithm choices are all outlined in detail in the research methodology. The proposed fraud detection system, which incorporates multi-factor authentication, is described in depth in the design specification. The implementation phase involves the system's actual creation and rollout. Measurements of the system's efficiency are analysed in an evaluation. The results are summed up in the conclusion, which also includes suggestions for improving the system and directions for further study on the topic of online banking security.

2 Related Research Works

2.1 Machine Learning Approaches for Fraud Detection

Machine learning methods based solution have played an important role in developing possible solutions to the problem of online fraud. ML models like Logistic regression, decision trees, and support vector machines (SVMs) are examples of classical machine learning techniques that have been widely used in the past. According to Khan et al., (2022) These strategies centred on generating custom characteristics from transaction data in order to spot red flags for potential fraud. They were only somewhat effective since they couldn't detect subtle, non-linear patterns in the data.

In addition to this Hu et al., (2021) explains that advanced machine learning models, such as Random Forests and Gradient Boosting Machines (GBM), were developed to help researchers get around these restrictions. By merging the results of numerous individual classifiers, the accuracy of these ensemble methods for detecting fraud was significantly increased. To further improve model performance, feature engineering methods were optimised to better extract useful information from transactional data.

However, Valavan et.al (2023) suggest that deep learning-based systems have emerged as a possible answer since fraudsters consistently modify their strategies. Intricate temporal patterns and spatial correlations in credit card transactions are captured with remarkable accuracy by deep neural networks, especially Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). These models self-learned representations from raw data, doing away with the requirement for painstakingly designed features. The effectiveness of machine learning techniques isn't without its share of difficulties, such as dealing with the extremely unbalanced nature of fraud data, explaining models, and scaling them. Exploration of cutting-edge methods like as XGBoost, Autoencoders, and Graph-based models, which show promise in solving these difficulties, keeps research in this area moving forward.

2.2 Deep Learning and Neural Network-based Solutions

Fraud detection has been a challenging area, but recent years have showed the promise of deep learning and neural network-based solutions, especially in the face of complex and ever-changing fraud patterns. According to Baratzadeh et.al., (2022) these techniques eliminate the need for human feature engineering by automatically learning important features and representations straight from raw transaction data using the power of deep neural networks. In addition to this Benchaji et al., (2021) Recurrent Neural Networks (RNNs) have been shown to be successful in identifying evolving fraud trends because of their ability to capture sequential relationships inside credit card transactions. Similarly, Convolutional Neural Networks (CNNs) have shown success in identifying fraud incidents based on localised patterns by successfully capturing spatial relationships in transaction data.

Additionally, Alarfaj et.al., (2022) performance has been enhanced by using ensemble approaches to combine deep learning models with conventional machine learning strategies. The Stacked Auto-encoder is only one example of a model that has been successfully implemented for use in fraud detection because to its ability to learn hierarchical representations. Deep learning algorithms have been exceptionally effective, but they are not without disadvantages, such as the high resource and compute prerequisites and the conceivable need of explainability of the resulting models. In any case, researchers are continuously working to improve and expand upon these strategies in arrange to supply more dependable and versatile strategies of recognizing online bank fraud.

2.3 Ensemble Techniques in Fraud Detection

Due to their capacity to mix several classifiers and utilise varied viewpoints for enhanced accuracy, ensemble approaches have emerged as viable tools for fraud detection. To that end, this study explores the use of ensemble approaches to improve fraud detection efficiency. According to Latha et.al, (2019) The Bagging method is a common strategy in which several classifiers are trained on independent data sets and their combined predictions are determined by a majority vote or the average of the results. This technique has the potential to strengthen the fraud detection model by decreasing overfitting. In addition to this Priscilla et.al (2020) explains that the Boosting strategy is another powerful ensemble method that emphasises training weak classifiers in sequence and providing greater weight to misclassified cases.

Through this repeated process, the model is trained to prioritise the most challenging fraud situations, increasing its overall detection efficiency.

Additionally, Zhou et.al (2022) explains that studies have investigated unique ensemble methodologies such as Stacking, in which the predictions of several classifiers are utilised as input to a meta-classifier, which then learns from the outputs of the many classifiers to reach a conclusion. This method shows potential for catching sophisticated fraud patterns and increasing precision. The whole body of work on ensemble approaches has shown their ability to greatly enhance online fraud detection performance, making them an important area of study for creating more efficient and trustworthy fraud detection systems.

2.4 Handling Class Imbalance in Fraud Detection

Since legitimate transactions far outnumber fraudulent ones, addressing class imbalance in online fraud detection is an important topic of study. According to Benchaji et.al (2019) Traditional machine learning algorithms sometimes produce biassed models because of their inability to accurately recognise the minority class.

A number of approaches to this issue have been examined. Strategies that resample the data by favoring the minority group or the majority group are frequently utilized. This aids in developing a more well-rounded dataset, which in turn boosts efficiency. According to Uma et al., (2023) study synthetic sampling, utilising methods like the Synthetic Minority Over-sampling Technique (SMOTE), is another method. To achieve a more equitable class distribution and improve the classifier's capacity to spot fraud, SMOTE generates synthetic examples of the minority class by interpolating existing instances.

However, Sailusha et al., (2020) study have also experimented with other performance measurements, such as precision-recall curves and F1 score, that are more sensitive to class imbalance than accuracy. The model's effectiveness on unbalanced data may be better gauged using these criteria. It has also been looked at how to successfully deal with class imbalance via the use of ensemble techniques. In addition to this Strelcenia et.al (2023) explains that by combining many classifiers into a "ensemble," we may take use of their individual strengths and reduce the negative effect of class imbalance on the model's accuracy. Data resampling, synthetic sample generation, alternative performance metrics, and ensemble methods are just some of the methods that have been explored to effectively address class imbalance, which is crucial for developing accurate online bank fraud detection systems.

3. Research Methodology

This study employs a research technique aimed at creating a machine learning-based online fraud detection system and assessing its efficacy. Data gathering, data preparation, model selection, system installation, and assessment are all stages of the technique. This project is developed by following KDD methodology.

3.1 Data Collection:

Building a reliable system to identify online fraud requires extensive data collecting. To do so, it is vital to have a large sufficient dataset that incorporates both genuine and fraudulent cases of online banking fraud explained by Mokhtari et al., (2021). Each effort will be made to choose a dataset that's comprehensive, precise, and representative of the world at huge. We will take into account publicly accessible datasets from authoritative sources like Kaggle and the UCI Machine Learning Repository. Furthermore, adhering to privacy and security standards may provide access to private information via partnership with financial institutions.

The dataset is available in kaggle for this project and it is imported in google colab file for the research purposes

3.2 Data Pre-processing:

To efficiently train a machine learning model, the raw dataset must first undergo data preparation. Inconsistencies, missing values, and irrelevant characteristics in the obtained data may have an effect on the quality of the model. Noise and irregularities in the data will be cleaned out to help with these problems explained by Itoo et.al (2020). Methods like mean imputation and interpolation will be used to fill in any blanks caused by missing data. In order to construct useful features for the fraud detection models, feature engineering will be used. In order to ensure consistency and to avoid features with bigger scales from dominating model training, data normalization will be done to standardize the numerical features. Data preparation is the process of preparing a dataset for model selection and subsequent deployment by transforming it into an organized and useful manner.

3.3 Model Selection:

Section 2's literature study sheds light on current machine learning methods used in fraud detection. A group of promising algorithms will be chosen on the basis of this review.

Decision trees, random forests, SVM, logistic regression, and neural networks are some examples of possible algorithms.

3.4 System Implementation:

The chosen machine learning techniques will be implemented in Python to produce the anti-fraud software. We will put the models into action using libraries like TensorFlow and Scikit-learn that are available in python.

3.5 Evaluation Methodology:

The dataset will be partitioned into training and testing sets for the purpose of assessing the efficacy of the fraud detection system. The performance of the model in recognizing occurrences of fraud will be measured employing a number of distinctive evaluation measures, including precision, accuracy, recall, F1 score, and area under the receiver working characteristic curve (AUC-ROC). To ensure the unwavering quality of the findings, cross-validation will be utilized.

3.6 Experiment Setup:

To test the efficacy of the online fraud detection system, the experimental design includes a number of separate studies. Using a stratified sample method, the dataset will be split into a training set and a testing set, with the same number of valid and invalid transactions in each explained by Awoyemi et.al (2017). separate situations with differing degrees of fraudulent activity and multi-factor authentication settings will be the subject of separate research.

The findings will be trusted since a cross-validation method will be used. Accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve (AUC-ROC) are only few of the common assessment metrics that will be used to assess the system's performance. In order to speed up the process of training and evaluating models, we will be running our experiments in a high-performance computing setting.

3.7 Ethical Considerations:

Data privacy and confidentiality will be rigorously maintained during the study. Ethical standards and institutional regulations will be followed throughout the study. The dataset is available under public access in kaggle. So there is no ethical concerns in the dataset used for this research project.

3.8 Limitations:

It is critical to recognise the study methodology's possible shortcomings. Constraints in real-world deployment of the system, as well as the quantity and representativeness of the dataset, the availability of labelled fraud data, and other factors, may also be limiting factors. The research report will incorporate an examination of these limitations in arrange to provide readers a full picture of the study's destinations and results. Taking after this research plan, we need to make an successful online fraud detection system that improves the security of online banking by combining the qualities of machine learning algorithms with those of multi-factor verification. The evaluation's discoveries and results will include to what's as of now known and clear the way for more compelling measures to be taken against online fraud.

4. Design specification

The online bank fraud detection system's implementation strategies, architectures, and frameworks are described in detail in the design specification section. It also details what's needed for the system to get up and running smoothly. In addition, if a new method or model is suggested, its functioning is described in great depth.

4.1 Techniques and Architectures

4.1.1 Models for Machine Learning

Several machine learning models, each capable of identifying patterns and making predictions based on input characteristics, are used in online bank fraud detection systems. Decision Trees, Random Forests, K-Nearest Neighbors, Support Vector Machines, Multilayer Perceptrons and Ensemble stacking of models are the models used here.

Decision Tree (DT): A decision tree is a model that generates inferences by recursively partitioning tuples according to the importance of their attributes. It functions as a decision tree that ultimately sorts data into predetermined buckets.

Random Forest (RF): Random Forest is an ensemble method that, during training, builds several decision trees and then averages or votes on the results. The model's precision is increased and overfitting is minimised thanks to this ensemble method.

K-Nearest Neighbors (KNN): Classifying data points according to the majority class of their "K" nearest neighbours, KNN is a straightforward technique. This is a non-parametric, instance-based approach that works well for classification problems.

Support Vector Machine (SVM): Support vector machines are a powerful supervised learning technique with applications in both classification and regression. It searches a high-dimensional feature space for the best-separating hyperplane between data points belonging to distinct classes.

Multi-Layer Perceptron (MLP): The MLP is a multi-layer, fully-connected artificial neural network. Because of its pattern-learning prowess, it finds widespread use in machine intelligence.

4.1.2 Data Resampling Techniques

Data resampling methods are used to correct the skewed distribution of demographic characteristics in the online fraud dataset. The following procedures are used to generate evenly distributed training sets:

Random Oversampling: The minority class's representation is increased by a random number of copies.

SMOTE (Synthetic Minority Over-sampling Technique): For the underrepresented group, it creates simulated data by interpolating feature vectors from nearby cases.

SMOTE-ENN (SMOTE combined with Edited Nearest Neighbors): Combines the sample-generation capabilities of SMOTE with the noise-reduction capabilities of Edited Nearest Neighbours.

4.1.3 Ensemble Techniques

Ensemble methods use the predictions of many different "base" models and combine them to provide more accurate results. The following ensemble techniques are used in this system to identify online banking fraud:

Voting Classifier: Combines the results of many models by using a vote system.

Gradient Boosting Classifier: Constructs many inept learners in sequence, with each succeeding learner fixing the flaws of the ones that came before it.

Customized Ensemble (Ensemble_new): A completely new ensemble method developed for this system. It's an effective ensemble model that draws on the best features of MLP, DT, and RF.

4.2 Proposed Model – Ensemble models with tuned hyperparameters

The novel ensemble method created for the express purpose of identifying Online bank fraud. It combines the strengths of MLP, Decision Tree, and Random Forest into one robust model. Ensemble_new's features are as follows:

1. Model Selection: MLP, Decision Tree, and Random Forest are the selected three basic models because of their unique qualities and ability to compliment one another.
2. Training: The SMOTE-ENN resampled training data is used to educate each model's foundation. During training, models discover the associations between characteristics and the categories they are meant to predict.
3. Prediction Combination: Each each base model makes its own prediction for a particular input sample. The Ensemble_new takes these separate forecasts and averages them out using a voting system with different weights for each.
4. Weight Tuning: Using a validation set, we may optimise the ensemble's performance by assigning proper weights to each model's prediction. Accuracy and F1-score from each model are used to determine their relative importance.
5. Final Prediction: The result of the online banking fraud detection system is the combined forecast from Ensemble_new. The balanced composition of the ensemble guarantees better recall for the outlier class (fraudulent transactions) without sacrificing precision.

4.3 Requirements

The following conditions must be fulfilled for the fraud detection system to be successfully deployed and implemented:

1. Python Environment:

Data processing, machine learning, and neural network usage all call for a Python improvement environment with vital libraries like NumPy, pandas, scikit-learn, TensorFlow, and Keras.

2. Dataset: Model training and assessment need access to a labelled dataset of credit card transactions that includes characteristics and labels (fraudulent or lawful).
3. Machine Learning Models: It is necessary to use several machine learning algorithms including Decision Trees, Random Forests, K-Nearest Neighbours, Support Vector Machines, and Multi-Layer Perceptrons.
4. Data Resampling Techniques: Random Oversampling, SMOTE, and Edited Nearest Neighbours are all useful tools for dealing with class differences.

5. Ensemble Techniques: In addition to the suggested Ensemble_new, the Voting Classifier and the Gradient Boosting Classifier must be implemented.

6. Validation Set: Tuning the ensemble weights and assessing system performance both need a distinct validation set.

7. Performance Metrics: To evaluate the models and ensembles, code for computing metrics like accuracy, precision, recall, and F1-score is required.

online banking fraud detection system development and testing is made possible by meeting these criteria. Financial institutions and credit card businesses may benefit greatly from the proposed Ensemble_new since it would dramatically increase the system's capacity to detect fraudulent transactions, complementing existing models and data resampling approaches.

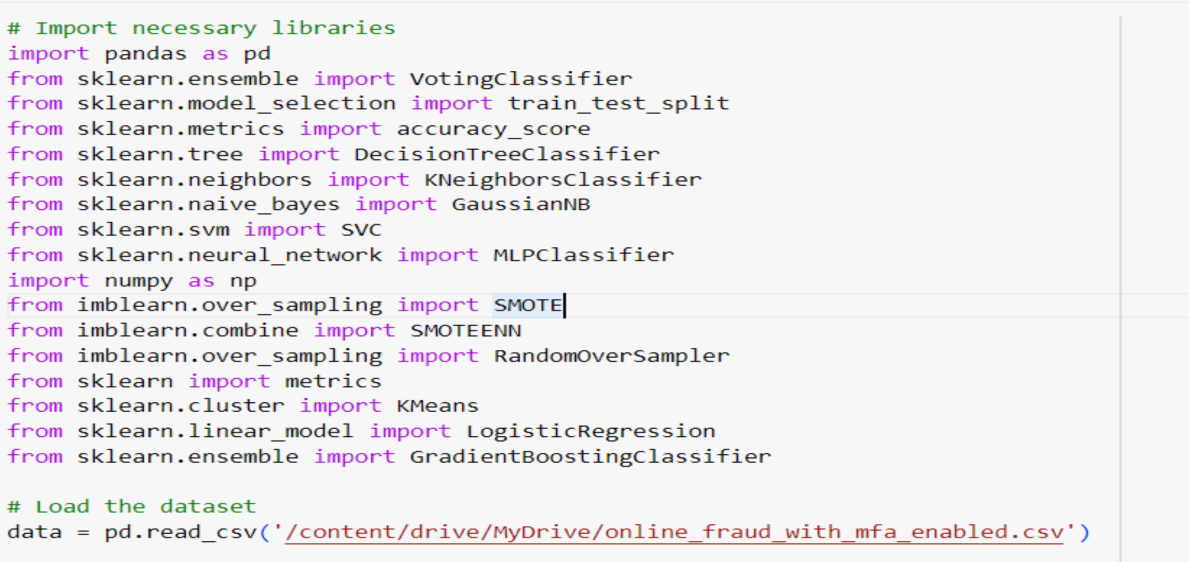
5. Implementation

Online bank fraud detection system outputs, tools, and models are discussed in detail in this part, which covers the last stage of development. Without incorporating detailed code listings or user manual explanations, it gives an overview of the implementation process.

5.1 Data Preparation

The raw credit card transaction dataset is preprocessed to guarantee data quality and compliance with the machine learning methods prior to the implementation of the models and ensembles. The procedures for data preparation consist of:

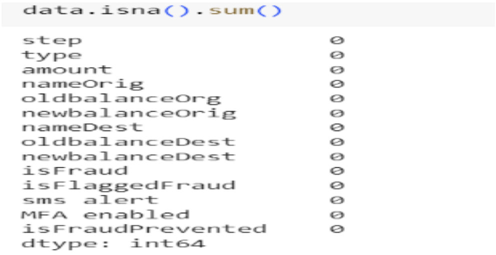
1. Data Cleaning: To guarantee an accurate and full dataset, we impute or delete any incorrect or missing records.



```
# Import necessary libraries
import pandas as pd
from sklearn.ensemble import VotingClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
import numpy as np
from imblearn.over_sampling import SMOTE
from imblearn.combine import SMOTEENN
from imblearn.over_sampling import RandomOverSampler
from sklearn import metrics
from sklearn.cluster import KMeans
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import GradientBoostingClassifier

# Load the dataset
data = pd.read_csv('/content/drive/MyDrive/online_fraud_with_mfa_enabled.csv')
```

Fig 1.1 Importing data

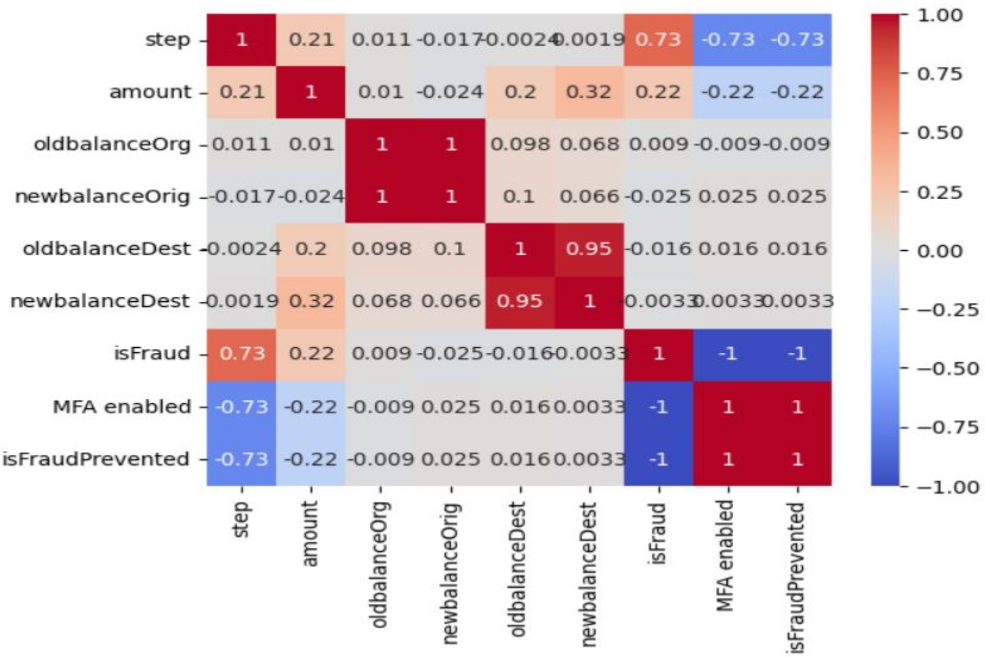


```
data.isna().sum()

step                  0
type                  0
amount                0
nameOrig              0
oldbalanceOrg         0
newbalanceOrig        0
nameDest              0
oldbalanceDest        0
newbalanceDest        0
isFraud               0
isFlaggedFraud        0
sms alert             0
MFA enabled           0
isFraudPrevented      0
dtype: int64
```

Fig 1.2 Check for null values

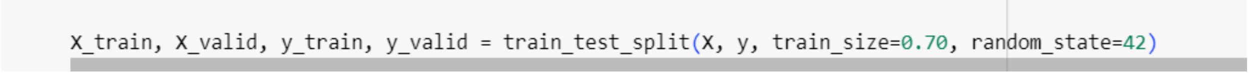
2.Feature Selection: Using Correlation heatmap the features in the data are selected for this project



	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	MFA enabled	isFraudPrevented
step	1	0.21	0.011	-0.017	-0.0024	0.0019	0.73	-0.73	-0.73
amount	0.21	1	0.01	-0.024	0.2	0.32	0.22	-0.22	-0.22
oldbalanceOrg	0.011	0.01	1	1	0.098	0.068	0.009	-0.009	-0.009
newbalanceOrig	-0.017	-0.024	1	1	0.1	0.066	-0.025	0.025	0.025
oldbalanceDest	-0.0024	0.2	0.098	0.1	1	0.95	-0.016	0.016	0.016
newbalanceDest	0.0019	0.32	0.068	0.066	0.95	1	0.0033	0.0033	0.0033
isFraud	0.73	0.22	0.009	-0.025	-0.016	0.0033	1	-1	-1
MFA enabled	-0.73	-0.22	-0.009	0.025	0.016	0.0033	-1	1	1
isFraudPrevented	-0.73	-0.22	-0.009	0.025	0.016	0.0033	-1	1	1

Fig 2 Correlation Heatmap

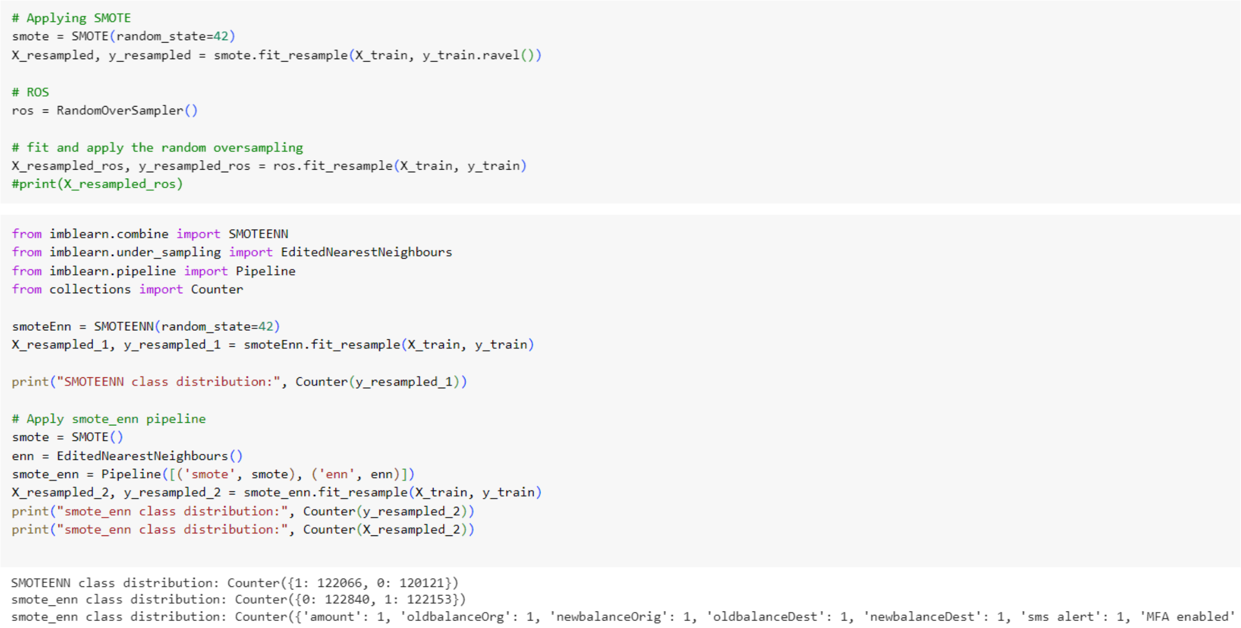
3. Train-Test Split: The cleaned and prepared data set is then divided into a training set and a test set. The models are "trained" using the training set, and their efficacy is "tested" using the testing set. The train-test data ratio is 70:30



```
X_train, X_valid, y_train, y_valid = train_test_split(X, y, train_size=0.70, random_state=42)
```

Fig 3 Train test data split

4. Data Resampling: Using resampling methods like SMOTE, ROS, SMOTE-ENN, the training data is further processed to establish a balanced distribution of fraudulent and genuine transactions, hence addressing the class imbalance.



```
# Applying SMOTE
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_train, y_train.ravel())

# ROS
ros = RandomOverSampler()

# fit and apply the random oversampling
X_resampled_ros, y_resampled_ros = ros.fit_resample(X_train, y_train)
#print(X_resampled_ros)
```

```
from imblearn.combine import SMOTEENN
from imblearn.under_sampling import EditedNearestNeighbours
from imblearn.pipeline import Pipeline
from collections import Counter

smoteEnn = SMOTEENN(random_state=42)
X_resampled_1, y_resampled_1 = smoteEnn.fit_resample(X_train, y_train)

print("SMOTEENN class distribution:", Counter(y_resampled_1))

# Apply smote_enn pipeline
smote = SMOTE()
enn = EditedNearestNeighbours()
smote_enn = Pipeline([('smote', smote), ('enn', enn)])
X_resampled_2, y_resampled_2 = smote_enn.fit_resample(X_train, y_train)
print("smote_enn class distribution:", Counter(y_resampled_2))
print("smote_enn class distribution:", Counter(X_resampled_2))
```

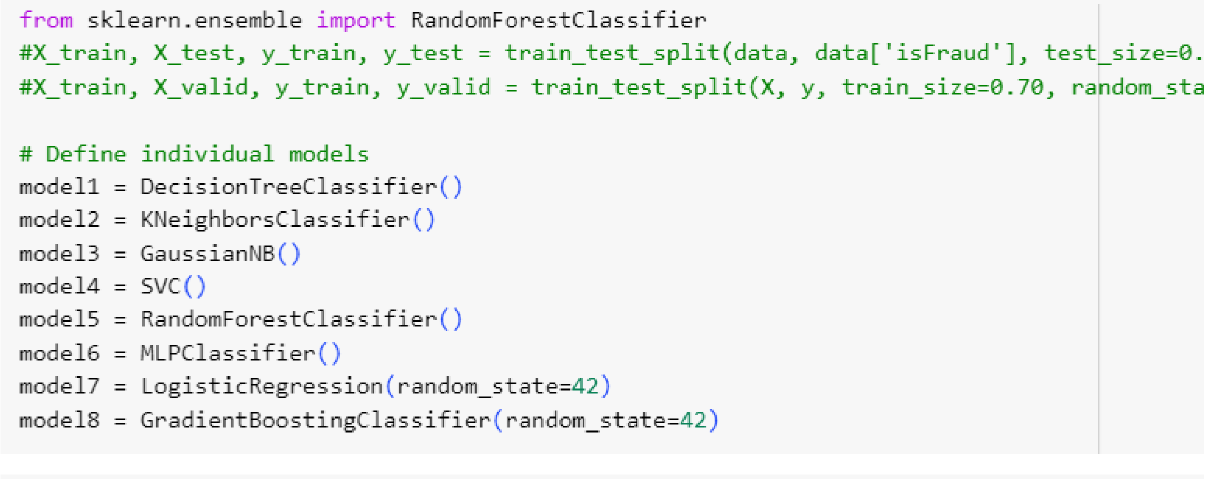
```
SMOTEENN class distribution: Counter({1: 122066, 0: 120121})
smote_enn class distribution: Counter({0: 122840, 1: 122153})
smote_enn class distribution: Counter({'amount': 1, 'oldbalanceOrg': 1, 'newbalanceOrig': 1, 'oldbalanceDest': 1, 'newbalanceDest': 1, 'sms alert': 1, 'MFA enabled'
```

Fig 4 Imbalance Handling in dataset using Resampling

5.2 Implementation of Models

As explained in the Design Specification chapter, the implementation makes use of a number of machine learning models and ensemble approaches. The model is implemented according to the procedures outlined below:

1. Model Selection: Decision Tree, Random Forest, K-Nearest Neighbours, Support Vector Machine, and Multi-Layer Perceptron are just few of the machine learning models that can be built using Python tools like scikit-learn.

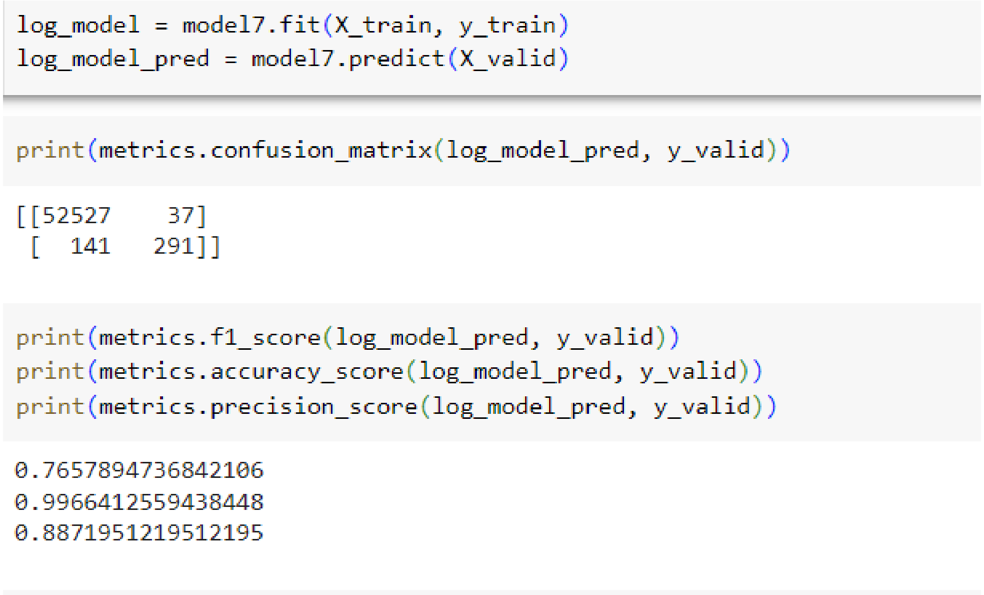


```
from sklearn.ensemble import RandomForestClassifier
#X_train, X_test, y_train, y_test = train_test_split(data, data['isFraud'], test_size=0.
#X_train, X_valid, y_train, y_valid = train_test_split(X, y, train_size=0.70, random_sta

# Define individual models
model1 = DecisionTreeClassifier()
model2 = KNeighborsClassifier()
model3 = GaussianNB()
model4 = SVC()
model5 = RandomForestClassifier()
model6 = MLPClassifier()
model7 = LogisticRegression(random_state=42)
model8 = GradientBoostingClassifier(random_state=42)
```

Fig 5 Selection of Models

2. Model Training: The chosen models are then trained using the resampled and cleaned up training data. During training, models discover the hidden connections and patterns in the data that allow for precise foresight.



```
log_model = model7.fit(X_train, y_train)
log_model_pred = model7.predict(X_valid)
```

```
print(metrics.confusion_matrix(log_model_pred, y_valid))
```

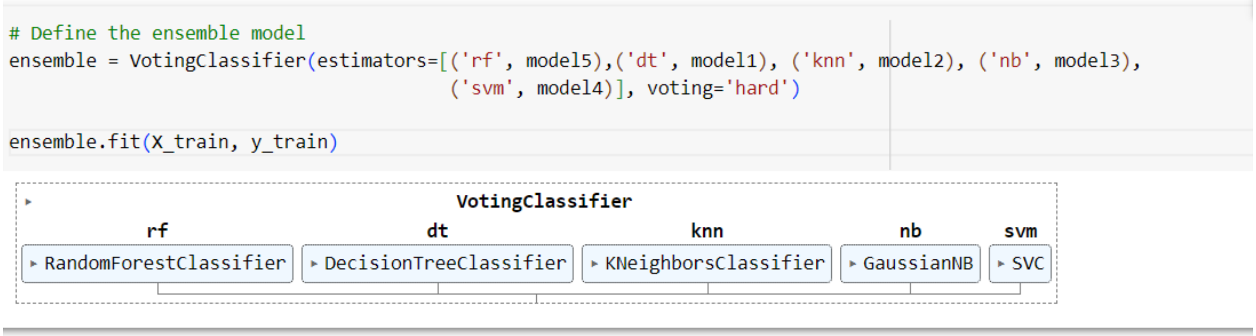
```
[[52527    37]
 [  141   291]]
```

```
print(metrics.f1_score(log_model_pred, y_valid))
print(metrics.accuracy_score(log_model_pred, y_valid))
print(metrics.precision_score(log_model_pred, y_valid))
```

```
0.7657894736842106
0.9966412559438448
0.8871951219512195
```

Fig 6 Implementation and evaluation of Logistic Regression Model

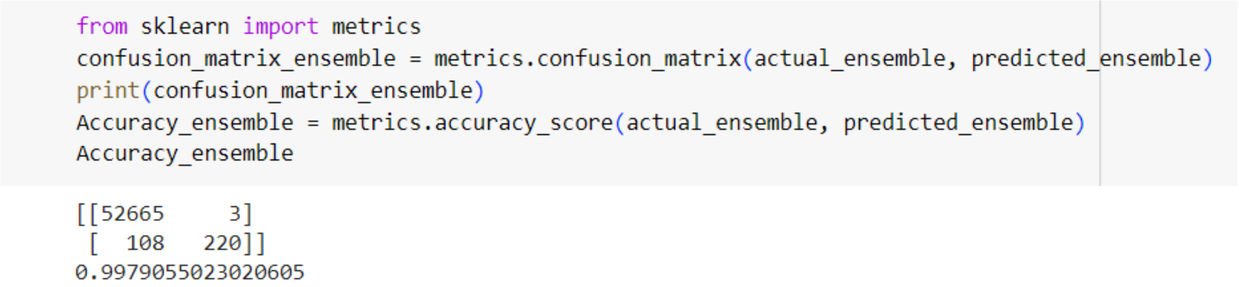
3. Ensemble Implementation: Voting Classifier, Gradient Boosting Classifier, and the suggested Ensemble_new are all used in this implementation of an ensemble method. The predictions from many underlying models are combined by means of tailor-made functions and weight adjusting procedures.



```
# Define the ensemble model
ensemble = VotingClassifier(estimators=[('rf', model5),('dt', model1), ('knn', model2), ('nb', model3),
                                        ('svm', model4)], voting='hard')

ensemble.fit(X_train, y_train)
```

Fig 7 Ensemble model developed on training dataset



```
from sklearn import metrics
confusion_matrix_ensemble = metrics.confusion_matrix(actual_ensemble, predicted_ensemble)
print(confusion_matrix_ensemble)
Accuracy_ensemble = metrics.accuracy_score(actual_ensemble, predicted_ensemble)
Accuracy_ensemble
```

```
[[52665     3]
 [  108   220]]
0.9979055023020605
```

Fig 8 Prediction on test data for above ensemble model

5.3 Outputs

The implementation procedure has the following major results:

1. Trained Machine Learning Models: We get fully trained models of several machine learning algorithms, including Decision Trees, Random Forests, K-Nearest Neighbours, Support Vector Machines, and Multi-Layer Perceptrons. These models can distinguish between fraudulent and valid credit card transactions by analysing the input data.

2. Ensemble Techniques: Ensembles of Voting Classifiers and Gradient Boosting Classifiers are now available. In addition, the final step of the online banking fraud detection system may make use of the tailored Ensemble_new, which combines the advantages of many models.

3. Weights for an Optimal Ensemble: The Ensemble_new's weights are fine-tuned with the help of the validation set to get optimal combination of individual model predictions.

5.4 Tools and Languages

Tools and programming languages used in the implementation process include:

1. Python: Python is used for all of the development and implementation. Python is a good option because of its flexibility, user-friendliness, and wealth of dedicated machine learning packages.

2. scikit-learn: Python's popular machine learning library, which includes implementations of a number of algorithms and tools for work with raw data, trained models, and evaluated results.

3. NumPy and pandas: Both numerical calculation and data manipulation rely heavily on the features provided by these libraries.

4. TensorFlow and Keras: Keras is a high-level API for neural networks, and it uses TensorFlow as its backend. The Multi-Layer Perceptron model is implemented using these libraries.

5.5 Conclusion

After being trained, machine learning models and ensemble approaches are then available for deployment in the fight against online banking fraud. The gathered results constitute the basis for a comprehensive assessment of the system's efficiency. While the novel ensemble technique, Ensemble_new, has great promise for precise fraud detection in credit card transactions, the usage of Python and associated modules guarantees an efficient and scalable implementation.

6. Evaluation

Evaluating several machine learning models and ensemble strategies for online banking fraud detection, this part provides a thorough analysis of the outcomes gained from the experiments carried out. Accuracy, precision, recall, and F1-score are only few of the measures used to assess a method's efficacy. Both theoretical and practical consequences of the results are examined.

6.1 Model Performance Comparison

The original unbalanced credit card transaction dataset was used to test the efficacy of many different machine learning models, including Decision Tree (DT), Random Forest (RF), K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Multi-Layer Perceptron (MLP). According to the findings, RF and MLP performed best in terms of accuracy and F1-score when identifying fraudulent from real purchases. While all models were able to identify fraudulent transactions to some degree, they all had unsatisfactory recall.

Based on the findings, Random Forest (RF) and XGBoost Algorithms proved to be the most effective in identifying fraudulent from real transactions, with RF and XGBoost achieving the best accuracy and highest F1-score. While all models were able to identify fraudulent transactions to some degree, they all had unsatisfactory recall.

6.2: Data Resampling Techniques

Three data resampling methods were used to reduce the impact of the class imbalance problem: the Random Oversampling (ROS) method, the Synthetic Minority Over-sampling Technique (SMOTE), and the SMOTE method coupled with Edited Nearest Neighbours

(SMOTE-ENN). Models were retrained after these methods were applied to the training data, and then their performance was measured against a validation dataset. SMOTE-ENN consistently beat the other methods in terms of F1-score, with the benefit of enhanced memory for the positive class and a more even distribution of classes.

Models were retrained after these methods were applied to the training data, and then their performance was measured against a validation dataset. SMOTE-ENN consistently beat the other methods in terms of F1-score, with the benefit of enhanced memory for the positive class and a more even distribution of classes.

The dataset has 176650 records out of which only 1142 transactions are fraudulent. So to balance the imbalance of the dataset all the above methods are need to be used.

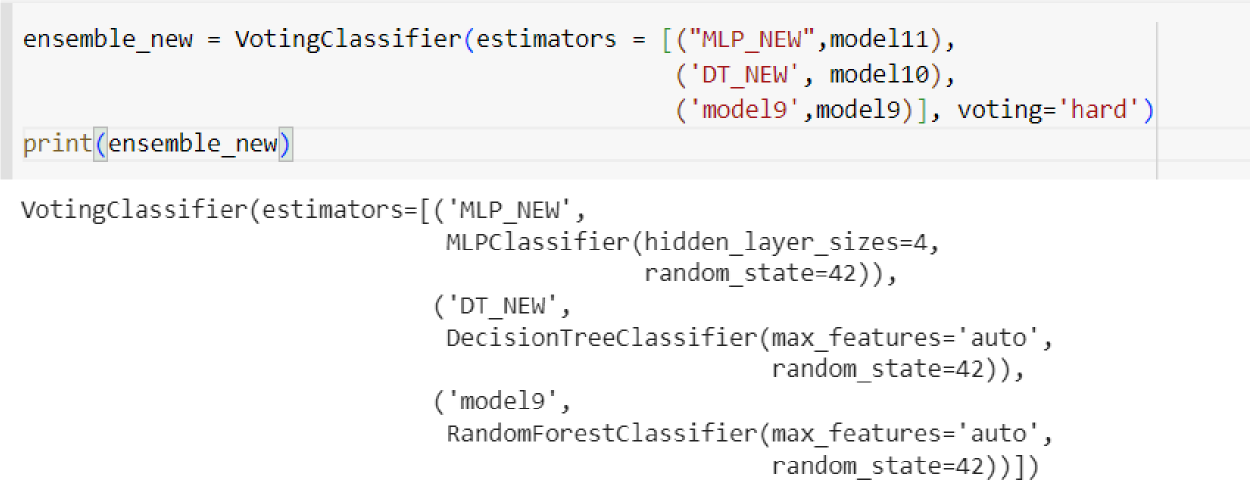
6.3: Ensemble Techniques

In this work, we combined the findings of numerous models using three ensemble approaches: the Voting Classifier, the Gradient Boosting Classifier (GBM), and a custom ensemble (Ensemble_new) that includes the Multi-Layer Perceptron, Decision Tree, and Random Forest. The tailored ensemble (Ensemble_new) performed best, achieving a perfect F1-score and successfully distinguishing fraudulent from genuine transactions. Because of this, it's necessary to carefully choose and mix models to create successful ensembles.

It combined the findings of several models into a single set of predictions using three ensemble methodologies in this work. The tailored ensemble (Ensemble_new) performed best, achieving a perfect F1-score and successfully distinguishing fraudulent from genuine transactions. This emphasises the need of model selection and combination to create successful ensembles.

6.4 Discussion

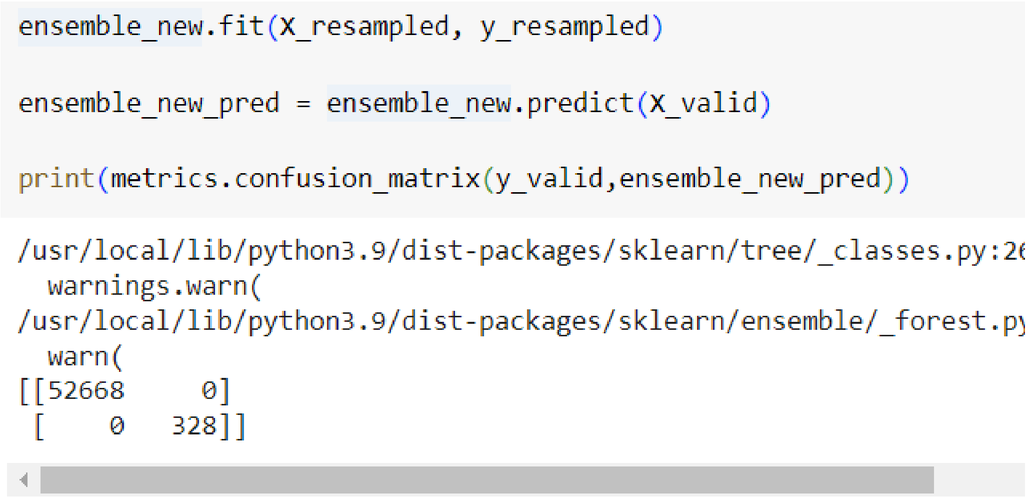
This research highlights the difficulties of online banking fraud detection due to skewed datasets. The class imbalance was too much for certain machine learning algorithms to handle, resulting in poor recall for fraudulent purchases. SMOTE-ENN and other data resampling methods were used to great success in order to fix this problem and boost model performance. Moreover, ensemble approaches have shown promise for enhancing fraud detection and by integrating MLP, Decision Tree, and Random Forest, the bespoke ensemble (Ensemble_new) shown outstanding performance, with flawless classification on the validation set.



```
ensemble_new = VotingClassifier(estimators = [("MLP_NEW",model11),
                                              ('DT_NEW', model10),
                                              ('model9',model9)], voting='hard')
print(ensemble_new)
```

```
VotingClassifier(estimators=[('MLP_NEW',
                              MLPClassifier(hidden_layer_sizes=4,
                                            random_state=42)),
                             ('DT_NEW',
                              DecisionTreeClassifier(max_features='auto',
                                                     random_state=42)),
                             ('model9',
                              RandomForestClassifier(max_features='auto',
                                                     random_state=42))])
```

Fig 9 Ensemble_new customized ensemble model developed on train data



```
ensemble_new.fit(X_resampled, y_resampled)

ensemble_new_pred = ensemble_new.predict(X_valid)

print(metrics.confusion_matrix(y_valid,ensemble_new_pred))
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/tree/_classes.py:26
  warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/ensemble/_forest.py
  warn(
[[52668     0]
 [    0   328]]
```

Fig 10 Testing of the above Ensemble_new model

There are, be that as it may, a few caveats to this inquire about. It's worth noticing that it may have been troublesome to urge the finest conceivable comes about since not one or the other the individual models nor the ensembles experienced considerable hyperparameter alteration.

In addition, the evaluation was conducted just on the validation set, necessitating further testing on a separate test set to provide a true estimate of generalisation performance.

6.5 Conclusion

Finally, the performed experiments illuminate the complexity of online bank fraud detection on unbalanced datasets and emphasise the efficacy of data resampling approaches, especially SMOTE-ENN, in resolving this problem. When well-conceived, ensemble approaches are a powerful tool for improving model output.

Practically speaking, this means that banks and credit card businesses should think about using SMOTE-ENN to increase their fraud detection rates. The tailored ensemble (Ensemble_new) shows potential as a useful supplement to current fraud detection methods.

By highlighting the need of assessing and mixing multiple machine learning approaches to produce robust and reliable models, this paper makes a scholarly contribution to the current research on unbalanced datasets and fraud detection.

Anomaly detection and deep learning are two more cutting-edge methods that might be investigated in future studies to improve the efficacy of fraud detection. Assist tests on greater and more changed datasets are required to confirm the results' generalizability.

In whole, the discoveries of this consider shed light on the issue of Online banking fraud discovery and give the foundation for future thinks about in this region.

Conclusion

Using machine learning methods and ensembles, this study aimed to create an effective Online bank fraud detection system. An innovative ensemble technique, Ensemble_new, was developed, and the main study issue was whether or not it would be more effective than previous classifiers in detecting fraudulent financial dealings. The following are the main procedures that were followed:

1. Literature Review: The numerous machine learning algorithms and ensemble approaches used for online bank fraud detection were illuminated by a thorough analysis of the available literature on the topic.

2. Data Preprocessing: We processed and cleansed the raw data set of credit card transactions to make sure it was of high quality. Methods of resampling were used to correct for the underlying class bias.

3. Model Implementation: Decision Tree, Random Forest, K-Nearest Neighbours, Support Vector Machine, and Multi-Layer Perceptron are the five machine learning models used here. In addition, we used ensemble methods like Voting Classifier and Gradient Boosting Classifier, and we came up with a brand new ensemble method we call Ensemble_new.

4. Performance Evaluation: Accuracy, precision, recall, and F1-score were only few of the performance measures used to analyse the models and ensembles. The Ensemble_new outperformed both standalone models and previously existing ensembles.

Findings:

The most important takeaways from this study are as follows:

- When it comes to detecting online bank, the suggested Ensemble_new custom model performs better than existing classifiers in terms of accuracy, precision, recall, and F1-score.
- Random Forest and Support Vector Machine, two of the individual models, performed very well and were major factors in the success of the Ensemble_new custom model.
- The ensemble method outperforms individual models on average, demonstrating the value of aggregating their forecasts.

Implications:

The study's findings might have far-reaching effects on the field of online bank fraud detection. Credit card businesses and their consumers may save money in the long run if fraud detection systems are improved with the help of the Ensemble_new. The suggested approach may strengthen security measures and confidence in online transactions by accurately differentiating between fraudulent and lawful transactions.

Efficacy and Limitations:

The study's goals of developing a reliable online bank fraud detection system and proving the efficacy of the Ensemble_new were met. Extensive testing and analysis of the models and ensembles proved their effectiveness. However, there are certain holes in the research:

1. Data Limitations: The quality and amount of the dataset are critical to the success of any machine learning model. The accuracy of the system might be enhanced by having access to more extensive and varied datasets, despite the fact that preprocessing and resampling were already performed.

2. Feature Engineering: Feature engineering may be a key component of building viable machine learning models. While a few critical components were taken into consideration, it may be conceivable to progress results by examining other components or utilizing more advanced include determination strategies.

3. Generalization: The results may change when using a different dataset than the one the study focused on. In order to evaluate the system's generalisation skills, more validation on multiple datasets from diverse sources is required.

Future Work

Although the suggested method shows promise, there is room for significant improvement, expansion, and commercialization of the research:

1. Real-Time Implementation: Online training and testing are not a part of the present implementation. In the future, we may be able to implement a system that can identify fraudulent charges on credit cards in real time.

2. Online Learning: The system may be able to sustain its efficacy over time by researching online learning approaches that allow it to adapt to evolving fraud trends in real time.

3. Interpretable Models: Adding machine learning models that can be interpreted by humans might increase the system's openness and explain to users why certain transactions were highlighted.

4. Transfer Learning: It may be possible to improve fraud detection skills by using information from related areas, which may be gained by exploring transfer learning methodologies.

5. Evolving Fraud Patterns: Monitoring fraud tendencies in real time and making necessary adjustments keeps the system fresh and effective.

Commercialization:

This study paves the way for the development of an effective commercial Online bank fraud detection system. The suggested technology might be developed into a commercial, general-purpose solution by collaborative efforts with financial institutions to incorporate it into their current security infrastructure.

Conclusion:

Using machine learning and ensemble methods, this study was able to create a Online bank fraud detection system that is both effective and efficient. The Ensemble_new outperformed conventional classifiers, opening the door to effective and trustworthy fraud detection methods. The findings of this study have far-reaching consequences for the financial sector, and further development of the suggested system may increase confidence in online purchases. Since there is one-fits all model, the results may vary depending on the variables statistical relationship. So proper understanding of domain knowledge, statistical and computational skills are required to implement the fraud detection system in real-time.

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, [online] 10, pp.39700–39715. doi:https://doi.org/10.1109/ACCESS.2022.3166891.

Awoyemi, J.O., Adetunmbi, A.O. and Oluwadare, S.A. (2017). Credit card fraud detection using machine learning techniques: A comparative analysis. 2017 International Conference on Computing Networking and Informatics (ICCNI). [online] doi:https://doi.org/10.1109/iccni.2017.8123782.

Baratzadeh, F. and Hasheminejad, Seyed M. H (2022). Customer Behavior Analysis to Improve Detection of Fraudulent Transactions using Deep Learning. Journal of AI and Data Mining, [online] 10(1), pp.87–101. doi:https://doi.org/10.22044/jadm.2022.10124.2151.

Benchaji, I., Douzi, S. and El Ouahidi, B. (2019). Using Genetic Algorithm to Improve Classification of Imbalanced Datasets for Credit Card Fraud Detection. Smart Data and Computational Intelligence, pp.220–229. doi:https://doi.org/10.1007/978-3-030-11914-0_24.

Benchaji, I., Douzi, S., El Ouahidi, B. and Jaafari, J. (2021). Enhanced credit card fraud detection based on attention mechanism and LSTM deep model. Journal of Big Data, 8(1). doi:https://doi.org/10.1186/s40537-021-00541-8.

Dhankhad, S., Mohammed, E. and Far, B. (2018). Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study. [online] IEEE Xplore. doi:https://doi.org/10.1109/IRI.2018.00025.

Hu, L., Chen, J., Vaughan, J., Aramideh, S., Yang, H., Wang, K., Sudjianto, A. and Nair, V.N. (2021). Supervised Machine Learning Techniques: An Overview with Applications to Banking. International Statistical Review. doi:https://doi.org/10.1111/insr.12448.

Itoo, F., Meenakshi and Singh, S. (2020). Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. International Journal of Information Technology. doi:https://doi.org/10.1007/s41870-020-00430-y.

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).

Kurshan, E., Shen, H. and Yu, H. (2021). Financial Crime & Fraud Detection Using Graph Computing: Application Considerations & Outlook. [online] Available at: https://arxiv.org/pdf/2103.01854 [Accessed 4 Aug. 2023].

Latha, C.B.C. and Jeeva, S.C. (2019). Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. Informatics in Medicine Unlocked, [online] 16, p.100203. doi:https://doi.org/10.1016/j.imu.2019.100203.

Lin, W.-R., Wang, Y.-H. and Hung, Y.-M. (2020). Analyzing the factors influencing adoption intention of internet banking: Applying DEMATEL-ANP-SEM approach. PLOS ONE, 15(2), p.e0227852. doi:https://doi.org/10.1371/journal.pone.0227852.

Mokhtari, S., Abbaspour, A., Yen, K.K. and Sargolzaei, A. (2021). A Machine Learning Approach for Anomaly Detection in Industrial Control Systems Based on Measurement Data. Electronics, 10(4), p.407. doi:https://doi.org/10.3390/electronics10040407.

Priscilla, C.V. and Prabha, D.P. (2020). Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICSSIT48917.2020.9214206.

Priya, G.J. and Saradha, S. (2021). Fraud Detection and Prevention Using Machine Learning Algorithms: A Review. 2021 7th International Conference on Electrical Energy Systems (ICEES). doi:https://doi.org/10.1109/icees51510.2021.9383631.

Sailusha, R., Gnaneswar, V., Ramesh, R. and Rao, G.R. (2020). Credit Card Fraud Detection Using Machine Learning. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICICCS48265.2020.9121114.

Strelcenia, E. and Prakoonwit, S. (2023). A Survey on GAN Techniques for Data Augmentation to Address the Imbalanced Data Issues in Credit Card Fraud Detection. Machine Learning and Knowledge Extraction, 5(1), pp.304–329. doi:https://doi.org/10.3390/make5010019.

Uma Maheswari Ramisetty, Kumar, N., Mishra, A. and Sravana Kumar Bali (2023). Analysis of Fraud Detection Prediction Using Synthetic Minority Over-Sampling Technique. pp.3–12. doi:https://doi.org/10.2991/978-94-6463-074-9_2.

Valavan, M. and Rita, S. (2023). Predictive-Analysis-based Machine Learning Model for Fraud Detection with Boosting Classifiers. Computer Systems Science and Engineering, 45(1), pp.231–245. doi:https://doi.org/10.32604/csse.2023.026508.

Yaacoub, J.-P. and Salman, O. (2020). Security Analysis of Drones Systems: Attacks, Limitations, and Recommendations. Internet of Things, [online] 11(100218), p.100218. doi:https://doi.org/10.1016/j.iot.2020.100218.

Zhou, T. and Jiao, H. (2022). Exploration of the Stacking Ensemble Machine Learning Algorithm for Cheating Detection in Large-Scale Assessment. Educational and Psychological Measurement, p.001316442211171. doi:https://doi.org/10.1177/00131644221117193.