CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING

INTERNSHIP PROJECT REPORT

Submitted to

MSME – TECHNOLOGY CENTRE
TOOL ROOM & TRAINING CENTRE, GUWAHATI

Submitted by

SAGAR CHOUDHURY	**210310007045**
SOHINI ROY	**210310007051**
LAKHYAJYOTI DOLEY	**220350007003**
RIMJIM TALUKDAR	**220350007005**
CHINMOY SARMAH	**210310007011**

Department of Computer Science & Engineering

Girijananda Chowdhury Institute of Management & Technology
(GIMT), Guwahat

BONAFIDE CERTIFICATE

This is to certify that this project report entitled **"CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING"** submitted to **Tool Room & Training Centre**, Guwahati Amingaon is a bonafide record of work done by **"SAGAR CHOUDHURY, SOHINI ROY, RIMJIM TALUKDAR, LAKHYAJYOTI DOLEY, CHINMOY SARMAH"** under my supervision from "10-6-2024" to "10-7-2024".

Ankan Sir
CS/IT Faculty

Place
Date

DECLARATION BY AUTHOR(S)

This is to declare that this report has been written by us. No part of the report is plagiarized from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be plagiarized, we are shall take full responsibility for it.

Sagar Choudhury
210310007045

Chinmoy Sarmah
210310007011

Sohini Roy
210310007051

Lakhyajyoti Doley
2203510007003

Rimjhim Talukdar
2203510007005

Place Guwahati
Date 8-7-2024

ABSTRACT

This project report presents a comprehensive study on credit card fraud detection using machine learning. The primary objective is to compare various machine learning algorithms and evaluate their performance to identify the most effective model for detecting fraudulent transactions. The study encompasses essential machine learning practices, including data visualization, preprocessing, and sampling methods, to address the challenges posed by an imbalanced dataset.

The dataset is pre-processed using the Standard Scaler to normalize the features, and oversampling techniques from the imbalanced-learn (imblearn) library are employed to mitigate the imbalance in the dataset. Visualization tools such as Seaborn and Matplotlib are utilized to provide insights into the data distribution and feature relationships.

The machine learning algorithms evaluated in this study include logistic regression, decision trees and random forests. Each model's performance is assessed based on key metrics such as accuracy, precision, recall and F1-score. The optimal findings are saved and managed using the Pickle library for model persistence and reproducibility.

The results indicate significant differences in the models' ability to detect fraud, with some algorithms demonstrating superior performance metrics. This comparative analysis provides valuable insights into selecting the most effective machine learning model for credit card fraud detection, contributing to the enhancement of security measures in financial transactions. Ultimately, the Random Forest model demonstrated superior performance and was selected as the optimal model for fraud detection.

To facilitate practical application, a Streamlit web application was developed, enabling users to input transaction details and receive real-time predictions on whether a transaction is fraudulent or normal. This project not only highlights the importance of preprocessing and handling imbalanced data but also demonstrates the practical implementation of machine learning models in a user-friendly interface for fraud detection.

TABLE OF CONTENTS

1. ABSTRACT 1
2. CHAPTER I: INTRODUCTION 3
3. CHAPTER II: DATASET 5
4. CHAPTER III: PRE-REQUISITE INFORMATION 7
5. CHAPTER IV: METHODOLOGY
    a. DATA PRE-PROCESSING 8
    b. DETERMINING THE DATASET AS IMBALANCED 8
    c. TRAIN TEST SPLIT AND MODEL TRAINING 9
    d. UNDER-SAMPLING 11
    e. OVER-SAMPLING 12
6. CHAPTER V: CONCLUSION 14
7. REFERNCES 15

CHAPTER I: INTRODUCTION

In today's digital era, the exponential growth of online transactions has simultaneously increased the vulnerability to credit card fraud. Financial institutions and consumers alike face significant challenges as traditional fraud detection methods struggle to keep pace with the sophistication and volume of fraudulent activities. As cybercriminals develop more advanced techniques, there is a pressing need for more robust and intelligent solutions to detect and prevent fraud effectively.

Machine learning (ML) has revolutionized the field of cybersecurity by providing powerful tools to analyse vast amounts of data, recognize patterns, and detect anomalies indicative of fraudulent activities. The integration of ML in cybersecurity enables organizations to identify and respond to threats more swiftly and accurately than traditional methods. This project focuses on harnessing the potential of machine learning to enhance credit card fraud detection, ensuring the security and trust of financial transactions.

The primary objective of this project is to evaluate and compare various machine learning algorithms to determine the most effective model for detecting credit card fraud. The study encompasses several critical steps, including data preprocessing, visualization, and handling imbalanced datasets—a common challenge in fraud detection. Key libraries such as Scikit-learn, StandardScaler, Seaborn, Matplotlib, Pickle, and Imbalanced-learn's over-sampling techniques are employed to preprocess and visualize the data, as well as to address the imbalance in the dataset.

Data imbalance poses a significant challenge in training effective fraud detection models, as fraudulent transactions are typically far less frequent than legitimate ones. To mitigate this issue, various sampling methods are applied to balance the dataset, enhancing the accuracy and reliability of the machine learning models. Through meticulous evaluation, several algorithms are compared, with their performance metrics analysed to identify the most promising approach.

After thorough analysis, the Random Forest model emerges as the superior algorithm for credit card fraud detection, demonstrating excellent performance in identifying fraudulent transactions. To translate this model into a practical tool, a Streamlit web application is developed. This application allows users to input transaction details and receive real-time predictions on the likelihood of fraud, making the technology accessible and user-friendly for end-users.

The significance of machine learning in cybersecurity extends beyond fraud detection to encompass a wide array of applications, including threat detection, anomaly detection, and intrusion prevention. By automating the processing and analysis of extensive datasets, ML empowers organizations to proactively address security threats, thereby enhancing overall cybersecurity measures.

In conclusion, this project underscores the transformative impact of machine learning in the domain of credit card fraud detection and cybersecurity at large. By implementing a robust ML model and a practical Streamlit application, the project not only improves the detection of fraudulent transactions but also illustrates the broader potential of ML in safeguarding digital environments. This work highlights the critical role of advanced technologies in maintaining the integrity and security of financial transactions, ultimately fostering greater confidence in the digital economy.

CHAPTER II : DATASET

Context:

Credit card companies must recognize fraudulent transactions to prevent customers from being charged for items they did not purchase. This task is crucial for maintaining customer trust and minimizing financial losses.

Content:

The dataset contains credit card transactions made by European cardholders in September 2013. It includes transactions from a two-day period, consisting of 284,807 transactions, out of which 492 are fraudulent. The dataset is highly unbalanced, with fraudulent transactions constituting only 0.172% of all transactions.

Features:

- The dataset includes only numerical input variables derived from a Principal Component Analysis (PCA) transformation.
- Due to confidentiality, the original features and more background information about the data are not provided.
- Features V1, V2, … V28 are the principal components obtained through PCA.
- The features 'Time' and 'Amount' have not been transformed with PCA.
  - Time: Represents the seconds elapsed between each transaction and the first transaction in the dataset.
  - Amount: Represents the transaction amount, which can be used for example-dependent cost-sensitive learning.
- Class: The response variable that indicates whether a transaction is fraudulent (1) or not (0).

Imbalanced Data:

Given the class imbalance ratio, traditional confusion matrix accuracy is not meaningful for unbalanced classification.

Data Source:

The dataset was collected and analyzed during a research collaboration between Worldline and the Machine Learning Group of ULB (Université Libre de Bruxelles) on big data mining and fraud detection. More information can be found at MLG ULB.

This dataset provides a practical framework for developing and testing machine learning models aimed at detecting credit card fraud, offering valuable insights into handling imbalanced datasets and applying PCA-transformed features for classification tasks.

CHAPTER III: PRE-REQUISITE INFORMATION

3.1 SOFTWARE AND HARDWARE REQUIREMENT

Hardware:

- OS – Windows 7, 8 and 10 (32 and 64 bit)
- RAM – 4GB

Software:

- Python
- Jupyter notebook & lab

3.2 SYSTEM ARCHITECTURE

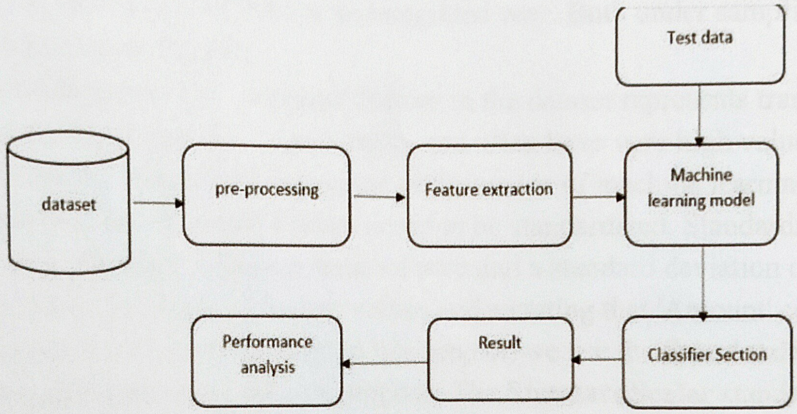

Fig 1 System Architecture

3.3 PACKAGES

Which are being used for data exploration, pre-processing and for using random forest algorithm are:

- **NumPy**: For simple arrays.
- **Pandas**: For reading the file.
- **SciKit Learn**- for pre-processing.
- **Matplotlib & Seaborn**: For plotting and representing confusion matrix colour format.
- **Pickle**: For picking out the model for further use.
- **Imbalanced learn**: For over-sampling.

CHAPTER IV: METHODOLOGY

4.1 DATA PRE-PROCESSING

Pre-processing is the process of three important and common steps as follows:

- **Formatting**: It is the process of putting the data in a legitimate way that it would be suitable to work with. Format of the data files should be formatted according to the need. Most recommended format is .csv files.
- **Cleaning**: Data cleaning is a very important procedure in the path of data science as it constitutes the major part of the work. It includes removing missing data and complexity with naming category and so on. Here the "Time" feature is not used , hence it is droped from the dataset.
- **Sampling**: This is the technique of analyzing the subsets from whole large datasets, which could provide a better result and help in understanding the behavior and pattern of data in an integrated way. Both under sampling and over sampling are performed.
- **Standardization**: The '**Amount**' feature in the dataset represents transaction amounts, which can vary significantly and often have very high values. Such variations can negatively impact the performance of machine learning models.To address this, the 'Amount' feature needs to be standardized. Standardization transforms the data to have a mean of zero and a standard deviation of one, normalizing the range of feature values and ensuring that 'Amount' contributes appropriately to model training.In this project, we use the **StandardScaler** from **sklearn.preprocessing** for this purpose. The **StandardScaler** standardizes features by removing the mean and scaling to unit variance. This preprocessing step improves the performance and convergence speed of many machine learning algorithms.

4.2 DETERMINING THE DATASET AS IMBALANCED

An imbalanced dataset is a type of dataset where the classes are not represented equally. This means that one class significantly outnumbers the other class or classes. In the context of binary classification, an imbalanced dataset typically has a majority class and a minority class.

The "**Class**" feature of the dataset is checked for a value count of its categorical data points. The distribution is represented in a 'ggplot' graph using **matplotlib & seaborn**.

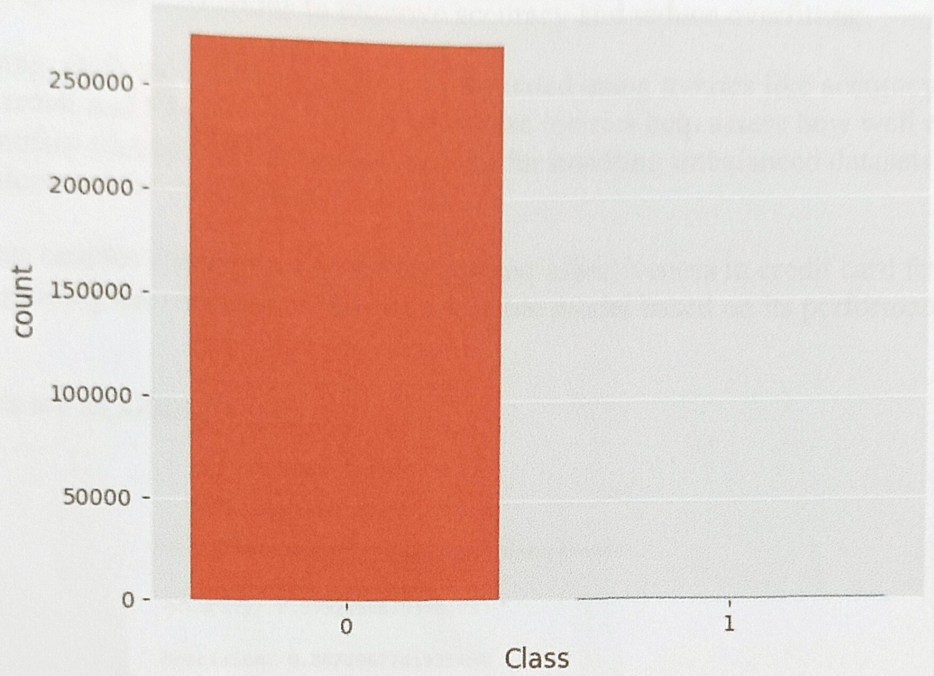

Fig 2. Value distribution of data points of 'Class'

As the figure suggests, the value distribution of the "0" i.e. Normal Transactions is far more then the "1" i.e. Fraudulent Transactions . This makes the dataset imbalanced and calls for some sampling of this feature for proper predictions.

4.3 TRAIN-TEST SPLIT AND MODEL TRAINING

The dataset is split into training and testing sets using the train_test_split function from sklearn.model_selection, with 80% allocated for training and 20% for testing. This approach ensures that models are trained on a subset of data and evaluated on unseen data to assess their generalization ability.

Three machine learning algorithms—Logistic Regression, Decision Tree, and Random Forest—are trained on the training set:

- **Logistic Regression:** A linear model that calculates probabilities and makes binary classifications based on a threshold.

- **Decision Tree:** Constructs a tree-like model by recursively splitting data based on feature values.
- **Random Forest:** An ensemble method that builds multiple decision trees and averages their predictions to improve accuracy and reduce overfitting.

After training, each model's performance is evaluated using metrics like accuracy, precision, recall and F1-scoreon the test set. These metrics help assess how well each model identifies fraudulent transactions, crucial for handling imbalanced datasets where accuracy alone can be misleading.

This process enables comparison of the algorithms' effectiveness in credit card fraud detection, guiding the selection of the most suitable model based on its performance metrics.

The metrics are as follows:

```
***********LogisticRegression*************
 Accuracy: 0.9992200678359603
 Precision: 0.8870967741935484
 Recall: 0.6043956043956044
 f1 score : 0.718954248366013
***********Decision Tree Classifier*************
 Accuracy: 0.9989479984764116
 Precision: 0.6774193548387096
 Recall: 0.6923076923076923
 f1 score : 0.684782608695652
***********Random Forest Classifier*************
 Accuracy: 0.9994014474089928
 Precision: 0.8918918918918919
 Recall: 0.7252747252747253
 f1 score : 0.8
```

Fig 3. Metrics before sampling

4.5 UNDER-SAMPLING

Under-sampling is a technique used to address the imbalance between classes in a dataset, particularly prevalent in scenarios like credit card fraud detection where fraudulent transactions (minority class) are significantly outnumbered by legitimate transactions (majority class).

Here under-sampling is performed to improve the precision, recall, F1 score of the classifiers. A sample of the majority class (Normal Transaction) of the same number of the minority class count.(i.e. 473) and then concatenated with the minority class.

```
#Undersampling

normal = data[data['Class']==0]
fraud = data[data['Class']==1]

normal.shape

(275190, 30)

fraud.shape

(473, 30)

normal_sample = normal.sample(n=473)

normal_sample.shape

(473, 30)

data1 = pd.concat([normal_sample,fraud], ignore_index = True)
```

Fig 4. Under-sampling

Now the metrics of all the classifiers are checked again after training the new dataset are:

```
**********LogisticRegression*************
 Accuracy: 0.9157894736842105
 Precision: 0.93
 Recall: 0.9117647058823529
 f1 score : 0.9207920792079208
**********Decision Tree Classifier*************
 Accuracy: 0.9052631578947369
 Precision: 0.9038461538461539
 Recall: 0.9215686274509803
 f1 score : 0.9126213592233009
**********Random Forest Classifier*************
 Accuracy: 0.9210526315789473
 Precision: 0.9393939393939394
 Recall: 0.9117647058823529
 f1 score : 0.9253731343283583
```

Fig 5. Metrics after Under-sampling

However, it carries the risk of information loss, where removing samples may lead to a dataset that no longer accurately represents the true population distribution. This can diminish the model's ability to generalize to new data and may bias it towards the retained instances, potentially overlooking important patterns or rare events such as fraudulent transactions in credit card fraud detection. Hence we use Over-sampling.

4.6 OVER-SAMPLING

Oversampling is a technique used in machine learning to address class imbalance by increasing the number of instances in the minority class to match the majority class. This approach is particularly useful in scenarios like credit card fraud detection, where fraudulent transactions (minority class) are significantly fewer than legitimate transactions (majority class). Here techniques like Synthetic Minority Over-sampling Technique (SMOTE) create synthetic instances of the minority class by interpolating between existing instances.

```
[54]: #oversampling

[55]: X = data.drop('Class', axis = 1)
      y = data['Class']

[57]: X.shape

[57]: (275663, 29)

[58]: y.shape

[58]: (275663,)

[64]: from imblearn.over_sampling import SMOTE

[65]: X_res, y_res = SMOTE().fit_resample(X,y)

[66]: y_res.value_counts()

[66]: Class
      0    275190
      1    275190
      Name: count, dtype: int64

[71]: X_train, X_test, y_train, y_test = train_test_split(X_res, y_res, test_size = 0.2, random_state = 42 )
```

Fig 6. Over-Sampling

Now the metrics of all the classifiers are checked again after training the new dataset are:

**********LogisticRegression**********

Accuracy: 0.944383880228206

Precision: 0.973295377524739

Recall: 0.9137683399087323

f1 score : 0.9425929746253822

**********Decision Tree Classifier**********

Accuracy: 0.9983375122642538

Precision: 0.9978567276954373

Recall: 0.9988182462774757

f1 score : 0.9983372554720649

**********Random Forest Classifier**********

Accuracy: 0.9999273229405138

Precision: 0.9998545745396376

Recall: 1.0

f1 score : 0.9999272819822931

Fig 7. Metrics after Over-sampling

CHAPTER V: CONCLUSION

Hence, we have acquired the result of an accurate value of credit card fraud detection i.e 0.9999273229405138 (99.93%) using a random forest algorithm with new enhancement s. In comparison to existing modules, this proposed module is applicable for the larger d ataset and provides more accurate results. The Random forest algorithm will provide bet ter performance with many training data, but speed during testing and application will st ill suffer. Usage of more pre-processing techniques would also assist

Our future work will try to represent this into a software application and provide a soluti on for credit card fraud using the new technologies like Machine Learning, Artificial Int elligence and Deep Learning.

REFERNCES

1. https://towardsdatascience.com/the-random-forestalgorithm-d457d499ffcd
2. https://www.xoriant.com/blog/productengineering/decision-trees-machine-learningalgorithm.html
3. Gupta, Shalini, and R. Johari. "A New Framework for Credit Card Transactions Involving Mutual Authentication between Cardholder and Merchant." International Conference on Communication Systems and Network Technologies IEEE, 2021:22-26.
4. Y. Gmbh and K. G. Co, "Global online payment methods: the Full year 2020," Tech. Rep., 3 2020.
5. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data