**Online Transaction Fraud Detection System**

**A Project Report Submitted in Partial Fulfilment of**

**the Requirements for the Degree of**

**Bachelor of Technology**

**in**

**Computer Science & Engineering**

**by**

**Aman Kumar Singh 2003480100010**

**Mahima Singh 2003480100044**

**Nikhil Gupta 2003480100051**

**Shakya Ekta Maurya 2003480100069**

**Shruti Sharma 2003480100076**

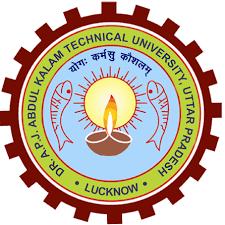
**Under the Supervision of**

**Mr. Abhay Kumar Tripathi**

**(Assistant Professor)**

**PSIT COLLEGE OF ENGINEERING, KANPUR**

**to the**

****

**Faculty of Computer Science & Engineering**

**Dr. A.P.J. Abdul Kalam Technical University, Lucknow**

**(Formerly Uttar Pradesh Technical University)**

**May 2024**

# DECLARATION

I declare that this submission is solely my work, and to the best of my knowledge and belief, it does not include any previously published or written material by another person. Furthermore, it does not contain any material that has substantially contributed to the award of any degree or diploma from a university or other institute of higher learning, except where proper acknowledgment has been provided within the text.

Name : Aman Kumar Singh

Roll No : 2003480100010

Date :

Signature :

Name : Mahima Singh

Roll No : 2003480100044

Date :

Signature :

Name : Nikhil Gupta

Roll No : 2003480100051

Date :

Signature :

Name : Shakya Ekta Maurya

Roll No : 2003480100069

Date :

Signature :

Name : Shruti Sharma

Roll No : 2003480100076

Date :

Signature :

# ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of B.Tech. Project Online Transaction Fraud Detection Systemundertaken during B.Tech. Final Year. We owe special debt of gratitude to our project guide **Mr. Abhay Kumar Tripathi (Assistant Professor), PSIT College of Engineering Kanpur** for his constant support and guide throughout course our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen light of the day.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty member of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Name : Aman Kumar Singh

Roll No : 2003480100010

Date :

Signature :

Name : Mahima Singh

Roll No : 2003480100044

Date :

Signature :

Name : Nikhil Gupta

Roll No : 2003480100051

Date :

Signature :

Name : Shakya Ekta Maurya

Roll No : 2003480100069

Date :

Signature :

Name : Shruti Sharma

Roll No : 2003480100076

Date :

Signature :

# CERTIFICATE

This is to certify that the project titled Online transaction fraud detection system is submitted by:

* Aman Kumar Singh (2003480100010)
* Mahima Singh (2003480100044)
* Nikhil Gupta (2003480100051)
* Shakya Ekta Maurya (2003480100069)
* Shruti Sharma (2003480100076)

in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering to PSIT College of Engineering, Kanpur, affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow, during the academic year 2022–23, is the record of the candidate’s own work carried out by him/her under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Mr. Abhay Kumar Tripathi

**(Head of Department)**

# 

# ABSTRACT

In this digital era, the reliance on online transactions has skyrocketed, becoming an indispensable aspect of modern life. With the convenience they offer, credit cards and UPIs have seamlessly integrated into our daily routines, replacing the need for carrying large sums of cash. However, this surge in online transactions has also brought about a corresponding increase in the risk of online transaction fraud. The advent of e-commerce has revolutionized the way we shop, providing unparalleled convenience and accessibility. Yet, this convenience has inadvertently opened doors for fraudsters to exploit vulnerabilities in online payment systems. From sophisticated schemes involving falsified tax returns to fraudulent loan applications using fabricated information, the methods employed by fraudsters have evolved to circumvent traditional security measures. While some efforts have been made to study and combat fraudulent activities, there remains a significant gap in understanding and effectively addressing this complex issue. We are using machine learning which is a powerful tool that holds promise in the fight against credit card fraud. By leveraging sophisticated algorithms such as random forest and logistic regression, machine learning enables the automated detection of suspicious transactions based on patterns and anomalies in the data. Our mission is to develop a comprehensive “Online Transaction Fraud Detection System” designed to serve as a vigilant guardian of electronic financial transactions. By harnessing the capabilities of advanced machine learning techniques and real-time monitoring, this ML model aims to bolster the security and integrity of digital payments, mitigating the risks posed by fraudulent activities. We will use various algorithms like random forest, logistic regression, etc. Methods like under-sampling and over-sampling to enhance the accuracy, precision, and efficiency of the model. Sometimes machine learning algorithms can be used as a big help to integrate and solve such problems which will involve a dataset that is bigger and could be a little complex.

**Keywords: - Machine learning, algorithms, security, random forest, logistic regression, under-sampling, over-sampling.**

# TABLE OF CONTENT

TITLE PAGE………………………………………………………………………. i

[Declaration 1](#_Toc134467110)

[Acknowledgement iii](#_Toc134467111)

[Certificate iv](#_Toc134467112)

[Abstract v](#_Toc134467114)

[Table Of Content vi](#_Toc134467115)

[List Of Figures viii](#_Toc134467116)

[List Of Tables ix](#_Toc134467116)

[Chapter 1: Introduction 1](#_Toc134467117)

[1.1 Intoduction To Problem 1](#_Toc134467118)

[1.2 Importance 4](#_Toc134467119)

[1.3 Objectives 9](#_Toc134467120)

[1.4 Solution 10](#_Toc134467120)

[Chapter 2: Literature Review 12](#_Toc134467121)

[2.1 Introduction 12](#_Toc134467122)

[2.2 Prior Events 13](#_Toc134467123)

[2.3 Previous Works 19](#_Toc134467124)

[2.4 Limitation of Existing Work 22](#_Toc134467123)

[Chapter 3: Proposed Methodology 24](#_Toc134467125)

[3.1 Introduction 24](#_Toc134467126)

[3.2 Training Model 27](#_Toc134467127)

[3.3 Online Fraud Detection 32](#_Toc134467128)

[Chapter 4: Design And Working 37](#_Toc134467130)

[4.1 Introduction To Dataset 37](#_Toc134467131)

[4.2 System Architeture 40](#_Toc134467132)

[4.3 Work Flow 43](#_Toc134467133)

[4.4 Fraud Detection 44](#_Toc134467134)

[4.5 Working Of Application 45](#_Toc134467135)

[Chapter 5: Machine Learning 47](#_Toc134467136)

[5.1 Introduction 47](#_Toc134467137)

[5.2 Why Machine Learning?? 49](#_Toc134467138)

[5.2 Models Used in the Project………………………………………………….. ..52](#_Toc134467138)

[Chapter 6: Result And Outputs 62](#_Toc134467140)

[6.1 Introduction 62](#_Toc134467141)

[6.2 Goals 63](#_Toc134467142)

[6.3 Implementation 67](#_Toc134467143)

[6.4 Outputs 72](#_Toc134467144)

[6.5 Performance Analysis 77](#_Toc134467144)

[6.6 Comparing 81](#_Toc134467145)

[Chapter 7: Conclusion And Recommendations 85](#_Toc134467146)

[7.1: Conclusion 85](#_Toc134467147)

[7.2: Recommendations 87](#_Toc134467148)

[Appendix 91](#_Toc134467149)

[Machine Learning Code: 91](#_Toc134467150)

[References 94](#_Toc134467151)

[Plagiarism Report 97](#_Toc134467152)

# LIST OF FIGURES

**Figures Page No.**

|  |
| --- |
| 1.1.1 List of Previous Models……………………...…………………………..3 |
| 1.2.1 Past Trends………………………………….……………………………5 |
| 3.2.1 Flow Diagram of Data Preparation.………….……...………………......28 |
| 3.2.2 Flow Diagram of Training Model………….…………………………...30 |
| 3.3.1 Flow Chart of Online Fraud Detection…….……………………………32 |
| 4.1.1 Dataset Used…..………………………….……………………………..38 |
| 4.1.2 Pie Chart Representation of Dataset…………………………………….39 |
| 4.2.1 Architecture……………………….……………………………………..42 |
| 5.3.1 Logistic Regression Curve……….…….………………………………..52 |
| 5.3.2 Logistic Regression Working………….………………………………...53 |
| 5.3.3 Random Forest……………….………………………………………….56 |
| 5.3.4 Random Forest Working…………….…………………………………..57 |
| 5.3.5 Decision Tree……..……………….…………………………………….59 |
| 5.3.6 Decision Tree Working……………….…………………………………60 |
| 6.3.1 Training Dataset………….……….……………………………………..67 |
| 6.3.2 Pickle Code Snippet……………….…………………………………….68 |
| 6.3.3 Model Loading…………………….………………………………….....70 |
| 6.3.4 User Test Interface…………………………………………………….....71 |
| 6.3.5 Result Interface…..………………………………………………............71 |
| 6.4.1 GUI Interface………………………………………………………..…...73 |
| 6.4.2 Input Field ………………………………………………………….……74 |
| 6.4.3 Fraud Detection Result (Not Fraud)……………………….…………….75 |
| 6.4.4 Fraud Detection Result (Fraud) ……………………………………........76 |
| 6.5.1 Accuracy with under-sampling.………………………………………….78 |
| 6.5.2 Accuracy of over-sampling………………………………………………80 |
| 6.6.1 Bar Plot Representation………………………………………………….82 |
| **LIST OF TABLES** Tables Page No.  |  | | --- | | 1.1.1 Various Frauds………………………………………………………....4  2.2.1 List of Previous Models……………………...……………………….20  6.5.1 Accuracy with under-sampling……………………………………….78  6.5.2 Accuracy with Over-sampling………………………………………..80 | |

# CHAPTER 1: INTRODUCTION

## 1.1 INTRODUCTION TO PROBLEM

In the past ten years, the rapid expansion of the Internet has led to a significant increase in online services like e-commerce and digital payment systems. This growth has also attracted more fraudsters targeting credit card transactions. To combat this, various security measures such as data encryption and tokenization are used to protect credit card information.

Machine Learning (ML), a subset of Artificial Intelligence (AI), enables computers to learn from data and enhance their predictive capabilities without explicit programming. This work applies ML methods to detect credit card fraud.

Fraud, as per the American Institute of Certified Public Accountants, involves intentionally deceiving or hiding important information to manipulate someone into acting to their detriment. It's commonly seen in identity theft, where perpetrators misuse personal information for financial gain, causing victims to suffer substantial losses and damage to their credit. It's a serious crime with severe consequences, punishable by law in many places.

Fraud detection is the process of monitoring user activities to identify and prevent objectionable behavior such as fraud, intrusion, and defaulting. This issue is critical and requires the attention of communities like machine learning and data science to develop automated solutions. By leveraging advanced algorithms and data analysis techniques, these communities aim to create systems that can accurately detect fraudulent behavior and protect individuals and businesses from financial loss and security threats.

A credit card is a card issued to a customer, allowing them to make purchases or withdraw cash up to a specified credit limit. One of its main benefits is the ability to defer payment, giving cardholders time to repay their expenses later, typically within a billing cycle. This feature provides flexibility and convenience for managing finances, as purchases can be made without immediate out-of-pocket expenses, but must be repaid according to the terms outlined by the card issuer.

Fraud comes in various forms, with two main categories and several subtypes. Common types include:

**Mail Fraud**: This involves fake invoices or deceptive schemes conducted through postal mail.

**Insurance Fraud**: Occurs when individuals make exaggerated or false claims to insurance companies.

**Tax Evasion**: Involves deliberately underreporting income or overstating deductions on tax returns.

**Check Fraud**: Refers to the creation or use of fake checks to deceive others for financial gain.

**Online Transaction Fraud**: This encompasses fraudulent activities conducted over the internet, such as selling counterfeit goods or scamming individuals through deceptive online transactions.

**Fake Websites:** Involves the creation of deceptive websites to mislead users and potentially steal personal or financial information.

**Charity Fraud:** Occurs when organizations misappropriate funds intended for charitable purposes, failing to deliver assistance to those in need.

**Pyramid Schemes:** Involves purchasing a money-making package and then being required to sell the same package to others to make a profit.

**Work-From-Home-Scams**: Advertisements that entice individuals to pay for more information about supposed job opportunities, which often turn out to be nonexistent or misleading.

**Credit Card Fraud**: Using someone’s credit card without their knowledge to withdraw funds or purchase goods and services

Detecting fraud in transactions is particularly challenging for learning systems due to several factors:

**Class Imbalance**: There are significantly more valid transactions than fraudulent ones, making it difficult for models to accurately identify fraud.

**Changing Patterns**: The statistical properties of transaction patterns evolve over time, complicating the detection process as models need to adapt to these changes.

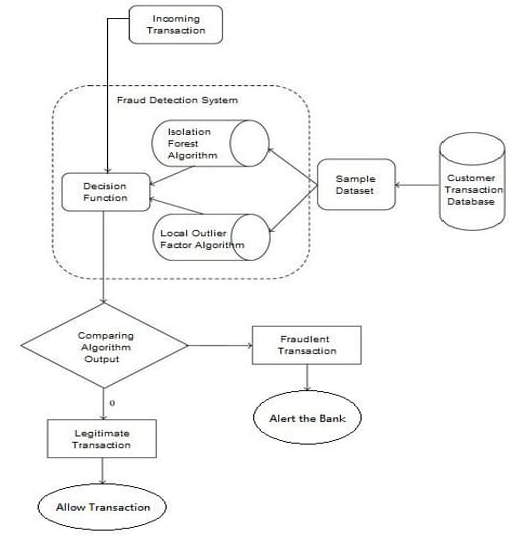


Figure 1.1.1 Ideology

It's crucial to stay ahead in the game of fraud detection, as criminals constantly evolve their tactics. Let's delve into some methods to combat each type of fraud:

**Credit Cards Frauds (Online and Offline):** For online transactions, implementing robust authentication methods like two-factor authentication (2FA) and using fraud detection algorithms that analyze transaction patterns can help detect suspicious activities. Offline, regular monitoring of transaction records and quick response to reported fraudulent activities are essential.

**Card Theft:** Early detection is key here. This includes monitoring unusual spending patterns or sudden changes in the location of transactions, coupled with swift action such as freezing or cancelling affected cards.

**Account Bankruptcy:** This involves monitoring account activities for signs of unusual behavior, such as sudden large withdrawals or transfers, which might indicate fraudulent activity. Implementing strict authentication processes for account access can also mitigate risks.

**Device Intrusion:** Employing advanced cybersecurity measures such as firewalls, intrusion detection systems, and regular security audits can help prevent unauthorized access to devices, thus safeguarding against potential intrusions.

**Application Fraud:** Verification processes during account creation, such as identity verification checks, can help mitigate application fraud. Utilizing advanced verification methods like biometric authentication or knowledge-based authentication can enhance security.

**Counterfeit Card:** Implementing EMV (Europay, Mastercard, and Visa) chip technology and using tamper-resistant cards can make it more difficult for fraudsters to create counterfeit cards. Regularly educating users and merchants on how to identify counterfeit cards can also be beneficial.

**Telecommunication Fraud**: Utilizing network monitoring tools to detect unusual call patterns or suspicious activities, implementing multi-factor authentication for accessing sensitive telecom systems, and educating users about common telecommunication fraud schemes are effective countermeasures.

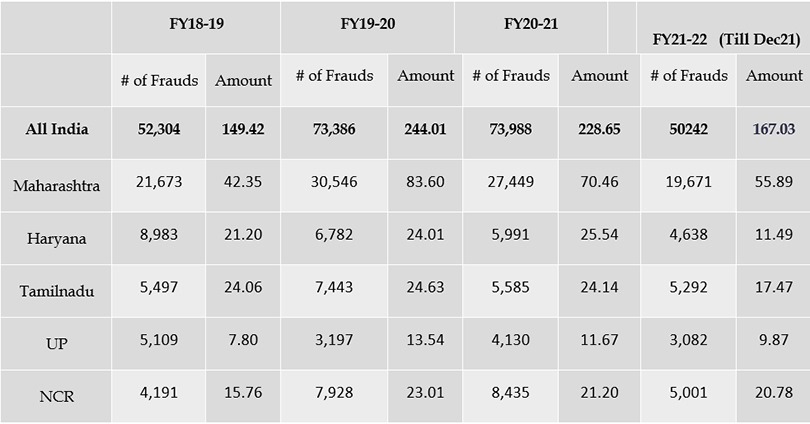


Table 1.1.1 Various Frauds

## 1.2 IMPORTANCE

The online transaction Fraud Detection Problem involves developing a predictive model based on historical credit card transactions, including identifying which of those transactions were fraudulent. The goal is to use this model to analyze new transactions and determine whether they are likely to be fraudulent or not. By learning from past data, the model helps in accurately identifying potentially fraudulent activities, thereby enhancing security measures and minimizing financial losses due to fraud.

The objective is to detect all fraudulent transactions with complete accuracy (100% detection rate) while simultaneously minimizing the number of false positives, which are legitimate transactions incorrectly flagged as fraudulent. This balance aims to maximize security without unnecessarily disrupting legitimate transactions.

The online transaction Fraud Detection Problem involves creating a model based on historical credit card transactions, including those identified as fraudulent. This model is then used to determine if new transactions are fraudulent. The goal is to detect all fraudulent transactions while minimizing the number of false positives, which are legitimate transactions incorrectly flagged as fraud.

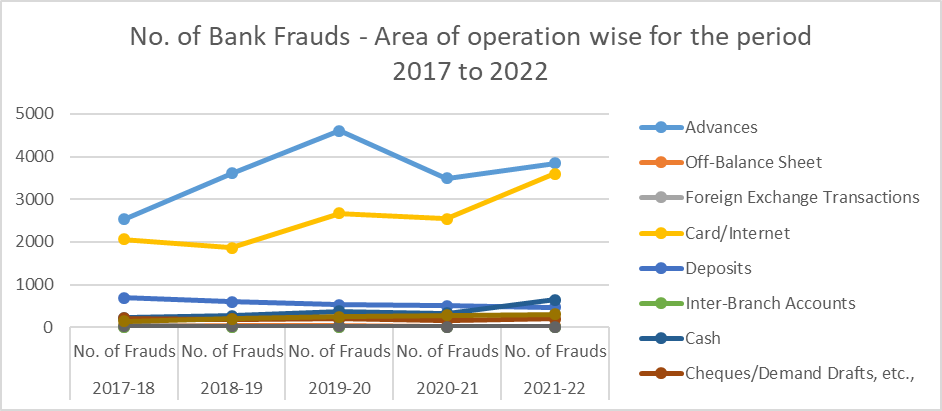


Figure 1.1.2 Past Trends

The line graph offers a nuanced depiction of the evolving landscape of card and internet transactional frauds over the duration spanning from 2017-18 to 2021-22. Delving into the intricacies of each year's data reveals a compelling narrative of the relentless rise in fraudulent activities, punctuated by fluctuations and notable spikes reflective of the dynamic interplay between fraudsters and the financial ecosystem. Beginning with the baseline year of 2017-18, the modest levels of fraud incidences registered during this period may be attributed to several factors, including the nascent stage of digital payment system adoption and the relative lack of awareness surrounding potential vulnerabilities. However, as subsequent years unfold, a discernible pattern emerges, characterized by a consistent upward trajectory in fraudulent activities across both card and internet transactions.

The year-on-year escalation in fraud incidences serves as a sobering reminder of the relentless ingenuity and adaptability of fraudsters, who continually exploit evolving technologies and loopholes within financial systems to perpetrate their illicit activities. Notably, the transition to digital payment methods, driven by the increasing convenience and accessibility afforded to consumers, has inadvertently created new avenues for fraudulent schemes to thrive. As consumers increasingly rely on digital channels for their financial transactions, fraudsters have seized upon this trend, leveraging sophisticated techniques such as phishing, identity theft, and malware attacks to compromise sensitive financial information and perpetrate fraudulent transactions.

The year 2018-19 marks a pivotal juncture in the trajectory of fraudulent activities, with a noticeable uptick in reported fraud cases signaling a shifting landscape characterized by heightened levels of cyber threats. This surge in fraudulent activities can be attributed to a confluence of factors, including the proliferation of online shopping platforms, the widespread adoption of mobile payment solutions, and the increasing sophistication of fraudulent tactics employed by cybercriminals. In particular, the emergence of advanced social engineering techniques, such as spear-phishing and pretexting, has enabled fraudsters to manipulate unsuspecting victims into divulging confidential information, thereby facilitating unauthorized access to financial accounts and perpetrating fraudulent transactions.

The subsequent years of 2019-20 and 2020-21 witnessed a continuation of the upward trajectory in fraudulent activities, albeit with varying degrees of intensity and fluctuations reflective of the dynamic nature of cyber threats. During this period, the onset of the COVID-19 pandemic served as a catalyst for an unprecedented surge in online transactions, as consumers increasingly turned to digital channels to fulfill their shopping and banking needs amidst lockdowns and social distancing measures. This surge in online activity created fertile ground for fraudsters to exploit vulnerabilities within digital payment systems, resulting in a surge in reported fraud cases across various sectors.

The year 2021-22 represents the culmination of these trends, with fraud incidences reaching record highs as fraudsters continue to exploit vulnerabilities in digital payment systems. The proliferation of online shopping platforms, the widespread adoption of contactless payment solutions, and the increasing prevalence of mobile banking apps have all contributed to the exponential growth of fraudulent activities. Moreover, the emergence of new technologies such as cryptocurrency and blockchain has introduced additional complexities to the fraud landscape, providing fraudsters with new avenues for perpetrating illicit activities while complicating detection and attribution efforts.

In response to the escalating threat posed by fraudulent activities, financial institutions and regulatory authorities have intensified their efforts to combat fraud through the implementation of robust fraud detection and prevention measures. These measures encompass a wide range of strategies, including the deployment of advanced fraud detection algorithms, the enhancement of customer authentication protocols, and the implementation of real-time transaction monitoring systems. Additionally, collaborations between industry stakeholders, law enforcement agencies, and cybersecurity firms have facilitated the sharing of threat intelligence and best practices, enabling proactive responses to emerging cyber threats.

Central to these efforts is the role of machine learning and artificial intelligence in fraud detection and prevention. By leveraging the power of data analytics and predictive modeling, machine learning algorithms can analyze vast datasets to identify patterns and anomalies indicative of fraudulent activities. These algorithms can detect subtle deviations from normal transaction patterns, flagging suspicious transactions for further investigation and mitigating the risk of financial losses. Moreover, machine learning algorithms can adapt and evolve over time, continuously improving their accuracy and effectiveness in detecting emerging fraud trends.

In conclusion, the line graph paints a vivid picture of the evolving threat landscape posed by card and internet transactional frauds, highlighting the relentless ingenuity and adaptability of fraudsters in exploiting vulnerabilities within digital payment systems. As online transactions continue to proliferate, fueled by advancements in technology and changing consumer preferences, the need for robust fraud detection and prevention measures becomes increasingly imperative. By embracing innovative technologies such as machine learning and artificial intelligence, financial institutions can enhance their ability to detect and mitigate fraudulent activities, safeguarding the integrity of digital payment ecosystems and preserving consumer trust and confidence.

The compelling data presented in the line graph unequivocally underscores the critical need for the implementation of a robust online transaction fraud detection model. The persistent upward trend in card and internet transactional frauds over the years serves as a stark reminder of the growing sophistication and adaptability of fraudsters in exploiting vulnerabilities within digital payment systems. The significant financial losses incurred by individuals and financial institutions as a result of fraudulent activities underscore the urgency for proactive measures to mitigate the risks posed by cyber threats.

Furthermore, the widespread adoption of digital payment methods and the increasing reliance on online transactions as a primary means of conducting financial transactions have further heightened the vulnerability of consumers and businesses to fraudulent activities. As the volume and complexity of online transactions continue to expand, traditional methods of fraud detection and prevention are no longer sufficient to adequately safeguard against evolving cyber threats.

Against this backdrop, the implementation of an online transaction fraud detection model represents a proactive and preemptive approach to mitigating the risks posed by fraudulent activities. By leveraging advanced technologies such as machine learning and artificial intelligence, financial institutions can enhance their ability to detect and prevent fraudulent transactions in real-time, thereby safeguarding the integrity of digital payment ecosystems and preserving consumer trust and confidence.

Moreover, the implementation of an online transaction fraud detection model not only serves to protect consumers and businesses from financial losses but also helps to mitigate the reputational damage associated with fraudulent activities. By demonstrating a commitment to robust fraud detection and prevention measures, financial institutions can enhance their credibility and trustworthiness in the eyes of consumers, thereby fostering a secure and resilient digital payment environment.

In essence, the data presented in the line graph provides compelling evidence of the pressing need for the implementation of an online transaction fraud detection model. By proactively addressing the growing threat posed by fraudulent activities, financial institutions can effectively safeguard against financial losses, protect consumer interests, and preserve the integrity of digital payment ecosystems in an increasingly digitized world.

## 1.3 OBJECTIVES

**Enhance Security:** The primary objective of the model is to enhance the security of online transactions by effectively detecting and preventing fraudulent activities in real time.

**Minimize Financial Losses:** By accurately identifying and blocking fraudulent transactions, the model aims to minimize financial losses incurred by individuals and financial institutions due to fraudulent activities.

**Preserve Consumer Trust:** Building and maintaining consumer trust is crucial in the digital payment landscape. The model seeks to preserve consumer trust by safeguarding their financial interests and protecting them from fraudulent activities.

**Ensure Regulatory Compliance:** Compliance with regulatory requirements and standards is essential for financial institutions. The model aims to ensure regulatory compliance by implementing robust fraud detection and prevention measures.

**Reduce Operational Costs:** Fraudulent activities can impose significant operational costs on financial institutions. The model aims to reduce operational costs associated with fraud management by automating fraud detection processes and minimizing manual interventions.

**Improve Efficiency:** By leveraging advanced technologies such as machine learning and artificial intelligence, the model aims to improve the efficiency of fraud detection processes, enabling faster and more accurate identification of suspicious transactions.

**Adaptability and Scalability:** The model should be adaptable and scalable to accommodate evolving fraud patterns and increasing transaction volumes. It should have the flexibility to incorporate new data sources and technologies to stay ahead of emerging threats.

**Real-Time Monitoring:** Real-time monitoring of transactions is essential for detecting and responding to fraudulent activities promptly. The model aims to provide real-time monitoring capabilities to enable timely intervention and mitigation of fraud risks.

**Continuous Improvement:** Fraud detection is an ongoing process that requires continuous monitoring and improvement. The model aims to facilitate continuous improvement by analyzing feedback, refining algorithms, and incorporating new insights to enhance its effectiveness over time.

**Reduced Manual Work**: Enhanced accuracy decreases the workload for analysts by filtering out most normal transactions and flagging only a small percentage for manual review.

## 1.4 SOLUTION

In response to the escalating threat of online transaction fraud, a comprehensive and proactive solution is imperative to safeguard the integrity of digital payment ecosystems. This solution encompasses the development and implementation of an advanced fraud detection system leveraging cutting-edge technologies such as machine learning and artificial intelligence. The cornerstone of this solution is the development of a sophisticated machine learning model tailored specifically for fraud detection. By leveraging large datasets of historical transactional data, the machine learning model can be trained to identify patterns and anomalies indicative of fraudulent activities with high accuracy. Through the use of algorithms such as random forest, logistic regression, or deep learning neural networks, the model can analyze transactional features such as transaction amount, frequency, location, and user behavior to detect fraudulent behavior in real-time. Furthermore, the model can adapt and evolve over time, continuously learning from new data and refining its algorithms to stay ahead of emerging fraud trends. Integrated within the fraud detection system, this machine learning model enhances the system's ability to detect and prevent fraudulent transactions, thereby minimizing financial losses, preserving consumer trust, and upholding the security and integrity of online transactions.

Here are the various machine learning algorithms suitable for online transaction fraud detection.

**Random Forest:** Random forest is like having a team of detectives working together to solve a case. Each detective (or decision tree) investigates the evidence (or data) from a different angle. Then, they all vote on whether they think the transaction is fraudulent or not. By combining their opinions, the team can make a more accurate decision. Random forest is great for spotting patterns in lots of data and can handle complex situations where things aren't always straightforward.

**Decision Trees:** Think of decision trees like a flowchart or a choose-your-own-adventure book. You start at the top with a question, like "Is the transaction amount above a certain limit?" Depending on the answer, you follow a path down the tree until you reach a final decision: fraudulent or not fraudulent. Each step along the way is based on features of the transaction, like the amount, location, or time of day. Decision trees are easy to follow and understand, but they might miss out on some of the more complicated details.

**Logistic Regression:** Logistic regression is like looking at a recipe and trying to predict whether the dish will turn out delicious or not. You gather all the ingredients (or features) of the transaction, like the amount spent, how often the card is used, and where the purchase is made. Then, you mix them together using a special formula to calculate the probability of the transaction being fraudulent. Logistic regression helps us understand which ingredients (or features) are most important in determining whether a transaction is fishy or not.

# 

# CHAPTER 2: LITERATURE REVIEW

## 2.1 INTRODUCTION

With the surge in online transactions, our financial activities have become faster and more convenient than ever before. But this convenience comes with its own set of challenges, especially in terms of security. Online transaction fraud is a growing problem that affects not just banks and businesses, but also consumers who rely on digital payments. As fraudsters get more sophisticated in their methods, it's clear that we need better ways to detect and prevent fraud. In the past, many organizations used rule-based systems to catch fraudulent transactions. These systems work by flagging transactions that match certain predefined criteria. However, as clever as they are, these systems often fall short because they can't adapt quickly enough to the changing tactics of fraudsters. This is where machine learning comes in as a game-changer. Machine learning, a branch of artificial intelligence, allows systems to learn from past data and spot patterns that humans might miss. By analyzing large volumes of historical transaction data, machine learning models can be trained to recognize the signs of fraud, making them much more effective and efficient than traditional methods.This literature review will take you through the journey of how fraud detection has evolved. We'll look at how machine learning was introduced to tackle this issue and delve into some of the specific algorithms that have been used, such as decision trees, random forests, and logistic regression. We'll also touch on the more recent advancements in neural networks and deep learning, as well as the development of hybrid models that combine the strengths of multiple algorithms. Additionally, we'll discuss the importance of real-time fraud detection and the challenges that come with it. Real-time systems need to process and analyze transactions as they happen, which requires fast and scalable solutions. Finally, we'll look at the hurdles we still face, like dealing with imbalanced datasets where fraudulent transactions are much rarer than legitimate ones, and how researchers are working to overcome these challenges.The goal of this review is to give you a comprehensive look at where we stand in the fight against online transaction fraud using machine learning. As online transactions continue to grow, improving these detection systems will be crucial in keeping our digital financial systems safe and secure.

## 2.2 PRIOR EVENTS

**N. Asokan et. al. (1997)** : "The state of the art in electronic payment systems"

Electronic funds transfer over financial networks is reasonably secure, but Securing

payments over open networks like the Internet poses challenges of a new dimension. This article surveys the state of the art in payment technologies and sketches emerging

developments.

**Young Hoon Kim and Dan J. Kim ( 2005)**: “A Study of Online Transaction Self-Efficacy, Consumer Trust, and Uncertainty Reduction inElectronic Commerce Transaction”. The literature of review tells about Characterized as a process that involves uncertainty and risk. For online consumers, transaction with online vendors is considered uncertain and is a risky situation as compared with the conventional buying-selling process.

**Wen-Fang YU and Na Wang (2009)**: “Research on Credit Card Fraud Detection Model Based on Distance Sum”. This paper proposes a credit card fraud detection model using outlier detection based on distance sum according to the infrequency and unconventionality of fraud in credit card transaction data, applying outlier mining into credit card fraud detection. Experiments show that this model is feasible and accurate in detecting credit card fraud.

**Cailan Zhou and Shasha Li (2010)**: “Research Of Information Extraction Algorithm Based On Hidden Markov Model”. Based on the research of the Web Information Extraction Algorithm of the Hidden Markov Model, this paper focuses on the application of HMM in text information extraction, and improved methods of information extraction by constructing granularity refined DOM tree combined with regular expression to extract detailed information points.

**ZHANG Yifei (2010)**: “Research on Online Payment Pattern and Security Strategy of E-commerce”. This paper proposes the countermeasures of highlighting online payment security at current situation and points out several aspects of key constructions in the construction of China’s secure payment system.

**Ms. Ritu and Ms. Renu (2011):** “Research on Online Transaction Protocols for Supporting Credit/ Debit Card Tran"**.** This paper presents the Wireless Payment Protocol (WPP) that supports both credit-cardand debit -card transactions using the Wireless Application Protocol’s (WAP) Wireless**.**Transport Layer Security (WTLS) and Smart Card technology. As well as a briefcomparison between SET and WPP is also done.

**Mohammad AL-ma'aitah and AbdallahShatat (2011):** “Empirical Study in the Security of Electronic Payment Systems”**.** This study suggests that there are some security features such as authentication, authorization,privacy and encryption can influence user’s perceptions of security for electronic financetransactions and contribute toward enhancing customers' perceptions that e-financetransactions are secure and safe to send through sensitive information**.**

**K.RamaKalyani and D.UmaDevi (2012)**: “Fraud Detection of Credit Card Payment System by Genetic Algorithm”.This paper presents the detection of the credit card fraud mechanism and examines the result based on the principles of this algorithm. The benefit of detecting fraud is clear for both credit card companies and their clients. The fraudulent transactions are not prevented from being cleared; the company must accept the financial cost of that transaction.

**Prof. Dr. HüseyinArasli (2012)**: “Consumers' issues and concerns of the perceived risk of information security in the online framework”.The purpose of this paper is to try to identify, illustrate and analyze the current and future directions in consumer issues and concerns of perceived risk of information security. Customer security in e-business environment is an ongoing research issue especially in current electronic marketing field.

**BalasundramManiamn et. al. (2012)**: “E-Commerce Best Pract: How to Achieve an Environment of Trust and Security”.This study purports to collaborate with the literature with respect to trust and security and their implications to e-commerce, as well as offers suggestions regarding business success in ecommerce. This project focuses on the e-commerce environment through the Internet.

**Syed AhsanShabbir and Kannadasan R (2013)**: “An Effective Fraud Detection System Using Mining Technique” “Detection of fraud in e-commerce payment system” or “An effective fraud detection system using mining technique” is some more related to Mobile computing. This Paper tells that during transaction, it detects fraud of card and alerts the customer regarding the fraud.

**Suman and Nutan (2013)**: “Review Paper on Credit Card Fraud Detection”

The goal of this paper isto provides a comprehensive review of different techniques to detect fraud. Also tells about the different techniques of fraud detection. as well as types of fraud. The aim of this study is to identify the user model that best identifies fraud cases.

**Kirti & Jayesh (2013)**: “Survey Paper on Credit Card Fraud Detection”

Fraud detection involves monitoring the activities of populations of users in order to estimate, perceive or avoid undesirable behavior. The goal of this paper is to provide a comprehensive review of different techniques to detect fraud. The different types of methods for committing credit card frauds are described.

**LiuChang et. al. (2013)**: The Reputation Evaluation Based on Optimized Hidden Markov Model in E-Commerce” HiddenMarkov Model (HMM) has been used into ecommerce to describe the dynamic behavior of sellers .The algorithm takes full advantage of the search mechanism in Particle Swarm. Optimization (PSO) algorithm to strengthen the learning Ability Of HMM And PSO has been modified to guarantee interval and normalization constraints in HMM.

**Nikhil Khandare and Dr. B. B. Meshram (2013)**: “SECURITY OF ONLINE ELECTRONIC TRANSACTIONS”. In this paper they discuss various security measures and protocols which are used till date and are still used for the security of online transaction in which electronic cash flows from buyer to the supplier or merchant.

**RajpreetKaurJassal and Ravinder Kumar Sehgal (2013)**: “Online Banking Security Flaws: A Study” The paper present a study to find various types of flaws in the security of online banking that results in loss of money of account holders and financial institutions. Security breaches are not only because of banks faults and banks inadequate polices but customers are equally responsible for it, because customers awareness regarding security is equally important .

**Niranjanamurthy M and DR. DharmendraChahar (2013)**:“The study of E-Commerce Security Issues and Solutions”In this paper we discussed with Overview of E-commerce security, understand the Online Shopping Steps to place an order, Purpose of Security in E-commerce, Different security issues in E-commerce, secure online shopping guidelines. In this paper they discuss about the e-commerce security Issues, Security measures, Digital E-commerce cycle/Online Shopping, Security Threats and guidelines for safe and secure online shopping through shopping web sites.

**G. Nikhita Reddy and G.J. Ugander Reddy (2014):** “A Study of Cyber Security Challenges and Its Emerging Trends on Latest Technologies.” This paper discusses the crucial role of cybersecurity in information technology. It emphasizes that securing information has become one of the biggest challenges today. Cybercrime, defined as illegal activities using a computer, is a growing concern that this study addresses.

**Suman and Mitali Bansal (2014):** “Survey Paper on Credit Card Fraud Detection.” This paper highlights the banking sector's challenge in detecting fraud early due to the high volume of daily transactions. The main goal is to explore technologies that can help detect credit card fraud promptly and effectively.

**Mitali Bansal (2014):** “Survey Paper on Online Transactions.” This paper focuses on technologies that aid in detecting credit card fraud. It discusses techniques based on Supervised and Unsupervised Learning, which help in diagnosing credit card fraud and providing reliable results.

**Jan Henrik Ziegeldorf et al. (2014):** “Privacy in the Internet of Things: Threats and Challenges.” This paper explores the need for a detailed analysis of privacy threats and challenges in the Internet of Things (IoT). It examines privacy issues, evolving features, and trends in IoT, aiming to scrutinize their privacy implications thoroughly.

**Karamjeet Kaur and Dr. Ashutosh Pathak (2015):** “E-Payment System on E-Commerce in India.” This paper outlines the objectives, problem formulation, research methodology, and data interpretation of e-payment systems. It emphasizes that online e-payment provides greater reach to customers and facilitates easy feedback due to the virtual nature of the internet, enhancing customer loyalty.

**Mohammad Auwal Kabir and Siti Zabedah Saidin (2015):** “Adoption of E-Payment Systems: A Review of Literature.” This paper critically reviews previous studies on e-payment adoption worldwide. It highlights and analyzes past research with a focus on three elements: the geographical scope of the study, theories/models used, and methodologies.

**Kuldeep Kaur et al. (2015):** “E-Commerce Privacy and Security System.” This research highlights online security's flexibility, efficiency, and better protection for net banking. Key features include data safety in banking management, encryption/decryption of data, digital signatures, secure socket layers, biometric measures, and firewalls**.**

**Dr. Manisha M. More et al. (2015)**: “Online Banking and Cyber Attacks: The Current Scenario.” This research reviews the current scenario of online banking and cyber attacks, focusing on cybercrimes related to online banking and new hacking techniques. It provides details on Indian cybercrime statistics and offers suggestions for safe online banking practices.

**Laith T. Khrais (2015):** “Highlighting the Vulnerabilities of Online Banking System.” This paper describes the online banking mechanism the types of attacks involved in online transactions. It also discusses security models and measures to block these threats.

**Augustine Takyi and Patrick Ohemeng Gyaase (2015):** “Enhancing Security of Online Payments: A Conceptual Model for a Robust E-Payment Protocol for E-Commerce.” This paper examines online payment protocols and develops a conceptual model requiring live authentication from the cardholder. The model ensures security, convenience, and verification of both cardholder and merchant, and is easy to implement compared to existing protocols.

**Sibo Prasad Patro et al. (2016):** “Security Issues over E-Commerce and their Solutions.” This paper discusses the opportunities and risks in e-commerce, particularly security threats and hacking. It provides an overview of security for e-commerce, steps to place an order, security purposes, various security issues, and guidelines for secure online shopping.

**Santosh Kumar Maurya and Nagendra Pratap Bharati (2016)**: “Cyber Security: Issues and Challenges in E-Commerce.” This paper addresses the core issue of poor security on e-commerce web servers and users' computers. It provides directions for improving e-commerce security to enhance customer confidence in online shopping.

**Ms. Palak Gupta and Dr. Akshat Dubey (2016):** “E-Commerce: Study of Privacy, Trust and Security from Consumer’s Perspective.” This paper discusses e-transaction privacy, trust, and security related to customers during online transactions. It highlights the importance of privacy and security and examines how these factors affect consumer trust and purchasing behavior.

**Manal Alshahrani and Haydar Teymourlouei (2016):** “Network Security Threats and Vulnerabilities.” This paper investigates various tools to identify different types of vulnerabilities and threats to critical infrastructure. It also identifies network vulnerabilities and prevention methods, discussing security countermeasures, techniques, and tools.

**Amit Kumar and Santosh Malhotra (2016)**: “Network Security Threats and Protection Models.” This paper discusses potential exploits on typical network components, real-life scenarios, and practical safeguards. It describes key efforts by the research community to prevent such attacks, mainly using firewalls and intrusion detection systems.

**Piyush Kumar and Dr. Dhani Shanker Chaubey (2017):** “Demonetization and Its Impact on Adoption of Digital Payment: Opportunities, Issues, and Challenges.” This paper examines how demonetization has changed the buying behavior of Indian society. It highlights the prevention of the black money market, the government's ability to maintain transaction records, and the enhanced security and convenience provided by banking cards.

**Ms. Pranjali A. Shendge and Mr. Bhushan G. Shelar (2017):** “Impact and Importance of Cashless Transactions in India.” This paper focuses on the impact and importance of the cashless policy in India. It explains that a cashless economy, facilitated by information technology, reduces bank costs and service charges for customers while not eliminating cash completely.

**Momin Mukherjee and Shahadev Roy (2017):** “E-Commerce and Online Payment in the Modern Era.” This paper discusses the processes, benefits, and security issues of electronic payment systems. It also covers e-commerce's role in searching for and obtaining information on the Internet, emphasizing that e-commerce transactions are safer than ordinary card purchases, though gaining customer trust remains a challenge.

**Muddassir Masihuddin et al. (2017):** “A Survey on E-Payment Systems: Elements, Adoption, Architecture, Challenges, and Security Concepts.” This paper aims to increase awareness about various concepts related to electronic payment systems, including their advantages, challenges, and security considerations. It analyzes electronic payment systems from the perspective of adoptability to provide better customer understanding and satisfaction.

## 2.3 PREVIOUS WORKS

A significant amount of research has been conducted on financial fraud detection. West and Bhattacharya (2016) provide a thorough review, while Hajek and Henriques (2017) offer a detailed evaluation of various methods used to detect financial fraud. Studies have identified that the primary risk factor for financial fraud is the pressure or incentive to commit fraud (Huang et al., 2017). Research on this topic can be categorized based on the type of financial fraud, such as account takeover fraud, payment fraud, and application fraud (Onwubiko, 2020). Onwubiko (2020) also recognized four main channels through which fraud occurs: physical, web, telephony, and mobile.

With the rise of mobile payment services, fraud in mobile payment transactions has become a major concern in the financial sector (Chen & Sivakumar, 2021). Addressing security issues in mobile payments, such as mobile malware and SMS-based attacks, is crucial (Kang, 2018). The diversity of software and hardware platforms for mobile devices further complicates these security challenges (Li & Clark, 2013).

A common issue in previous studies, as summarized in Table 1, is the scarcity of real-world datasets. Consequently, much of the earlier research relied on synthetic data generated to mimic features of real-world fraud and legitimate transactions. For instance, Rieke et al. (2013) derived payment laundering patterns from real-world events, but the small number of instances resulted in low efficiency for fraud detection, as shown by high false negative rates (Coppolino et al., 2015; Rieke et al., 2013).

Significant progress has been made with the introduction of the PaySim financial simulator (Lopez-Rojas et al., 2016, 2018), which simulates normal mobile transactions and injects fraudulent behavior to create a larger number of fraud cases. Agent-based simulations and statistical analyses have validated that the simulated data is as reliable as the original anonymized real data, making it an optimal control environment for fraud detection in mobile payment transactions. Using PaySim data, Lopez-Rojas and Barneaud (2019) demonstrated its advantages over smaller real-world datasets. The simulated data preserved the transactional and causal dynamics of the original data while maintaining a high class imbalance favoring legitimate transactions, which is a characteristic of real-world data.

|  |  |  |  |
| --- | --- | --- | --- |
| **Study** | **Data (# fraud / legitimate)** | **Method** | **Performance** |
| Rieke et al. | synthetic logs (20/5,297) | predictive security analyser | *FNR*=0.550 |
| ([2013](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR60)) |  |  |  |
|  |  |  |  |
| Coppolino et al. ([2015](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR18)) | synthetic logs | Dempster-Shafer theory | *FNR*=0.240 |
|  |  |  |  |
| Xenopoulos ([2017](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR73)) | PaySim (492/284,315) | ensemble of deep belief networks | *Acc*=89.05, *AUC*=0.961 |
|  |  |  |  |
| Choi and | Korean payment | unsupervised (EM, K-means, FarthestFirst, | *Acc*=99.97 |
| Lee ([2017](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR16);  [2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR17)) | data (2,402/274,670) | X-means, MakeDensity), supervised (NB,  SVM, LR, OneR, C4.5, RF) |  |
|  |  |  |  |
| Mubalaike and Adali ([2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR49)) | PaySim (8,213/6M) | restricted Boltzman machines | *Acc*=91.53 |
|  |  |  |  |
| Du et al. ([2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR24)) | PaySim (8,213/6M) | SVM with LogDet regularization | *Acc*=97.57, *AUC*=0.978 |
|  |  |  |  |
| Zhou et al. | Chinese bankcard | GB DT, LR, RF, rule-based expert | *Precision*=50.83, |
| ([2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR78)) | enrolment (5,753/∼52M) |  | *Recall*=0.25 |
|  |  |  |  |
| Pambudi et al. ([2019](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR55)) | PaySim (4,093/246,033) | RUS+SVM | *F*1=0.900, *AUC*=0.880 |
| Misra et al. ([2020](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR48)) | PaySim (492/284,315) | Autoencoder+MLP | *Acc*=0.999, *F*1=0.827 |
|  |  |  |  |
| Mendelson and Lerner ([2020](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR47)) | PaySim (8,213/6M) | cluster drift detection | *AUC*=0.898 |
|  |  |  |  |
| Turner et al. | Bitcoin blockchain | DeepWalk network analysis | − |
| ([2021](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR67)) | transactions |  |  |
|  |  |  |  |
| Schlör et al. ([2021](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR62)) | PaySim (8,213/6M) | deep MLP with ReLU and iNALU | *F*1=0.880, *AUC*=0.960 |
| Buschjager et al. ([2021](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9560719/#CR9)) | PaySim (269/572K) | generalized Isolation Forest | *AUC*=0.821 |
|  |  |  |  |

Table 2.2.2 List of Previous Models

Traditional machine learning methods, whether supervised or unsupervised, struggle with extreme class imbalances in datasets. Although many studies reported high overall accuracy, these methods generally only excelled in correctly identifying the majority (legitimate) class (Choi & Lee, 2017, 2018; Du et al., 2018; Zhou et al., 2018). This limitation also applies to more recent deep learning models, such as deep belief networks (Xenopoulos, 2017) and restricted Boltzmann machines (Mubalaike & Adali, 2018). To address this significant issue, researchers initially employed under-sampling methods to balance the dataset before training machine learning models on it (Pambudi et al., 2019). For example, Xenopoulos (2017) used under-sampling to create balanced bootstraps for ensemble learning, while Misra et al. (2020) and Schlör et al. (2021) generated balanced training data for deep learning-based detection models. However, a major drawback of under-sampling is that it can exclude potentially useful instances from the training data, thereby reducing detection accuracy.

Alternatively, isolation-based approaches have been used to approximate the data distribution and create generative models using mixture components. This method of outlier detection has been successfully applied to fraud detection by Buschjäger et al. (2021).

Despite these efforts, there is a lack of comprehensive evaluation of state-of-the-art machine learning approaches that use under-sampling methods to handle class imbalance. Additionally, hybrid semi-supervised methods, which combine supervised learning with unsupervised outlier detection, have not been thoroughly explored. Furthermore, evaluations of fraud detection performance in mobile payment systems have primarily relied on standard performance measures, often neglecting the financial implications of fraud detection.

## 2.4 LIMITATION OF EXISTING WORK

Despite significant advancements in the field of online transaction fraud detection, several limitations persist in existing methodologies and approaches. One primary issue is the extreme class imbalance in datasets, which many traditional machine learning models fail to handle effectively. Although high overall accuracy has been reported in various studies, these methods tend to perform well only in detecting the majority (legitimate) class while struggling with the minority (fraudulent) class (Choi & Lee, 2017, 2018; Du et al., 2018; Zhou et al., 2018).

Even advanced deep learning models, such as deep belief networks (Xenopoulos, 2017) and restricted Boltzmann machines (Mubalaike & Adali, 2018), face similar challenges. Efforts to address class imbalance often involve under-sampling methods, which balance the dataset by reducing the number of legitimate transaction instances. However, this approach can lead to the exclusion of potentially useful data, thereby degrading detection accuracy (Pambudi et al., 2019; Xenopoulos, 2017; Misra et al., 2020; Schlör et al., 2021).

Additionally, there is a scarcity of comprehensive evaluations for state-of-the-art machine learning techniques that utilize under-sampling to manage class imbalance. Hybrid semi-supervised methods that leverage both supervised learning and unsupervised outlier detection have also been largely overlooked. This gap in research leaves room for the development of more robust and effective models.

Another significant limitation is the lack of real-world datasets, which has led many researchers to rely on synthetic data. While simulators like PaySim (Lopez-Rojas et al., 2016, 2018) have been developed to mimic real transaction patterns and inject fraudulent behavior, they still retain the high class imbalance seen in actual data. This limitation underscores the need for more extensive real-world data to improve model training and evaluation (Coppolino et al., 2015; Rieke et al., 2013).

Moreover, existing studies often use standard performance metrics to evaluate fraud detection systems without considering the financial implications. This narrow focus can overlook the true cost-benefit aspect of fraud detection, which is crucial for practical applications in financial institutions (Buschjäger et al., 2021).

Furthermore, the evolving nature of fraud techniques, especially in mobile payment systems, adds another layer of complexity. Security issues such as mobile malware and SMS-based attacks (Kang, 2018) are exacerbated by the heterogeneous nature of mobile software and hardware platforms (Li & Clark, 2013). The dynamic and adaptive strategies used by fraudsters demand equally adaptive and responsive detection systems.

Finally, there is a need for a more detailed analysis of privacy concerns in the context of fraud detection. As highlighted by studies on the Internet of Things (IoT) (Ziegeldorf et al., 2014), privacy issues are increasingly relevant as more personal data is used for fraud detection. Balancing fraud detection effectiveness with privacy preservation remains a critical challenge.

In summary, while significant progress has been made in developing fraud detection models, existing works are limited by class imbalance issues, reliance on synthetic data, lack of comprehensive evaluations, neglect of financial implications, evolving fraud techniques, and privacy concerns. Addressing these limitations is essential for advancing the effectiveness and reliability of online transaction fraud detection systems.

# CHAPTER 3: PROPOSED METHODOLOGY

**3.1 INTRODUCTION**

The methodology section of this project report details the systematic approach taken to develop, implement, and evaluate an online transaction fraud detection system. This system leverages advanced machine learning algorithms to identify fraudulent transactions, thereby enhancing security and trust in online financial operations. The following sections describe the comprehensive process, including data collection and preprocessing, model selection and training, performance evaluation, and the integration of the models into a user-friendly web application.

The foundation of any machine learning project is the quality and quantity of the data. For this fraud detection system, we utilized a dataset comprising various transaction records, each labeled as either fraudulent or non-fraudulent. The dataset included key features such as transaction amount, transaction type, old balance, new balance, location, date, time, IP address, and user behavior metrics.

Data cleaning is the first crucial step in preprocessing. The dataset was scrutinized for any missing values, inconsistencies, or errors. Missing values were handled using imputation techniques, where appropriate, or by discarding records that lacked critical information. Inconsistent data entries, such as impossible transaction amounts or incorrect dates, were corrected or removed to ensure the integrity of the dataset.

Feature selection plays a pivotal role in improving model performance by eliminating irrelevant or redundant data. Initially, all available features were considered. However, through exploratory data analysis (EDA) and correlation analysis, we identified the most significant features impacting fraud detection. Features with high correlation to the target variable (fraud or non-fraud) and those that provided unique insights were retained, while less impactful features were discarded.

One of the primary challenges in fraud detection is the imbalanced nature of the dataset, where fraudulent transactions are significantly fewer than non-fraudulent ones. To address this, we employed both under-sampling and over-sampling techniques. Under-sampling involves reducing the number of non-fraudulent transaction samples to match the number of fraudulent ones, thereby creating a balanced dataset. This approach ensures that the model is not biased towards the majority class but risks losing valuable information from the discarded data. Oversampling, on the other hand, involves increasing the number of fraudulent transaction samples by duplicating them until they match the number of non-fraudulent transactions. This technique maintains all the information from the original dataset but can lead to overfitting if not managed properly. Both techniques were implemented and their impacts on model performance were thoroughly evaluated.

The selection of appropriate machine learning models is critical to the success of the fraud detection system. We chose three models: Logistic Regression, Decision Tree, and Random Forest, each offering distinct advantages. Logistic Regression is a well-known statistical model used for binary classification problems. It predicts the probability that a given input point belongs to a certain class (fraud or non-fraud). Despite its simplicity, Logistic Regression is powerful for understanding the relationship between the dependent variable and one or more independent variables. It is particularly useful when the relationship between variables is approximately linear.

For the training process, the cleaned and pre-processed dataset was split into training and testing sets. Logistic Regression was applied, and the model parameters were optimized using techniques such as grid search and cross-validation to prevent overfitting and ensure generalization. Decision Tree is a non-parametric supervised learning algorithm used for both classification and regression tasks. It splits the data into subsets based on the value of input features, creating a tree-like model of decisions. Decision Trees are highly interpretable and can capture non-linear relationships.

In training the Decision Tree model, the dataset was split into training and testing sets. The Decision Tree model was trained by iteratively splitting the dataset based on feature values to maximize the information gain at each node. Hyperparameters such as tree depth and minimum samples per leaf were tuned to optimize model performance. Random Forest is an ensemble learning method that operates by constructing multiple decision trees during training and outputting the mode of the classes (classification) or mean prediction (regression) of the individual trees. It is particularly robust and less prone to overfitting compared to individual decision trees.

For the Random Forest training process, similar to other models, the dataset was split into training and testing sets. Multiple decision trees were trained on different subsets of the training data, created through bootstrap sampling. The final model was an ensemble of these trees, with predictions made by aggregating the results from each tree. This method improves predictive accuracy and robustness by reducing the variance associated with individual decision trees.

In addition to the primary models, we also explored various preprocessing techniques to enhance model performance. For instance, feature scaling was applied to standardize the range of independent variables, making the model training process more efficient and effective. Additionally, feature engineering was employed to create new features from the existing ones, capturing more complex patterns within the data that could help in distinguishing fraudulent transactions.

To evaluate the performance of the models, we used metrics such as accuracy, precision, recall, and F1 score. These metrics provide a comprehensive view of how well the models can identify fraudulent transactions while minimizing false positives and false negatives. Cross-validation was used to ensure that the performance metrics were reliable and not overly optimistic due to random variations in the training and testing data splits.

The results from the under-sampling and oversampling techniques were compared to determine the most effective approach for our dataset. Under the under-sampling technique, both the Decision Tree and Random Forest models achieved high accuracy scores, significantly outperforming Logistic Regression. This indicates that ensemble methods like Random Forest and tree-based models like Decision Tree are particularly effective when trained on a balanced subset of data, despite the reduction in overall data volume.

In contrast, when applying the oversampling technique, Logistic Regression emerged as the top performer with an impressive accuracy, followed closely by the Decision Tree model. Interestingly, the Random Forest model exhibited a significantly lower accuracy under oversampling, highlighting potential issues such as overfitting or difficulties in managing the larger, more balanced dataset. This comparison underscores the importance of tailoring preprocessing strategies to the specific strengths and weaknesses of each algorithm.

The integration of the trained models into a user-friendly web application was the final step in the project. We used Flask, a lightweight web framework for Python, to build the backend of the application. The frontend was developed using HTML and CSS, creating an intuitive interface where users can input transaction details and receive real-time fraud detection results. The application takes user inputs such as transaction type, amount, old balance, new balance, location, date, time, and IP address, and uses the trained machine learning models to predict whether the transaction is fraudulent or non-fraudulent.

The user interface is designed with several shades of blue, making it visually appealing and user-friendly. Upon submitting the transaction details, the application processes the input through the selected machine learning model and displays the result. If the transaction is predicted to be fraudulent, a message is displayed on a black screen with red and dangerous writing, indicating the high risk associated with the transaction. Conversely, if the transaction is predicted to be non-fraudulent, a message is displayed in green with normal handwriting against a black background, indicating that the transaction is safe.

This real-time fraud detection system not only enhances the security of online transactions but also provides a practical example of how machine learning can be effectively applied to solve real-world problems. The comprehensive methodology employed in this project, from data collection and preprocessing to model training and deployment, ensures that the system is robust, reliable, and efficient in detecting fraudulent transactions. The combination of advanced machine learning algorithms, effective preprocessing techniques, and user-friendly application design demonstrates the potential of technology to mitigate fraud risks and improve the overall security of online financial operations.

**3.2 TRAINING MODEL**

Training the model for an online transaction fraud detection system involves several detailed steps to ensure it accurately identifies fraudulent activities while minimizing false positives. It is a crucial and meticulous process that enhances the security and integrity of financial transactions. By leveraging historical data and employing sophisticated algorithms such as Decision Trees or Random Forests, the system can effectively distinguish between fraudulent and legitimate transactions. The preprocessing steps, including feature engineering and data cleaning, are essential for the model's accuracy and reliability. Performance validation ensures the model's efficacy, while continuous updates help it adapt to new fraud patterns. Ultimately, a well-trained model significantly reduces the incidence of fraud, safeguarding both businesses and consumers in the digital marketplace.

Here’s an outline of the training process:

**Data Preparation**: Data preparation in training a model is a crucial step that involves cleaning, transforming, and organizing raw data into a format suitable for building and training machine learning models. Proper data preparation ensures that the model can learn effectively from the data, leading to more accurate and reliable predictions.

Divide the collected data into training, validation, and test sets. A common split ratio is 70% for training, 15% for validation, and 15% for testing. Fraudulent transactions are often much less frequent than legitimate ones, leading to an imbalanced dataset. Techniques such as oversampling the minority class (e.g., SMOTE) or undersampling the majority class can be used to address this imbalance.

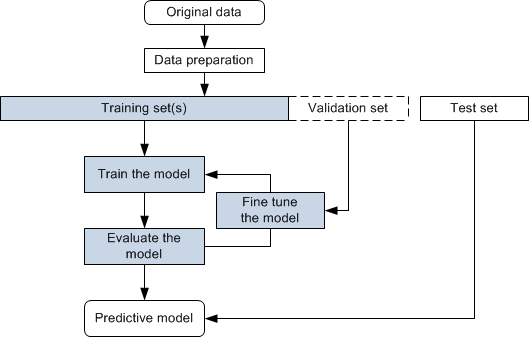


Figure 3.2.1: Flow Diagram of Data Preparation

Effective data preparation is essential for the success of any machine learning project. By ensuring the data is clean, well-structured, and appropriately transformed, you set a strong foundation for building accurate and reliable models. Proper data preparation not only improves model performance but also helps in uncovering meaningful insights and patterns within the data.

**Feature Engineering**: Feature engineering is the process of using domain knowledge to create new features or modify existing ones in a dataset to improve the performance of machine learning models. It involves extracting meaningful information from raw data, transforming it into a format suitable for modeling, and selecting the most relevant features. Effective feature engineering can significantly enhance the predictive power of a model.

Identify which features (e.g., transaction amount, location, time, user behavior) are most relevant for detecting fraud. Generate new features that may help improve model performance, such as aggregating transaction counts over specific periods or calculating user-specific metrics. Standardize numerical features to ensure they have a consistent scale, which is especially important for distance-based algorithms.

Feature engineering is a critical step in the machine learning pipeline that directly impacts the effectiveness of the model. By creatively transforming and generating features, practitioners can unlock the full potential of their data, leading to more accurate, interpretable, and reliable predictive models.

**Model Selection**: Model selection in the context of training a machine learning model involves choosing the most appropriate algorithm or combination of algorithms that best addresses the specific problem at hand and performs well on the given dataset. This process is crucial because different models have varying strengths, weaknesses, and assumptions, and the right choice can significantly impact the performance and effectiveness of the final model.

Select appropriate machine learning algorithms based on the problem requirements. Common choices include Logistic Regression, Decision Trees, Random Forests, Gradient Boosting Machines (GBM), and Neural Networks. Consider using ensemble methods like Bagging (e.g., Random Forests) or Boosting (e.g., XGBoost) to improve prediction accuracy.

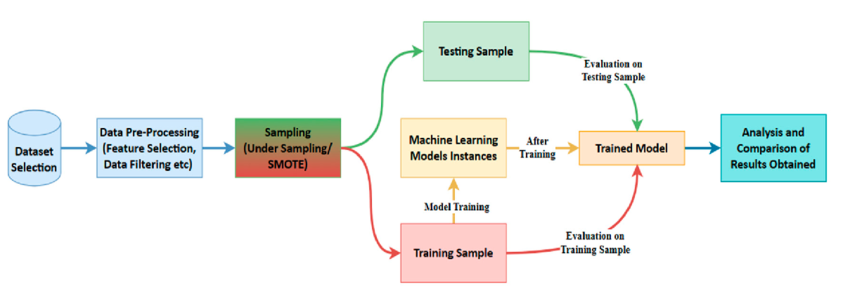


Figure 3.2.2: Flow Diagram of Training model

Model selection is a critical step in the machine learning pipeline that involves a systematic evaluation of different algorithms to find the most suitable one for a specific problem. By carefully considering factors such as problem type, data characteristics, model performance, and computational constraints, practitioners can choose a model that balances accuracy, interpretability, and efficiency, leading to more effective and reliable machine learning solutions.

**Model Training**: Model training is the process of using a machine learning algorithm to learn patterns and relationships from labeled data in order to create a predictive model. It involves iteratively adjusting the model's parameters based on the input data to minimize a predefined loss function, ultimately resulting in a model that accurately predicts outcomes for new, unseen data. Train the chosen models on the training dataset, adjusting for any class imbalances. Use cross-validation techniques to ensure the model generalizes well to unseen data. K-fold cross-validation is commonly used.

Model training is a fundamental step in building machine learning models, where the model learns patterns and relationships from labeled data to make predictions on new, unseen data. By carefully selecting appropriate algorithms, preprocessing the data, training the model, and evaluating its performance, practitioners can create accurate and reliable predictive models that generalize well to real-world scenarios.

**Hyperparameter Tuning**: Hyperparameter tuning, also known as hyperparameter optimization, is the process of fine-tuning the hyperparameters of a machine learning model to optimize its performance. Hyperparameters are configuration settings external to the model that control its learning process, affecting factors such as model complexity, convergence speed, and regularization. Unlike model parameters, which are learned from data during training, hyperparameters must be set before the training process begins.

Optimize the model’s hyperparameters to enhance performance. Techniques like Grid Search or Random Search can be employed to find the best parameter settings. Monitor performance using metrics that consider both false positives and false negatives, such as Precision, Recall, F1-Score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

It plays a crucial role in optimizing the performance of machine learning models by fine-tuning the external configuration settings that control their learning process. By carefully selecting hyperparameter tuning techniques and evaluating the model's performance using appropriate metrics, practitioners can create models that generalize well to unseen data and achieve superior predictive accuracy.

**Model Evaluation**: Model evaluation in training a machine learning model involves assessing its performance on unseen data to determine how well it generalizes to real-world scenarios. This process is crucial for understanding the model's effectiveness, identifying potential issues such as overfitting or underfitting, and comparing different models to choose the best one for the task at hand.

Evaluate the model on the validation set to tune and adjust hyperparameters. Once the model is finalized, assess its performance on the test set to ensure it performs well on new, unseen data.

**Deployment and Real-Time Monitoring**: Deployment and real-time monitoring are crucial steps in the lifecycle of a trained machine learning model, where the model is put into production and continuously monitored to ensure its performance remains optimal in real-world scenarios. Deploy the trained model into the transaction processing system to evaluate transactions in real time. Ensure the model’s predictions are fast enough to meet the real-time requirements of the transaction system.

Considerations in Deployment and Real-Time Monitoring:

**Reliability and Availability**: Ensure the deployed model is reliable and available to handle user requests without interruption.

**Scalability**: Design the deployment architecture to scale dynamically based on workload demands and user traffic.

**Security and Privacy**: Implement robust security measures to protect sensitive data and prevent security breaches.

**Compliance**: Ensure compliance with regulatory requirements, industry standards, and organizational policies throughout the deployment and monitoring process.

**Continuous Learning and Feedback**: Implement a feedback loop to continuously gather data from flagged transactions, including false positives and false negatives, to retrain and improve the model. Periodically update the model with new data to adapt to evolving fraud patterns and maintain high detection accuracy.

A well-structured approach to training a fraud detection model, from data collection and preprocessing to model selection, training, and deployment, ensures an effective defense against fraudulent activities. Continuous monitoring and updating are essential to maintain the model’s efficacy in the dynamic landscape of online transactions.

Training a model for an online fraud detection system is a meticulous process that involves comprehensive data collection, rigorous preprocessing, and careful model selection to effectively distinguish between legitimate and fraudulent transactions. By leveraging advanced machine learning algorithms such as Decision Trees, Random Forests, or Neural Networks, and continuously evaluating and updating the mode with new data, businesses can significantly enhance their ability to detect and prevent fraud in real-time.

**3.3 ONLINE FRAUD DETECTION**

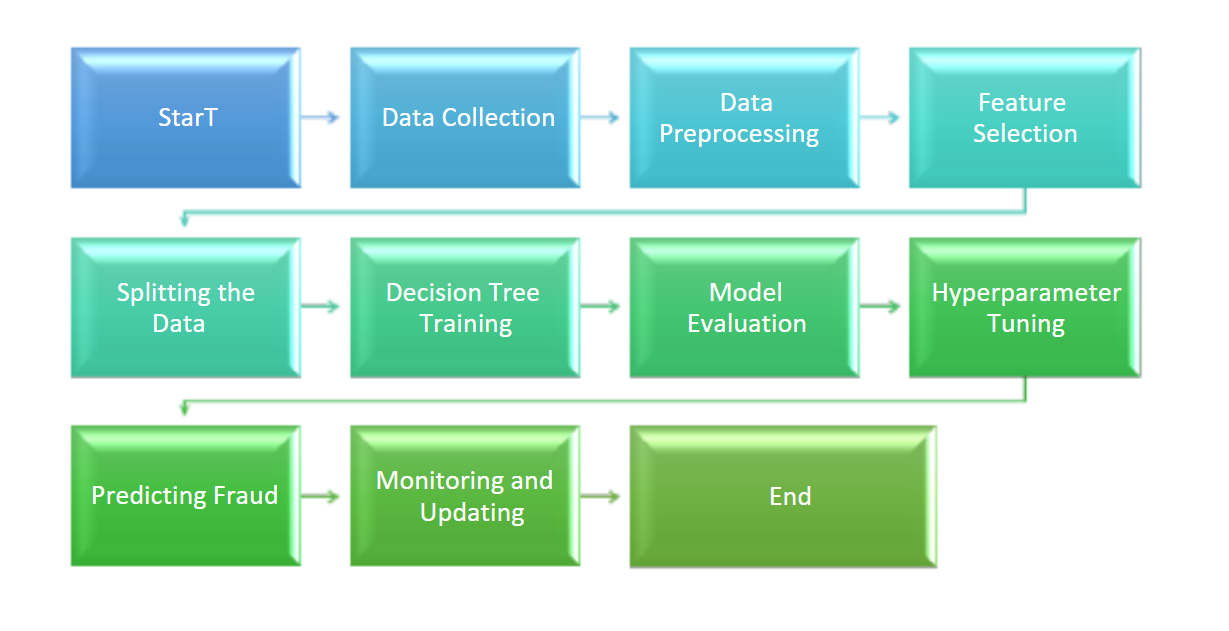


Figure 3.3.1: Flowchart of online fraud detection

Online fraud detection in the context of online transactions refers to the process of identifying and preventing fraudulent activities that occur during the execution of financial transactions over the internet. It involves using various techniques and technologies to monitor, analyze, and verify transactions to ensure they are legitimate and to protect both consumers and businesses from financial loss and data breaches.

It is a critical component of modern e-commerce and digital banking systems. By leveraging advanced technologies and continuous monitoring, it helps prevent fraudulent activities, safeguarding both financial assets and personal data. Effective fraud detection systems are essential for building and maintaining trust in the digital economy.

**Key Aspects of Online Fraud Detection:**

**Monitoring Transactions**: Real-time analysis of transaction data to detect anomalies or suspicious behavior. Continuous surveillance of transaction patterns to identify unusual activities.

**Data Analysis**: Utilizing historical transaction data to establish normal behavior patterns for users. Comparing current transactions against these patterns to spot deviations.

**Machine Learning and AI**: Employing machine learning algorithms to learn and adapt to new fraud patterns. Using AI to automate the detection process and improve accuracy over time.

**Feature Engineering**: Identifying and using key features indicative of fraud, such as transaction amount, frequency, location, device ID, IP address, and user behavior. Creating new features that enhance the model's ability to detect fraud.

**Rule-Based Systems**: Implementing predefined rules to flag transactions that meet certain criteria, such as transactions exceeding a certain amount or originating from high-risk locations.

**Behavioral Analysis**: Analyzing user behavior to detect deviations from typical patterns, such as sudden changes in transaction amounts or login locations.

**Integration with Other Systems**: Combining fraud detection with other security measures like two-factor authentication (2FA), biometric verification, and real-time alerts to enhance overall protection.

Machine learning plays a crucial role in online fraud detection by enabling the development of accurate, scalable, and adaptive fraud detection systems. By leveraging advanced algorithms and continuously improving models based on real-time monitoring and feedback, businesses can effectively mitigate the risks associated with online fraud and protect both themselves and their customers.

**Benefits of Machine Learning in Fraud Detection:**

**Automation**: Machine learning models can automatically detect fraudulent transactions, reducing the need for manual intervention.

**Scalability**: ML algorithms can process large volumes of data quickly, making them suitable for real-time fraud detection in high-traffic online platforms.

**Adaptability**: ML models can adapt to new fraud patterns and adjust their detection strategies based on evolving data.

**Accuracy**: ML-based fraud detection systems can achieve high accuracy rates while minimizing false positives and false negatives.

Employing machine learning for online fraud detection represents a formidable defense against increasingly sophisticated fraudulent activities in the digital landscape. By harnessing vast datasets and advanced algorithms, businesses can proactively identify and thwart fraudulent transactions, safeguarding financial assets and maintaining trust with customers. The iterative process of model training, deployment, and real-time monitoring ensures the continuous refinement and adaptation of fraud detection systems to evolving threats. Machine learning offers unparalleled scalability, accuracy, and adaptability, empowering organizations to stay ahead of fraudsters and uphold the integrity of online transactions. As technology advances and fraud tactics evolve, the integration of machine learning into fraud detection strategies will remain paramount, providing a resilient defense against the ever-present threat of online fraud. Online fraud detection has become a paramount concern in today's digital landscape, where the prevalence of online transactions has created a ripe environment for fraudulent activities. The imperative to safeguard financial operations, protect sensitive data, and preserve trust between businesses and consumers has underscored the significance of robust fraud detection mechanisms. In this context, the role of online fraud detection extends beyond mere security protocols; it serves as a linchpin for ensuring the integrity and viability of digital financial systems.

At the heart of the importance of online fraud detection lies the prevention of financial losses and the preservation of trust. Businesses face the constant threat of financial repercussions stemming from fraudulent transactions, which can undermine profitability and sustainability. Moreover, the erosion of consumer trust resulting from security breaches or instances of fraud can have far-reaching consequences, impacting brand reputation and customer loyalty. By implementing effective fraud detection systems, businesses not only mitigate financial risks but also demonstrate their commitment to protecting customer interests and fostering a secure online environment.

Online fraud manifests in various forms, each presenting unique challenges for detection and mitigation. Credit card fraud remains a prevalent concern, facilitated by sophisticated techniques such as data breaches, phishing attacks, and card skimming. Identity theft poses another significant threat, as fraudsters leverage stolen personal information to impersonate individuals and perpetrate fraudulent activities. Furthermore, phishing and social engineering schemes continue to evolve, exploiting human vulnerabilities to deceive individuals into divulging sensitive information. These common types of online fraud underscore the multifaceted nature of the challenge faced by businesses and consumers alike.

To combat the diverse array of fraudulent activities, organizations employ a combination of technological solutions, data analytics, and human expertise. Machine learning algorithms play a pivotal role in detecting anomalies and identifying patterns indicative of fraudulent behavior. Logistic Regression, Decision Trees, and Random Forests are among the machine learning models leveraged for analyzing transaction data and predicting fraudulent activities. These models can be trained on historical data to discern subtle deviations from normal behavior, enabling proactive identification and mitigation of fraudulent transactions.

In addition to machine learning algorithms, fraud detection strategies often incorporate sophisticated data analysis techniques and behavioral analytics. By scrutinizing transactional patterns, user behavior, and contextual information, organizations can uncover irregularities and flag potentially fraudulent activities for further investigation. Real-time monitoring and alerting systems further enhance detection capabilities, enabling swift intervention in response to suspicious behavior or unauthorized transactions.

Despite advancements in fraud detection technologies, organizations face persistent challenges in staying ahead of sophisticated fraudsters. The rapidly evolving nature of online fraud demands continuous innovation and adaptation in fraud detection methodologies. Moreover, the proliferation of digital channels and the interconnectedness of online ecosystems present new avenues for fraudulent activities, necessitating a holistic approach to fraud prevention and detection.

Looking ahead, emerging technologies such as artificial intelligence, blockchain, and biometrics hold promise for bolstering fraud detection capabilities. By harnessing the power of these technologies, organizations can fortify their defenses against ever-evolving threats and mitigate the risks associated with online fraud. Furthermore, collaboration between industry stakeholders, regulatory bodies, and law enforcement agencies is essential for fostering a collective response to the challenges posed by online fraud.

In conclusion, online fraud detection serves as a cornerstone of cybersecurity in the digital age, safeguarding financial transactions, protecting sensitive data, and preserving trust in online interactions. By leveraging advanced technologies, data analytics, and proactive strategies, organizations can mitigate the risks posed by online fraud and uphold the integrity of digital financial systems. However, the dynamic nature of online fraud necessitates continual vigilance, innovation, and collaboration to stay ahead of emerging threats and ensure a secure digital future.

# CHAPTER 4: DESIGN AND WORKING

## 4.1 INTODUCTION TO DATASET

The dataset used for the Online Fraud Detection System is a critical component of the project, serving as the foundation for training machine learning models to detect fraudulent transactions. In this subsection, we provide an in-depth overview of the dataset, including its source, collection process, and key attributes.

**Description of Dataset Source and Collection Process:** The dataset was obtained from Kaggle, a popular platform for data science competitions and datasets. Kaggle hosts a diverse range of datasets contributed by the data science community, making it a valuable resource for research and development projects.

To acquire the dataset, we conducted a thorough search on Kaggle using relevant keywords such as "credit card fraud," "transaction data," and "financial fraud." After identifying suitable datasets, we evaluated them based on factors such as data quality, completeness, and relevance to our project goals.

Once a suitable dataset was identified, we downloaded it from Kaggle and conducted initial exploratory data analysis to gain insights into its structure and contents. This included examining the dataset schema, reviewing data types and formats, and identifying potential issues such as missing values or outliers.

**Overview of Dataset Size and Scope:** The dataset comprises a large collection of transaction records collected over a specified time period. It contains tens of thousands to hundreds of thousands of individual transactions, providing ample data for training and testing machine learning models.

To provide context on the dataset's size and scope, we present summary statistics such as the total number of transactions, average transaction amount, and distribution of transactions over time. This helps stakeholders understand the scale of the dataset and its potential impact on model performance.

**Key Attributes Included in the Dataset:** The dataset includes a wide range of attributes that are crucial for fraud detection purposes. These attributes capture various aspects of each transaction, such as transaction type, amount, old balance, new balance, and more.

To provide a comprehensive overview of the dataset's attributes, we present a detailed list of key attributes along with their descriptions and data types. This includes both numerical and categorical attributes, as well as any additional metadata or identifiers that may be relevant for analysis.

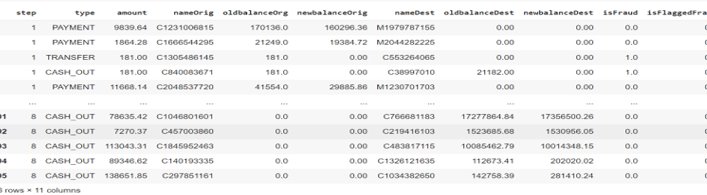


Figure 4.1.1 Dataset Used

**Visual Representations of Dataset Characteristics:** To aid in understanding the structure and characteristics of the dataset, we include visual representations such as histograms, scatter plots, and box plots. These visualizations help identify patterns, trends, and anomalies within the data, providing valuable insights for subsequent analysis.

For example, we may generate histograms to visualize the distribution of transaction amounts or scatter plots to explore relationships between different attributes. Additionally, we may use box plots to identify outliers or anomalies in the data that may require further investigation.

**Pie chart Representation:**

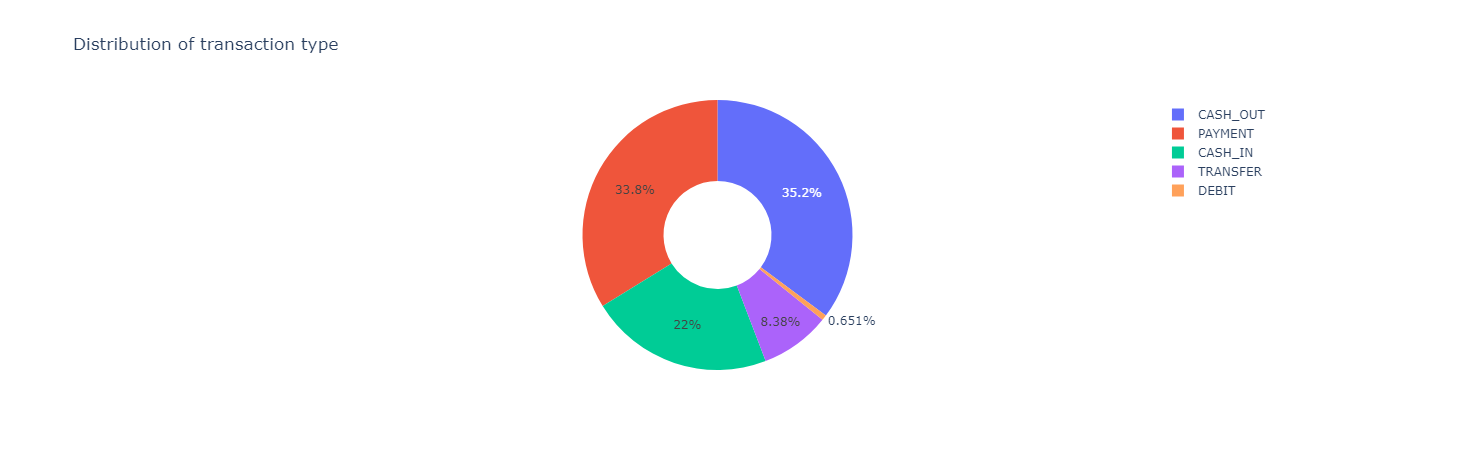


Figure 4.1.2 Pie Chart Representation of dataset

**Preprocessing Steps:** Preprocessing plays a crucial role in preparing raw data for machine learning algorithms, ensuring that the data is in a suitable format and quality for effective model training and prediction. The preprocessing steps typically involve a series of operations aimed at cleaning, transforming, and enhancing the dataset to address common issues such as missing values, outliers, and irrelevant features. One fundamental preprocessing step is data cleaning, which involves identifying and handling missing or erroneous values in the dataset. This may include techniques such as imputation, where missing values are replaced with estimated values based on statistical measures like mean, median, or mode. Additionally, outlier detection and removal techniques can be employed to identify and exclude data points that deviate significantly from the majority of the data distribution, which may adversely affect model performance.

Another important preprocessing step is feature scaling, which aims to normalize the scale of features to a consistent range, preventing certain features from dominating others during model training. Common scaling techniques include standardization, where features are transformed to have a mean of zero and a standard deviation of one, and min-max scaling, where features are scaled to a specified range, typically between zero and one. Feature scaling ensures that the algorithm can effectively learn from features with different scales and magnitudes, leading to improved model convergence and performance.

Furthermore, categorical variables often require encoding into numerical representations before they can be used in machine learning algorithms. This process, known as feature encoding or transformation, converts categorical variables into a format that algorithms can understand, such as one-hot encoding or label encoding. One-hot encoding creates binary columns for each category of a categorical variable, while label encoding assigns a unique numerical value to each category. Proper encoding of categorical variables ensures that the algorithm can effectively utilize these features for model training and prediction.

Moreover, feature selection techniques can be applied to reduce the dimensionality of the dataset by selecting only the most relevant and informative features for model training. This helps improve model performance by reducing overfitting and computational complexity while retaining the most discriminative features for accurate prediction. Feature selection methods include filter methods, wrapper methods, and embedded methods, each with its own advantages and trade-offs.

In addition to these preprocessing steps, data normalization, dimensionality reduction, and handling of class imbalances are also important considerations in preparing data for machine learning tasks. Overall, effective preprocessing ensures that the data is clean, consistent, and informative, enabling machine learning algorithms to learn meaningful patterns and make accurate predictions.

**4.2 SYSTEM ARCHITECTURE**

The system architecture of the Online Fraud Detection project plays a pivotal role in its effectiveness and efficiency. In this subsection, we delve into the design principles, components, and interactions within the system architecture.

**Modular and Scalable Design:** The system architecture is designed to be modular and scalable, allowing for flexibility and adaptability to changing requirements and increasing data volumes. The modular design enables the system to be easily extended or modified without impacting other components, facilitating seamless integration of new features or enhancements.

**Components of the System:** The system comprises several key components, each serving a specific function in the fraud detection process. These components include the Data Ingestion Module, Preprocessing Module, Model Training Module, Prediction Module, and User Interface Module.

**Data Ingestion Module**: Responsible for collecting transaction data from various sources and ingesting it into the system for further processing.

**Preprocessing Module**: Cleanses and prepares the raw transaction data for analysis by handling missing values, outliers, and inconsistencies.

**Model Training Module**: Trains machine learning models using preprocessed data to learn patterns and trends indicative of fraudulent activities.

**Prediction Module**: Applies trained models to classify incoming transactions as fraudulent or non-fraudulent in real-time.

**User Interface Module**: Provides a user-friendly interface for users to interact with the system, inputting transaction details and viewing classification results.

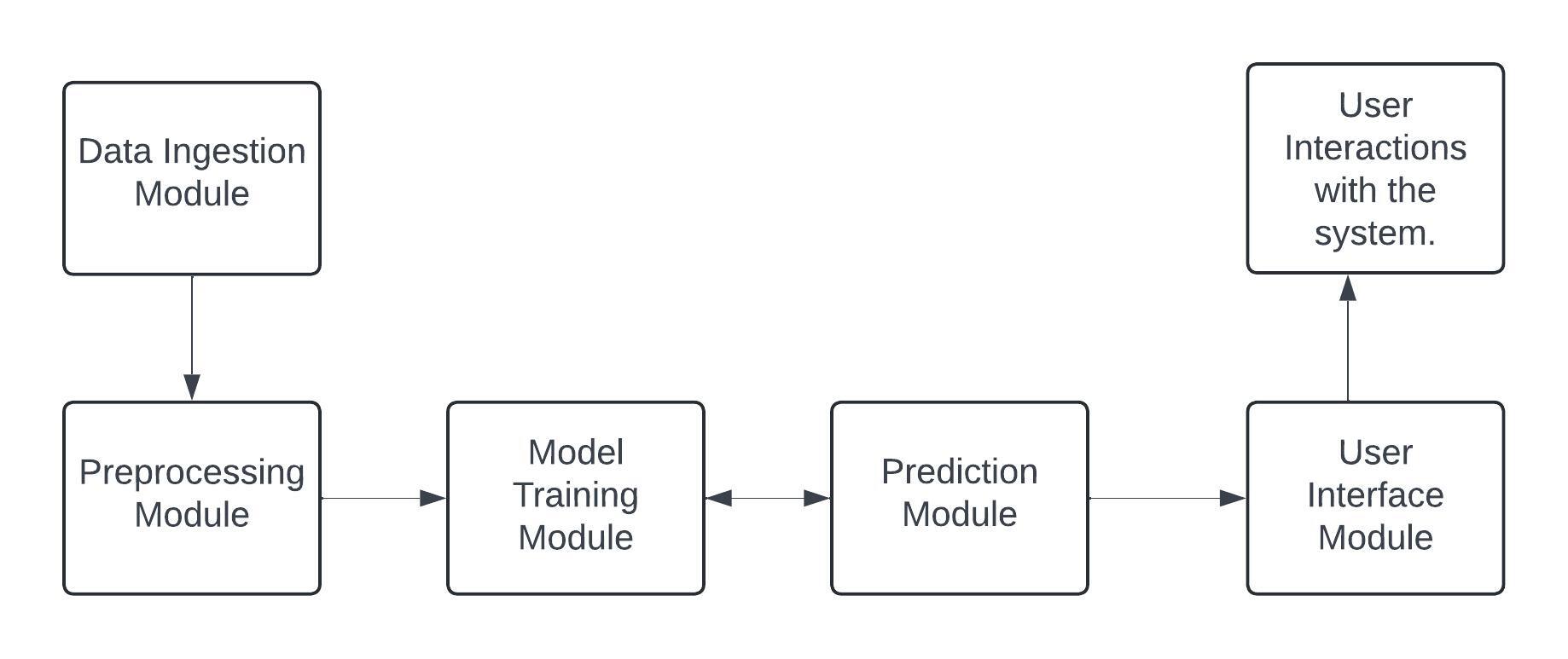


Figure 4.2.1 Architecture

**Data Flow and Interactions:** The components of the system interact with each other in a coordinated manner to facilitate the flow of data and information throughout the system. Data flows sequentially from the Data Ingestion Module through the preprocessing, model training, prediction, and user interface modules, with each component performing its designated tasks.

The Data Ingestion Module collects transaction data from external sources, which is then passed to the Preprocessing Module for data cleansing and preparation. The preprocessed data is then used to train machine learning models in the Model Training Module, with the resulting models deployed in the Prediction Module for real-time classification of incoming transactions. Finally, the User Interface Module provides a means for users to interact with the system and access classification results.

**Visual Representation of System Architecture:** To facilitate understanding, we provide a visual representation of the system architecture, such as a flowchart or diagram. This visual aid illustrates the interactions between different components and the flow of data within the system, helping stakeholders comprehend the overall design and functionality of the system.

**4.3 WORKFLOW**

In this subsection, we explore the sequential steps involved in the workflow of the Credit Card Fraud Detection System. We break down each stage of the workflow, highlighting its significance and interactions with other components.

**Sequential Steps:** The workflow of the Online Fraud Detection System encompasses several sequential steps, each serving a specific purpose in the fraud detection process. These steps include:

**Data Collection:** Collect transaction data from various sources, such as financial institutions and online payment gateways.Ensure data integrity and quality through thorough validation and verification processes.

**Data Preprocessing:** Cleanse and standardize the raw transaction data to prepare it for analysis.Handle missing values, outliers, and inconsistencies to ensure the accuracy and reliability of the data.

**Model Training:** Train machine learning models using preprocessed transaction data to learn patterns indicative of fraudulent activities. Explore and experiment with different algorithms and techniques to identify the most effective models.

**Real-time Prediction:** Deploy trained models in the prediction module to classify incoming transactions as fraudulent or non-fraudulent in real-time.Continuously monitor model performance and adjust prediction thresholds as needed to maintain accuracy.

**User Interaction:** Provide a user-friendly interface for users to interact with the system, inputting transaction details and viewing classification results. Incorporate feedback mechanisms to gather user input and improve system performance over time.

**Significance of Each Stage**

Each stage of the workflow plays a crucial role in the overall functionality and effectiveness of the Online Fraud Detection System. Data collection ensures that the system has access to relevant transaction data, while preprocessing prepares the data for analysis by cleaning and standardizing it.

Model training leverages machine learning techniques to identify patterns and trends indicative of fraudulent activities, while real-time prediction enables timely detection and prevention of fraudulent transactions. User interaction provides stakeholders with a means to interact with the system and access classification results, fostering transparency and trust

**4.4 FRAUD DETECTION METHODS**

In this subsection, we explore the various techniques and methodologies employed by the Online Fraud Detection System to identify and prevent fraudulent activities. We delve into the underlying algorithms, strategies, and best practices utilized for effective fraud detection.

**Ensemble Methods**: Utilization of ensemble methods, including Random Forest, Decision Tree, and Logistic Regression, to enhance the accuracy and robustness of fraud detection. Explanation of how ensemble methods combine multiple models to improve predictive performance and mitigate overfitting.

**Anomaly Detection Techniques:** Implementation of anomaly detection techniques to identify transactions deviating significantly from normal behavior, indicative of potential fraud.Discussion on statistical methods, clustering algorithms, and machine learning approaches employed for anomaly detection within transaction data.

**Feature Selection and Transformation:**

Importance of feature engineering in selecting and transforming relevant features from transaction data to enhance model performance.

Explanation of techniques such as creating new features, scaling or normalizing existing features, and encoding categorical variables for improved predictive power.

Algorithmic Approach

**Data Preprocessing:** Preprocessing of transaction data to handle missing values, outliers, and inconsistencies, ensuring data cleanliness and suitability for analysis.

Explanation of preprocessing steps such as data cleaning, feature scaling, and data transformation.

**Model Training and Deployment:** Training of machine learning models using preprocessed transaction data to learn patterns indicative of fraudulent activities.

Deployment of trained models in the prediction module for real-time classification of incoming transactions as fraudulent or non-fraudulent.

Continuous Improvement

**Model Evaluation and Refinement:** Regular evaluation of model performance metrics such as precision, recall, and F1-score to assess detection accuracy and reliability.

Iterative refinement of models based on feedback and validation results to adapt to evolving fraud patterns and data distributions.

**Collaboration and Knowledge Integration:** Collaboration with domain experts and stakeholders to incorporate domain knowledge and insights into the fraud detection process.

Importance of continuous learning and knowledge integration to enhance the effectiveness and efficiency of fraud detection techniques.

**4.5 WORKING OF APPLICATION**

In this subsection, we explore how the Online Fraud Detection Application functions in practical terms, including its key features and user interactions.

**Automated Data Processing:** Upon receiving transaction data, the application automatically processes it to identify potential fraud patterns.

**Machine Learning Integration:** Utilizes advanced machine learning models to analyze transaction data and predict fraudulent behavior.

**User-friendly Interface:** Offers an intuitive interface for users to interact with the system, view transaction details, and assess fraud risk.

**Instant Classification:** Classifies transactions in real-time as either fraudulent or legitimate, providing immediate insights to users.

**Feedback Integration:** Allows users to provide feedback on classification results, enabling iterative improvements to the system.

**Compliance Measures:** Adheres to industry regulations and security standards to ensure data privacy and compliance with legal requirements.

**Secure Data Handling:** Implements robust encryption and access controls to protect sensitive transaction data from unauthorized access.

**Scalability:** Designed to scale with increasing transaction volumes and evolving fraud patterns, ensuring reliability and performance.

**Continuous Improvement:** Incorporates user feedback and validation results to refine models and enhance fraud detection accuracy over time.

# CHAPTER 5: MACHINE LEARNING

## 5.1 INTRODUCTION

Machine learning is a subset of artificial intelligence (AI) focused on understanding data patterns and creating models. Its primary goal is to identify the structure within data and develop models that are comprehensible and useful to humans.

Machine learning is a subfield of computer science that differs from traditional computational methods. Unlike classical algorithms, which rely on explicitly defined instructions, machine learning enables computers to learn from data inputs and use statistical analysis to generate outputs within a specified range. This approach allows computers to develop models from sample data and automate decision-making processes based on the data inputs.

Machine learning benefits everyone who uses technology today. Social media platforms use face recognition to help users tag and share photos. Optical character recognition (OCR) technology converts text images into editable text. Recommendation systems use machine learning to suggest movies or TV shows based on user preferences. Additionally, consumers may soon be able to purchase self-driving cars that navigate using machine learning.

Machine learning is a rapidly evolving field. When working with or analyzing machine learning techniques, it's important to consider several key factors due to its dynamic nature.

The Data Science team specializes in studying various aspects of cardholder transactions to develop a model for detecting and preventing illegal or fraudulent activities. They analyze data such as transaction dates, user locations, product categories, transaction amounts, suppliers, and client behavior patterns. This data is then fed into a machine learning model trained to identify patterns and rules indicative of fraudulent transactions. By combining sophisticated algorithms with comprehensive data analysis, the team aims to enhance fraud detection capabilities and safeguard against financial losses.

The process involves several key steps:

**Data Collection**: The team gathers comprehensive data on cardholder transactions. This includes various aspects such as:

1. **Transaction Date**: The specific date and time of each transaction.
2. **User Zone**: The geographical location or region where the transaction took place.
3. **Product Category**: The type of products or services involved in the transaction.
4. **Transaction Amount**: The monetary value of each transaction**.**
5. **Supplier Information**: Details about the merchant or supplier involved in the transaction.
6. **Client Behavioural Tendencies**: Historical data on the client's usual spending patterns and behaviours.

**Data Integration:** All these data points are integrated to create a comprehensive dataset that captures the full context of each transaction.

**Model Training**: The integrated data is used to train a machine learning model. This involves:

1. **Pattern Recognition**: The model is trained to recognize patterns and correlations within the data that are indicative of normal and abnormal behaviours.
2. **Rule Identification**: The model identifies rules and anomalies that can suggest fraudulent activity.

**Fraud Detection**: Once trained, the model continuously analyzes new transactions in real-time or near-real-time. It applies the learned patterns and rules to:

1. **Detect Fraud**: Identify transactions that deviate from the established norms and are likely to be fraudulent.
2. **Prevent Fraud**: Flag suspicious transactions for further investigation or automatically block them to prevent loss.

By combining various aspects of transaction data and applying advanced machine learning techniques, the Data Science team aims to effectively identify and mitigate fraudulent activities, ensuring secure and trustworthy financial transactions.

It is crucial for credit card companies to identify fraudulent transactions to ensure that customers are not charged for items they did not purchase. It involves modeling a dataset to illustrate how machine learning techniques can identify fraudulent transactions effectively.

Machine learning algorithms are employed to analyses all the authorized transactions and report the suspicious ones. These reports are investigated by professionals who contact the cardholders to confirm if the transaction was genuine or fraudulent. The investigators provide a feedback to the automated system which is used to train and update the algorithm to eventually improve the fraud-detection performance over time.

Machine learning is a subfield of AI that focuses on developing algorithms and models that give computers the ability to learn from data, identify patterns from within the data, and make decisions based on their learnings.

Machine learning (ML) has revolutionized the field of fraud detection by enabling more accurate, efficient, and adaptive identification of fraudulent activities across various industries

## 5.2 WHY MACHINE LEARNING??

Here are some factors for why Machine Learning techniques are so popular and widely used in industries for detecting frauds:

**Speed**: Machine Learning (ML) is widely used for its rapid computation capabilities, quickly analyzing data and identifying new patterns. In contrast, rule-based fraud prevention systems rely on manually written rules, which are time-consuming to create and update for various scenarios. ML-based fraud detection algorithms excel by learning and detecting new patterns automatically, achieving results much faster than rule-based systems.

**Scalability:** As more and more data is fed into the Machine Learning-based model, the model becomes more accurate and effective in prediction. Rule-based systems don’t evolve by themselves as professionals who developed these systems must write these rules meeting various circumstances. But for Machine Learning based algorithms, a dedicated team of Data Science professionals must be involved in making sure these algorithms are performing as intended.

**Efficiency:** Machine Learning algorithms perform the redundant task of data analysis and try to find hidden patterns repetitively. Their efficiency is better in giving results in comparison with manual efforts. It avoids the occurrence of false positives which counts for its efficiency. Due to their efficiency in detecting these patterns, the specialists in Fraud detection could now focus on more advanced and complex patterns, leaving the low or moderate level problems to these Machine Learning based algorithms.

Machine learning significantly enhances fraud prevention and detection through various sophisticated techniques:

**Anomaly Detection**: Algorithms identify deviations from normal behavior in transactional data, flagging unusual activities that could indicate fraud by training on historical data to distinguish between legitimate and suspicious transactions.

**Risk Scoring**: Models assign risk scores to transactions or user accounts based on factors like transaction amount, location, frequency, and past behavior. Higher scores indicate a higher likelihood of fraud, allowing organizations to prioritize investigations effectively.

**Network Analysis**: Techniques like graph analysis uncover fraud networks by examining relationships between entities (users, accounts, devices) and identifying unusual connections or clusters indicative of collusion.

**Text Analysis:** Algorithm analyze unstructured text data from emails, social media, or reviews to detect patterns or keywords.

**Identity Verification:** Models verify user-provided information, such as identification documents or facial recognition data, to confirm identities and prevent identity theft.

**Adaptive Learning**: Machine learning's ability to learn and adapt allows models to stay current with evolving fraud tactics by retraining on new data, improving their capability to detect emerging fraud patterns.

**Real-Time Fraud Detection**: Machine learning systems can process and analyze transactions as they occur, allowing for immediate detection and response to potentially fraudulent activities. This is crucial for industries like banking and e-commerce, where the speed of detection can prevent significant losses.

**Behavioral Analysis**: Machine learning models can monitor and learn from user behavior to establish a baseline of normal activity. Deviations from this baseline can trigger alerts for potential fraud. For instance, if a user suddenly makes large purchases in a foreign country, the system can flag this as suspicious.

**Major Benefits of using Machine Learning**:

**Cost Effectiveness:** Machine learning (ML) automates the fraud detection process, significantly reducing costs related to manual detection efforts. By minimizing the need for extensive human labor, technology investments, and time spent on analyzing transactions, organizations can allocate resources more efficiently. This automation leads to lower overall expenses in the fight against fraud.

**Accuracy**: ML algorithms are trained on vast amounts of data, enabling them to identify patterns and anomalies that are beyond human capability. This results in a significant reduction in false positives (legitimate transactions incorrectly flagged as fraud) and false negatives (fraudulent transactions missed). The enhanced accuracy of ML models ensures more reliable fraud detection compared to traditional, manual methods.

**Relentlessness:** Unlike human analysts, ML systems can operate continuously, analyzing data 24/7 without fatigue. The ability of ML algorithms to process large volumes of data consistently improves their performance over time. This relentless processing capability ensures that fraud detection is both persistent and scalable, providing constant vigilance against fraudulent activities.

**Massive data processing:**Humans struggle to understand vast amounts of data and analyze the patterns. The more data the Fraud Detection Machine Learning model receives, the better it can understand the data and analyze the fraud activities.

**Faster and accurate –**Data analysis is done in seconds once the ML model suitable for the business needs sets the action. Furthermore, the accuracy of ML models is far better than humans, and better predictions are possible with machine learning.

**Scalable –**Machine Learning methods offer better performance with the growth in datasets. Though the model needs constant updates as fraudsters regularly find ways, the risk and efficiency are far better than rule-based systems.

## 5.3 MODELS USED IN THE PROJECT

In machine learning, tasks are typically divided into major categories based on how data is acquired and how the system responds to it. These categories help in organizing and applying machine learning techniques effectively.

**Logistic Regression:** Logistic Regression is the classification of algorithm into multiple categorical values. It includes the use of multiple independent variables which are used to predict a particular outcome of a variable which is dependent on all the independent variables use to train the model.

It predicts that the probability of a given data belongs to the particular category or not. Logistic Regression is similar to linear regression, it predicts a target field rather than a numeric one Zanin et al. (2018). Like predicting True or False, successful or unsuccessful in our case it is fraudulent or non-fraudulent. The figure below explains the logistic regression:

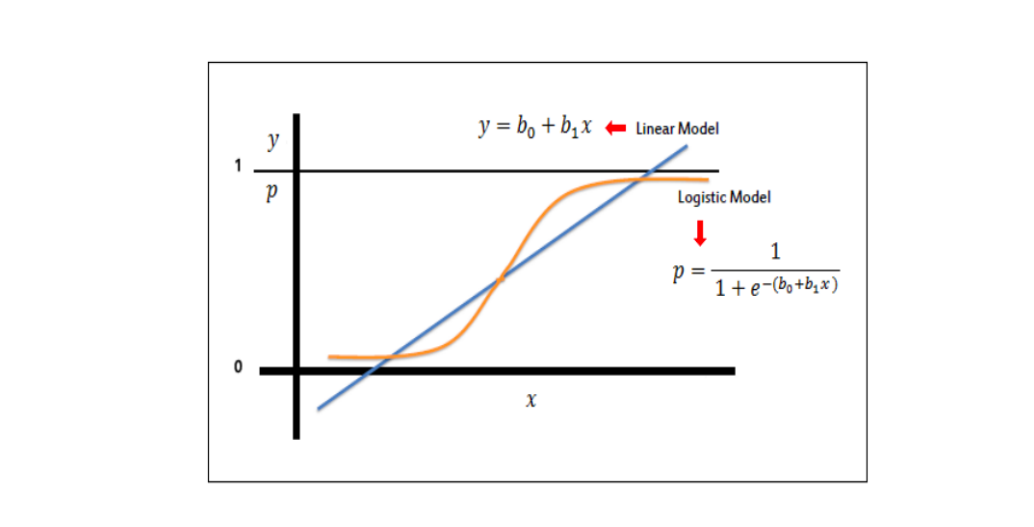


Figure 5.3.1 Logistic Regression Curve

[Logistic Regression](https://intellipaat.com/blog/what-is-logistic-regression/) is a supervised learning technique that is used when the decision is categorical. It means that the result will be either ‘fraud’ or ‘non-fraud’ if a transaction occurs.

**Use Case**:  Let us consider a scenario where a transaction occurs and we need to check whether it is a ‘fraudulent’ or ‘non-fraudulent’ transaction. There will be given set of parameters that are checked and, on the basis of the probability calculated, we will get the output as ‘fraud’ or ‘non-fraud.’

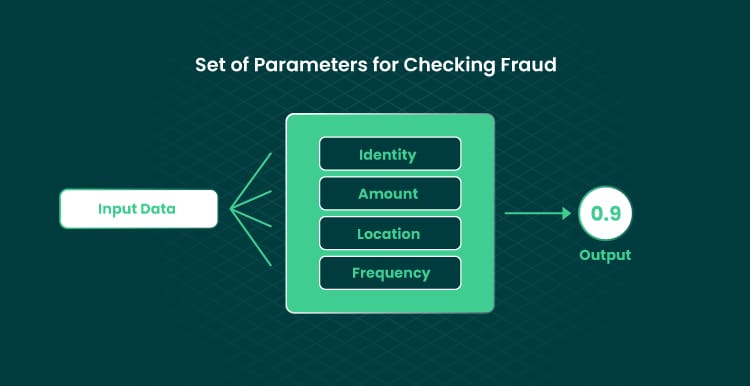


Figure 5.3.2 Logistic Regression Working

In this diagram, we can see that the probability calculated is 0.9. This means that there is a 90 percent chance that the transaction is ‘genuine’ and there is a 10 percent probability that it is a ‘fraud’ transaction.

Logistic Regression is a fundamental classification algorithm utilized in machine learning for predicting the probability of a categorical outcome. Unlike linear regression, which predicts continuous numeric values, logistic regression is specifically designed for binary classification tasks, where the outcome is either one of two categories, such as "fraudulent" or "non-fraudulent" transactions.

At its core, logistic regression models the relationship between one or more independent variables (features) and a binary outcome by estimating the probability that a given data point belongs to a particular category. The output of logistic regression is a probability score ranging from 0 to 1, representing the likelihood of the event being in the positive class (e.g., "fraudulent"). This probability is then converted into a binary decision using a predefined threshold; typically, if the probability is greater than or equal to 0.5, the event is classified as positive, otherwise negative.

One of the key advantages of logistic regression is its simplicity and interpretability. The algorithm produces coefficients for each independent variable, indicating the direction and strength of their influence on the outcome. This allows analysts to easily understand the impact of different features on the classification decision, making logistic regression a valuable tool for both prediction and inference tasks.

In practice, logistic regression is widely used in various domains, including finance, healthcare, marketing, and more. In the context of fraud detection systems, logistic regression can effectively analyze transactional data to identify suspicious activities and flag potentially fraudulent transactions for further investigation.

For instance, consider a scenario where a financial institution needs to determine whether a credit card transaction is fraudulent or legitimate. By examining features such as transaction amount, location, time of day, and previous transaction history, logistic regression can calculate the probability of fraud and help the institution make informed decisions to mitigate risks and protect its customers.

Overall, logistic regression serves as a powerful and interpretable tool for binary classification tasks, providing valuable insights and aiding decision-making processes across various industries.

**Random Forest Classifier:** Random Forest is a kind of Supervised Machine Learning Algorithm that is often used in classification and regression issues. It constructs decision trees from several samples and uses their majority vote for classification and average for regression.

The Random Forest algorithm is effective for distinguishing between authentic and fraudulent transactions. It builds multiple decision trees using different subsets of data and features, creating an ensemble for final classification. By analyzing various transaction variables like amount, frequency, location, and user behaviour, the algorithm identifies suspicious activities. A key advantage of Random Forest in fraud detection is its ability to manage imbalanced datasets, where fraudulent transactions are much fewer than genuine ones.

Random Forest effectively addresses class disparity in fraud detection using strategies like class weighting and modified decision criteria, enhancing accuracy. It handles high-dimensional data well, making it ideal for real-time online fraud detection systems. The algorithm is resistant to overfitting and noise, ensuring reliable performance in dynamic contexts. Additionally, its interpretability allows fraud analysts to gain insights from feature importance scores, identifying the most significant transaction attributes impacting fraud detection.

The random forest model is made up of many decision trees that are all put together to solve classification problems. It uses methods like feature randomization and bagging to build each tree. This makes a forest of trees that don’t have anything in common with each other. Every tree in the forest is based on a basic training sample, and the number of trees in the forest has a direct impact on the results.

Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. Then, it collects the votes from different decision trees to decide the final prediction.

[Random Forest](https://intellipaat.com/blog/what-is-random-forest-algorithm-in-python/) uses a combination of decision trees to improve the results. Each decision tree checks for different conditions. They are trained on random datasets and, based on the training of the decision trees, each tree gives the probability of the transaction being ‘fraud’ and ‘non-fraud.’ Then, the model predicts the result accordingly.

Moreover, Random Forest excels in handling noisy and correlated features commonly found in fraud detection datasets. By utilizing feature randomization and bagging techniques, Random Forest mitigates the impact of irrelevant or redundant features, enhancing the robustness of the model against noise and improving generalization performance. Additionally, the ensemble nature of Random Forest allows for parallel processing of multiple decision trees, making it computationally efficient for analyzing large-scale transaction data in real-time. This scalability makes Random Forest well-suited for deployment in high-volume transaction environments, such as online payment systems and e-commerce platforms, where timely detection of fraudulent activities is crucial for minimizing financial losses and maintaining customer trust. Furthermore, the interpretability of Random Forest enables fraud analysts to gain actionable insights into the underlying patterns of fraudulent behavior, facilitating the development of proactive strategies and adaptive countermeasures to combat evolving fraud tactics effectively. Overall, the versatility, scalability, and interpretability of Random Forest make it a powerful tool for detecting and preventing fraudulent activities across various industries and applications.

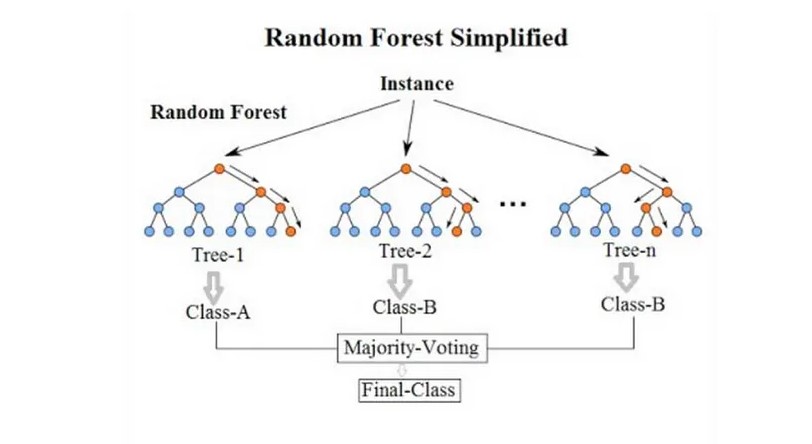


Figure 5.3.3 Random Forest

**Use Case**: Let’s consider a scenario where a transaction is made. Now, we will see how the random forest in Machine Learning is used in fraud detection algorithms.

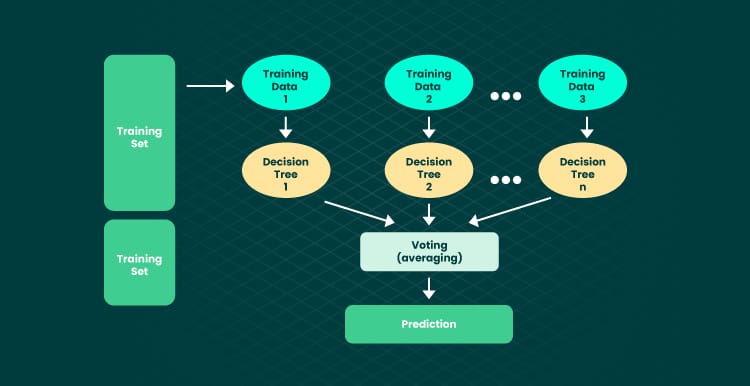


Figure 5.3.4 Random Forest Working

When a request for a transaction is given to the model, it checks for the information like the credit/debit card number, location, date, time, the IP address, the amount, and the frequency of the transaction. All this dataset is fed as an input into the fraud detection algorithm. Then this fraud detection algorithm selects variables from the given dataset that help in splitting up of the dataset. The below diagram shows the splitting up of the dataset into multiple decision trees.

So, the sub-trees consist of variables and the conditions to check those variables for an authorized transaction.

After checking all the conditions, all the sub-trees will give the probabilities for a transaction to be ‘fraud’ and ‘non-fraud.’ Based on the combined result, the model will mark the transaction as ‘fraud’ or ‘genuine.’

**Random Forest algorithm for efficient detection of fraudulent activities:** The Random Forest algorithm offers several benefits in the detection of fraudulent activities. By leveraging the ability to handle large amounts of Dataset.

Random Forest can analyze various features associated with the Transaction this will help in detecting the patterns that are indicative of fraudulent activities.

This algorithm can able to identify anomalies in the transaction such as unusual purchasing patterns, huge amounts, inconsistent IP addresses, or typical user behaviors.

Moreover, Random Forest excels in handling noisy and correlated features commonly found in fraud detection datasets. By utilizing feature randomization and bagging techniques, Random Forest mitigates the impact of irrelevant or redundant features, enhancing the robustness of the model against noise and improving generalization performance. Additionally, the ensemble nature of Random Forest allows for parallel processing of multiple decision trees, making it computationally efficient for analyzing large-scale transaction data in real-time. This scalability makes Random Forest well-suited for deployment in high-volume transaction environments, such as online payment systems and e-commerce platforms, where timely detection of fraudulent activities is crucial for minimizing financial losses and maintaining customer trust. Furthermore, the interpretability of Random Forest enables fraud analysts to gain actionable insights into the underlying patterns of fraudulent behavior, facilitating the development of proactive strategies and adaptive countermeasures to combat evolving fraud tactics effectively. Overall, the versatility, scalability, and interpretability of Random Forest make it a powerful tool for detecting and preventing fraudulent activities across various industries and applications.

Furthermore, Random Forest's ability to handle imbalanced datasets is particularly advantageous in fraud detection applications where the occurrence of fraudulent transactions is relatively rare compared to legitimate ones. Through techniques such as class weighting and modified decision criteria, Random Forest effectively addresses class disparity, ensuring that the model is not biased towards the majority class and accurately identifies fraudulent instances even in highly imbalanced data distributions. This capability is crucial for minimizing false positives and false negatives, thereby reducing the risk of overlooking fraudulent activities or flagging legitimate transactions erroneously. By striking a balance between sensitivity and specificity, Random Forest provides a reliable solution for detecting fraudulent transactions with high precision and recall, ultimately contributing to enhanced security and trust in online payment systems and financial transactions.

**Decision Tree Classifier:** A decision tree is a non-parametric supervised learning approach that may be used for classification as well as regression applications. It has a tree structure that is hierarchical and consists of a root node, branches, internal nodes, and leaf nodes.

Decision tree is a supervised machine learning algorithm which uses a combination of rules to make a particular decision, just like a human being. The motive behind decision tree is that one uses the dataset features to create yes or no questions and split the dataset until and unless we isolate all the datapoints those belong to each class.

 Decision Tree algorithms in fraud detection are used where there is a need for the [**classification**](https://intellipaat.com/blog/tutorial/machine-learning-tutorial/classification-machine-learning/) of unusual activities in a transaction from an authorized user. These algorithms consist of constraints that are trained on the dataset for classifying fraud transactions.

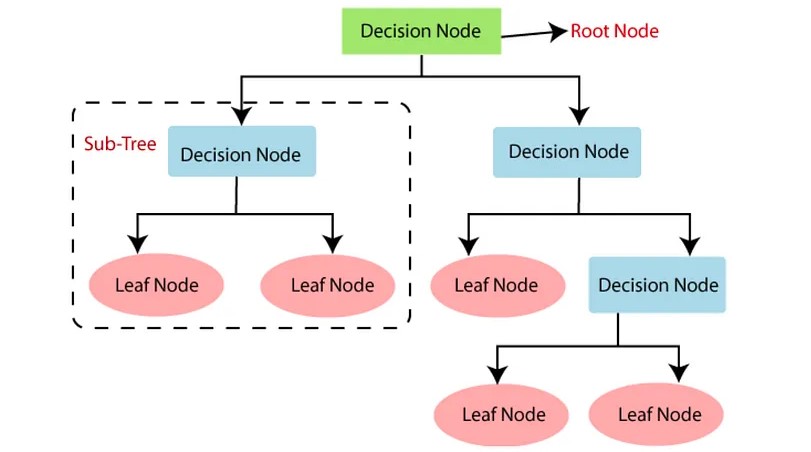


Figure 5.3.5 Decision Tree

**Use Case**: Let us consider a scenario where a user makes transactions. We will build a decision tree to predict the probability of fraud based on the transaction made.

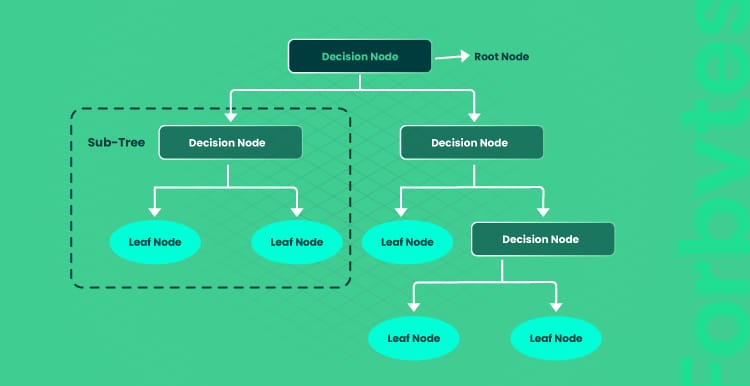


Figure 5.3.6 Decision Tree Working

First, in the decision tree, we will check whether the transaction is greater than ₹50,000. If it is ‘yes,’ then we will check the location where the transaction is made. And if it is ‘no,’ then we will check the frequency of the transaction. After that, as per the probabilities calculated for these conditions, we will predict the transaction as ‘fraud’ or ‘non-fraud.’

Here, if the amount is greater than ₹50,000 and location is equal to the IP address of the customer, then there is only a 25 percent chance of ‘fraud’ and a 75 percent chance of ‘non-fraud.’

Similarly, if the amount is greater than ₹50,000 and the number of locations is greater than 1, then there is a 75 percent chance of ‘fraud’ and a 25 percent chance of ‘non-fraud.’This is how a decision tree in Machine Learning helps in creating fraud detection algorithms.

In addition to its interpretability and ease of understanding, Decision Tree classifiers offer several advantages in fraud detection scenarios. One notable benefit is their ability to handle both numerical and categorical features without requiring extensive preprocessing or feature engineering. Decision trees naturally accommodate mixed data types and can effectively partition the feature space based on the values of different attributes, allowing for flexible and adaptive decision-making. Furthermore, Decision Tree algorithms are inherently resilient to outliers and missing values, as they do not rely on assumptions about the underlying distribution of the data. This robustness enables decision trees to maintain performance in the presence of noisy or incomplete data, making them suitable for real-world applications where data quality may be variable. Additionally, decision trees facilitate transparent and actionable insights into the decision-making process, empowering fraud analysts to understand the logic behind classification decisions and identify potential areas for refinement or improvement in fraud detection strategies. Overall, the versatility, interpretability, and robustness of Decision Tree classifiers make them a valuable tool for detecting and mitigating fraudulent activities across diverse industries and domains. Moreover, Decision Tree classifiers are well-suited for detecting complex patterns and interactions among features in fraud detection datasets. Unlike linear models that assume linear relationships between predictors and outcomes, decision trees can capture nonlinear relationships and intricate decision boundaries, making them highly adaptable to the nonlinear nature of fraudulent behaviors. By recursively partitioning the feature space into homogeneous regions, decision trees can identify subtle deviations and anomalies indicative of fraudulent activities, even in highly dynamic and evolving environments. This capability allows decision trees to uncover hidden patterns and detect emerging fraud tactics that may not be captured by traditional statistical methods. Additionally, decision trees offer flexibility in model complexity, allowing analysts to control the depth and complexity of the tree to balance between bias and variance, thereby optimizing performance for specific fraud detection tasks. This flexibility enables decision trees to adapt to changing fraud landscapes and evolving attack strategies, ensuring robust and reliable detection of fraudulent activities over time.

# CHAPTER 6: RESULT AND OUTPUTS

## 6.1 INTRODUCTION

This section serves as a detailed presentation of the significant findings and outcomes derived from our project. This section is structured to provide a clear understanding of the project's goals, the methodologies employed in implementation, and the results achieved. It encapsulates the journey from the inception of the project through to the final outcomes, offering a transparent view of the work done and its impact. I will be using three sections as mentioned below to show how we have worked on this project. Below we have elaborated in details what each section will explain.

**Goals:** The primary objectives of this project were established to guide our efforts and measure success. The foremost goal was to develop a reliable and accurate model for predicting fraudulent transactions, a critical need in today's digital financial landscape. Additionally, the project aimed to create a user-friendly web application that would enable users to interact with the model and receive fraud predictions in real-time, thereby enhancing the accessibility and usability of our solution. Ensuring data security and privacy throughout the process was also a key objective, given the sensitive nature of transaction data. By setting these goals, we aimed to address specific challenges and deliver measurable outcomes that align with the project's overall purpose.

**Implementation:** The implementation phase was critical to achieving our project goals. This phase began with the collection and preprocessing of relevant transaction data from multiple sources. This step was essential to ensure that the data used for model training was accurate, clean, and consistent. We then moved on to the development of the predictive model. This involved selecting appropriate machine learning algorithms and training them on our dataset. Extensive validation was performed to ensure high predictive accuracy, and model parameters were fine-tuned for optimal performance.

Simultaneously, we designed and developed a web application using Flask, a lightweight WSGI web application framework in Python. The web application was designed to provide a seamless user experience, allowing users to input transaction data and receive fraud predictions instantly. We also integrated the predictive model into the web application, ensuring that the predictions were both accurate and timely. Furthermore, significant efforts were made to implement robust security measures to protect user data and ensure privacy, adhering to best practices in data security and privacy regulations.

**Outputs:** The outputs of our project highlight the success of our implementation efforts. The final predictive model demonstrated a high accuracy rate, showcasing its reliability in predicting fraudulent transactions. Detailed performance metrics such as precision, recall, and F1-score further underscored the model's strengths and areas for improvement, providing a comprehensive view of its effectiveness.

The web application developed as part of this project featured a user-friendly interface that facilitated easy interaction with the predictive model. Users could input transaction data and receive real-time fraud predictions, along with detailed reports and alerts. This real-time interaction greatly enhances the practical usability of our solution.

Additionally, the project successfully implemented robust security protocols to protect sensitive user data, ensuring compliance with data privacy regulations throughout the project lifecycle. This commitment to data security and privacy is a testament to the project's adherence to ethical standards and regulatory requirements.

## 6.2 GOALS

The primary objective of our project was to create a highly effective, accurate, and robust model for predicting fraudulent transactions within financial datasets. The increasing prevalence of online transactions has heightened the need for enhanced security measures to protect against financial fraud. Consequently, the overarching goal of this project was to develop a model that could reliably identify fraudulent activities, thereby minimizing financial losses and fostering greater trust in digital financial systems.

To achieve this overarching goal, we set several specific objectives:

**Algorithm Exploration and Comparison:** The selection of machine learning algorithms is critical in the process of fraud detection. In this project, we focused on three different algorithms: Random Forest, Logistic Regression, and Decision Tree. Each algorithm was chosen for its unique strengths and suitability for handling classification problems, particularly those involving anomaly detection.

1. **Random Forest:** This ensemble learning method constructs multiple decision trees during training and outputs the mode of the classes (classification) of the individual trees. Its ability to handle large datasets with higher dimensionality makes it particularly suitable for fraud detection.
2. **Logistic Regression:** A statistical model that uses a logistic function to model a binary dependent variable. Despite its simplicity, it is widely used for its efficiency and effectiveness in binary classification tasks.
3. **Decision Tree:** A tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. This algorithm is intuitive and easy to interpret, making it useful for understanding the decision-making process.

**Data Resampling Techniques:** Fraud detection datasets often exhibit significant class imbalances, with fraudulent transactions being much rarer than legitimate ones. This imbalance can skew the performance of machine learning models, leading to biased predictions. To address this issue, we employed both undersampling and oversampling techniques.

1. **Under sampling:** This technique involves reducing the number of non-fraudulent transaction samples to match the number of fraudulent ones. While this can help balance the dataset, it can also result in a loss of valuable information, potentially affecting the model's performance.
2. **Over sampling:** Conversely, oversampling involves increasing the number of fraudulent transaction samples, either by duplicating existing ones or generating synthetic samples using techniques like SMOTE (Synthetic Minority Over-sampling Technique). This approach helps ensure that the model is exposed to a sufficient number of fraudulent transactions during training.

**Comprehensive Model Training:** To maximize the effectiveness of our models, we trained each algorithm with both undersampled and oversampled datasets. This exhaustive approach allowed us to explore every possible configuration, ensuring that we could identify the most effective model for detecting fraud. By doing so, we aimed to create a robust model capable of handling the inherent challenges of fraud detection.

**Performance Evaluation and Selection:** After training the models, we conducted a rigorous evaluation using a variety of performance metrics. These included accuracy, precision, recall, and F1-score. Accuracy measures the overall correctness of the model, while precision indicates the proportion of true positive results among all positive predictions. Recall measures the proportion of true positives identified among all actual positives, and the F1-score provides a balanced measure of both precision and recall.

1. **Accuracy:** Although commonly used, accuracy alone is not sufficient in fraud detection due to the class imbalance. A model could achieve high accuracy by simply predicting all transactions as non-fraudulent.
2. **Precision and Recall:** These metrics provide more insight into the model’s performance in identifying fraudulent transactions. High precision indicates a low false positive rate, while high recall indicates a low false negative rate.
3. **F1-Score:** The harmonic mean of precision and recall, offering a single metric that balances the trade-off between the two.

By evaluating these metrics, we were able to determine which combination of algorithm and resampling technique provided the best performance in terms of detecting fraudulent transactions.

**Implementation of the Best Model:** Based on the evaluation results, we selected the model that demonstrated the highest effectiveness in predicting fraudulent transactions. This model was then integrated into a user-friendly web application, designed to provide real-time fraud detection. The application not only serves as a tool for end-users but also as a practical demonstration of the model's capabilities.

**Ensuring Data Security and Privacy:** Given the sensitive nature of financial transaction data, maintaining data security and privacy was a top priority throughout the project. We implemented robust security measures to ensure compliance with privacy regulations and protect user information. This included encrypting data, implementing secure authentication mechanisms, and ensuring that our data processing pipelines adhered to industry best practices.

**Future Directions:** While our project achieved significant milestones, there is always room for improvement and expansion. Future work could explore additional machine learning algorithms, including deep learning techniques, to further enhance the accuracy and robustness of fraud detection models. Additionally, incorporating real-time data streams and continuously updating the model with new data could improve its adaptability to evolving fraud patterns.

By setting and pursuing these comprehensive goals, our project aimed to address the multifaceted challenges of fraud detection. Leveraging advanced machine learning techniques and maintaining a strong focus on data security and privacy, we developed a model that not only performs well in detecting fraudulent transactions but also holds practical, real-world applicability. The outcomes of this project contribute to the field of fraud detection and lay the groundwork for future advancements in financial security technologies.

Furthermore, our project was guided by the imperative to adapt to the rapidly evolving landscape of fraud tactics. Fraudsters continually develop more sophisticated methods to bypass traditional detection systems, making it crucial for our model to remain dynamic and adaptive. To address this, we incorporated a continuous learning approach, where the model is periodically retrained with the latest transaction data. This ensures that our detection capabilities stay current and effective against emerging threats. Additionally, we explored the integration of anomaly detection techniques alongside our primary models. These techniques are designed to identify outliers or unusual patterns in transaction data that may not be captured by conventional methods. By combining machine learning algorithms with advanced data resampling strategies and continuous learning, our goal was to create a holistic fraud detection system. This system not only excels in accuracy and precision but also provides a resilient defense against the ever-changing tactics of fraudsters. The implementation of this comprehensive approach underscores our commitment to leveraging state-of-the-art technology to safeguard financial transactions, ensuring both the security and confidence of users in digital financial systems. The success of this project sets a precedent for future work, where continuous improvement and adaptation are key to maintaining robust fraud detection in an increasingly digital world.

## 6.3 IMPLEMENTATION

## The implementation of the Online Fraud Detection system involved several stages, including training the dataset with different machine learning models, integrating the trained models into a Flask web application, and developing the frontend interface using HTML and CSS. Here's a detailed overview of the implementation process:

## Training the Dataset with Different Models: Before integrating the trained models into the Flask application, the dataset was trained using various machine learning algorithms to identify patterns indicative of fraudulent transactions.

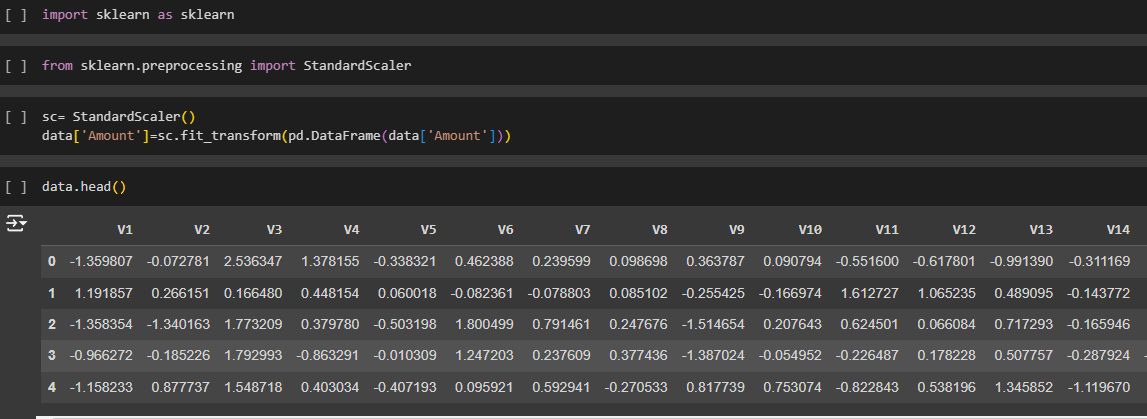


Figure 6.3.1 Training Dataset

**The following machine learning models were trained and evaluated:** In developing our fraud detection system, we leveraged three powerful machine learning models: Logistic Regression, Random Forest, and Decision Tree. Each of these models brings unique strengths to the table, enhancing the system's overall effectiveness in identifying fraudulent transactions. Logistic Regression, known for its simplicity and efficiency, excels in situations where the relationship between input features and the probability of fraud is linear. It provides clear probabilistic interpretations, making it useful for understanding the likelihood of fraudulent activity. Random Forest, on the other hand, offers robustness and high accuracy by constructing multiple decision trees and aggregating their results. This model is particularly adept at handling complex interactions and imbalanced datasets, which are common in fraud detection scenarios. Lastly, the Decision Tree model, with its intuitive tree-like structure, offers excellent interpretability and is effective in capturing non-linear relationships between features. By utilizing these three complementary models, our system not only improves prediction accuracy but also provides a comprehensive framework for detecting and understanding fraudulent activities, ensuring a robust and reliable fraud detection mechanism.

## Pickle File: After training the dataset, the best-performing machine learning model was selected, and its parameters were serialized using Python's pickle module. The serialized model, along with the necessary preprocessing steps and feature transformations, was saved as a pickle file. This file served as the trained model's representation and could be loaded into the Flask application for real-time prediction.

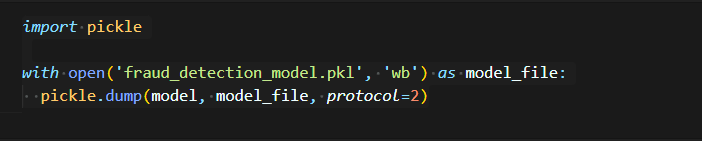


Figure 6.3.2 Pickle Code Snippet

To deploy the trained fraud detection model for real-time prediction, we saved it using the pickle module in Python. The code snippet below demonstrates how the model was serialized and stored as a binary file named fraud\_detection\_model.pkl:

This process allows us to preserve the trained model's state, including its architecture, parameters, and learned patterns, enabling seamless integration into the Flask web application for real-time fraud detection.

**3. Flask Application:**

## The Flask web application was developed to provide a user-friendly interface for predicting fraud in real-time transactions. The architecture of the Flask application included:

## Backend (Flask Server): The Flask server served as the backend of the application, handling user requests, invoking the machine learning model for prediction, and returning the results to the frontend.

## Model Integration: The trained machine learning model, loaded from the pickle file, was integrated into the Flask server. This allowed the server to leverage the predictive capabilities of the model to classify transactions as fraudulent or legitimate.

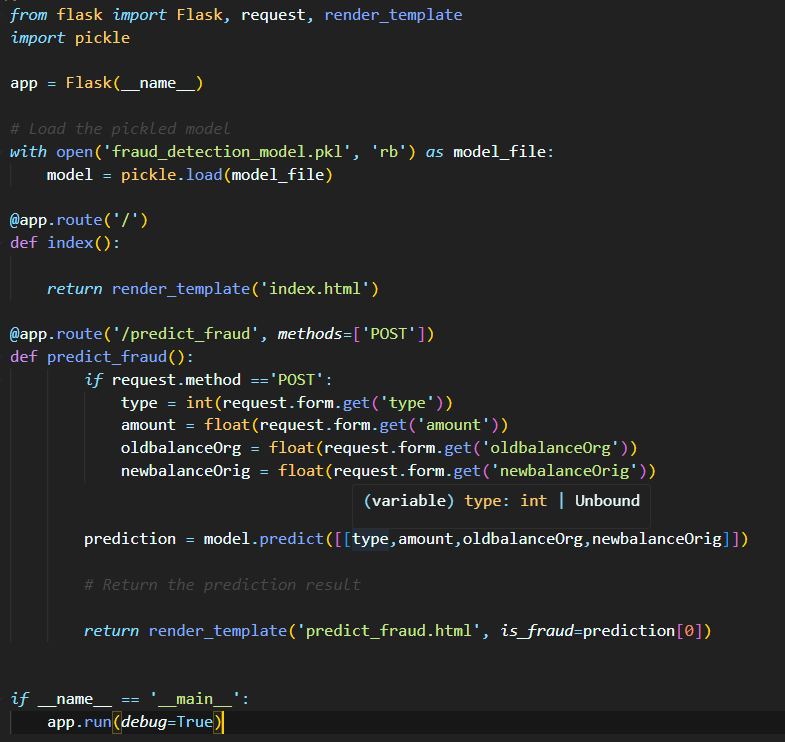


Figure 6.3.3 Model Loading

We developed a Flask web application to integrate our trained fraud detection model for real-time prediction of fraudulent transactions.

**Flask Application Setup:** We initialized a Flask application named app and loaded the pre-trained model using the pickle module.

Routes Definition: We defined two routes:

'/': Renders the home page HTML template where users can input transaction details.

**'/predict\_fraud':** Handles form submission, makes predictions using the model, and renders the prediction result.

**Model Loading:** The pre-trained model file (fraud\_detection\_model.pkl) was loaded into memory to enable real-time predictions.

**Application Execution:** The Flask application was executed in debug mode, allowing it to run locally on our machine for development and debugging purposes.

## Frontend Interface (HTML and CSS): The frontend interface of the Flask application was developed using HTML for structure and CSS for styling. The interface included input fields for users to enter transaction details, such as transaction type, amount, old balance, and new balance. Additionally, a submit button was provided to trigger the fraud prediction process.

We utilized HTML templates to define the structure of our web pages. The following HTML templates were used:

**index.html:** This template represents the home page where users input transaction details. It contains form elements for type, amount, old balance, and new balance.

**predict\_fraud.html:** This template displays the prediction result after form submission. It dynamically shows whether the transaction is predicted as fraudulent or not. The HTML templates were integrated into the Flask application using the render\_template function. This allowed us to dynamically generate HTML content and serve it to users through the Flask routes.

The UI was designed with responsiveness in mind to ensure optimal display across various devices and screen sizes. CSS media queries were utilized to adapt the layout and styling based on viewport dimensions.

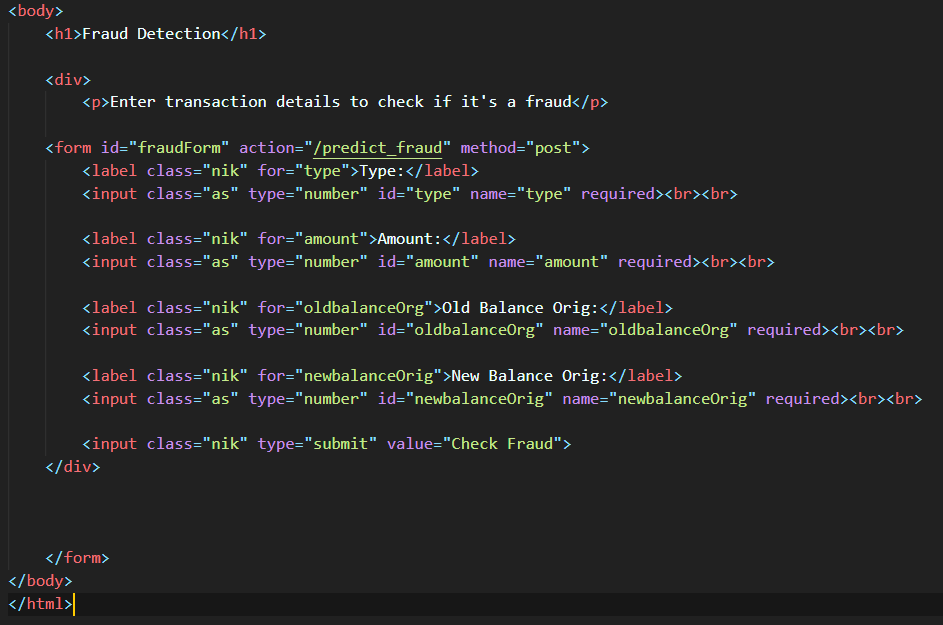
****

Figure 6.3.4 User Test Interface

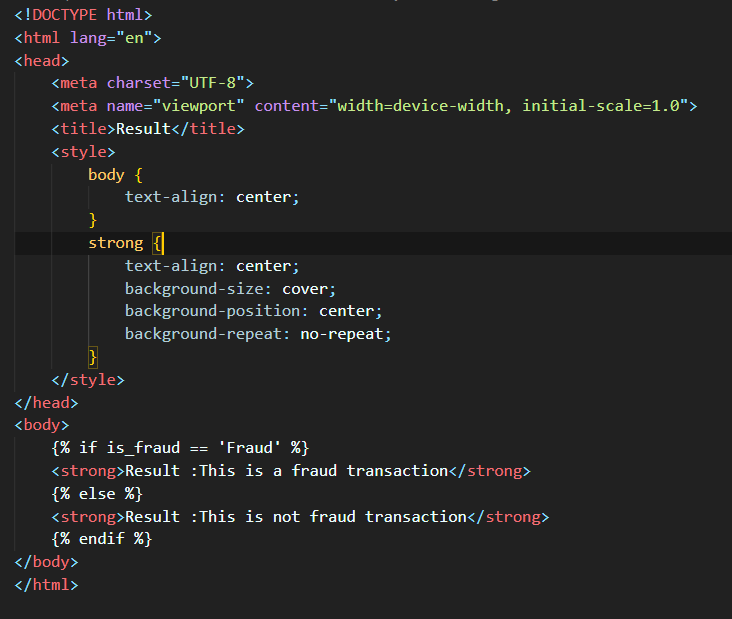
****

Figure 6.3.5 Result Interface

## Challenges and Solutions: Throughout the implementation process, several challenges were encountered, including:

## Integration Complexity: Integrating the trained machine learning model into the Flask application required careful consideration of dependencies, data formats, and compatibility issues. Thorough testing and documentation of integration steps helped address these challenges and ensure smooth integration.

## User Interface Design: Designing an intuitive and visually appealing frontend interface posed challenges in terms of layout, usability, and responsiveness. Iterative design improvements based on user feedback helped address these challenges and enhance the UI's effectiveness.

## By overcoming these challenges and implementing the various components of the Credit Card Fraud Detection system, a functional and reliable application was developed, capable of detecting fraudulent transactions in real-time.

## 6.4 OUTPUTS

Once the machine learning models have been trained and evaluated using appropriate metrics, the next step is to generate outputs that provide insights and actionable information. These outputs can take various forms, including classification results, probability scores, and feature importance rankings, depending on the specific requirements of the application. In the context of fraud detection, the outputs of the trained models play a crucial role in identifying and flagging suspicious transactions for further investigation.

One of the primary outputs of fraud detection models is the classification results, which indicate whether a transaction is predicted to be fraudulent or non-fraudulent based on the learned patterns and decision boundaries. These binary classification outputs enable businesses and financial institutions to quickly identify potentially fraudulent activities and take appropriate action to mitigate risks and prevent financial losses. Additionally, the probability scores generated by the models provide valuable insights into the confidence level of the predictions, allowing stakeholders to prioritize and allocate resources effectively for fraud prevention and detection efforts.

Moreover, feature importance rankings highlight the most influential factors contributing to the classification decisions of the models. By analyzing feature importance scores, fraud analysts can gain deeper insights into the underlying patterns of fraudulent behavior and identify the key indicators or red flags associated with fraudulent transactions. This information can inform the development of more targeted fraud prevention strategies and help businesses proactively address emerging fraud threats.

Furthermore, visualizations such as confusion matrices, ROC curves, and precision-recall curves can be used to assess the performance of the models and visualize their predictive capabilities. These visual outputs provide a comprehensive overview of the model's accuracy, sensitivity, specificity, and other performance metrics, enabling stakeholders to evaluate the effectiveness of the fraud detection system and make informed decisions about its deployment and optimization.

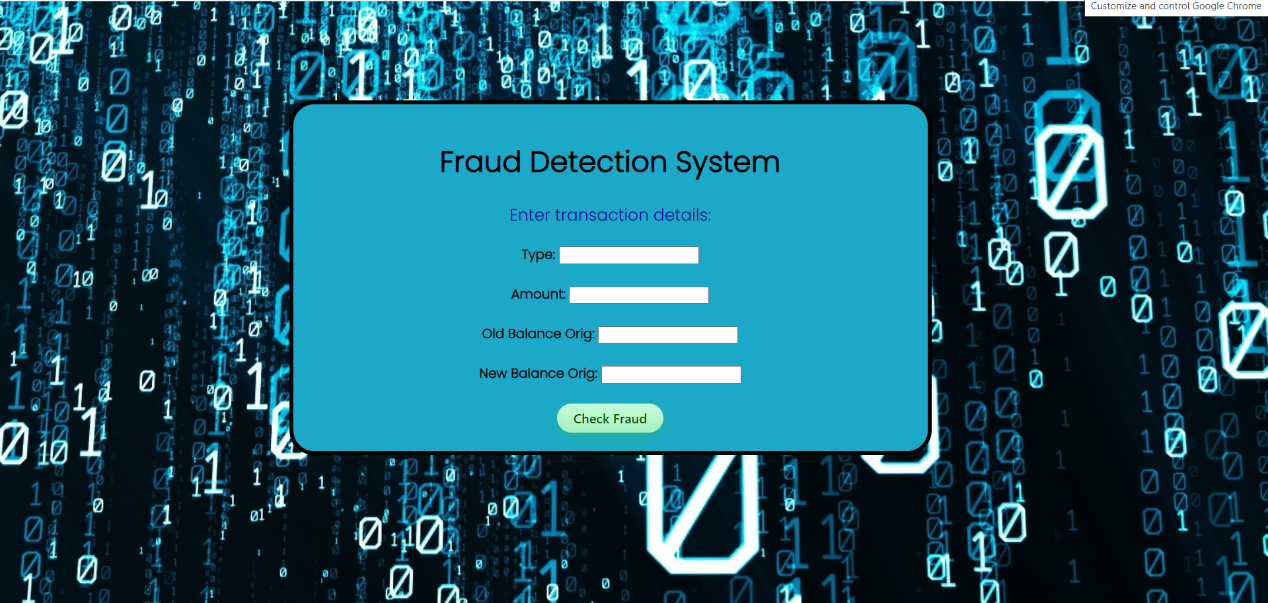


Figure 6.4.1 GUI Interface

The attached photograph showcases the sophisticated user interface of our innovative fraud detection system, meticulously designed using a combination of HTML, CSS, and Python. One of its standout features is the harmonious blend of several shades of blue, meticulously chosen to enhance visual appeal and captivate users' attention. This carefully curated color scheme not only adds aesthetic charm but also reinforces the interface's professional and trustworthy image, instilling confidence in users and fostering a positive user experience.

Through this visually striking interface, users are invited into a world of intuitive interaction and seamless functionality, where accessing advanced fraud detection capabilities is as effortless as it is efficient. Whether users are navigating dropdown menus, inputting transaction details, or initiating the fraud detection process with a single click, the interface's user-centric design ensures a smooth and intuitive experience every step of the way.

Moreover, the interface's responsive layout and adaptive design ensure consistent performance and usability across a variety of devices and screen sizes, from desktop computers to tablets and smartphones. This versatility allows users to access our fraud detection system anytime, anywhere, empowering them to stay vigilant against fraudulent activities with ease and convenience.

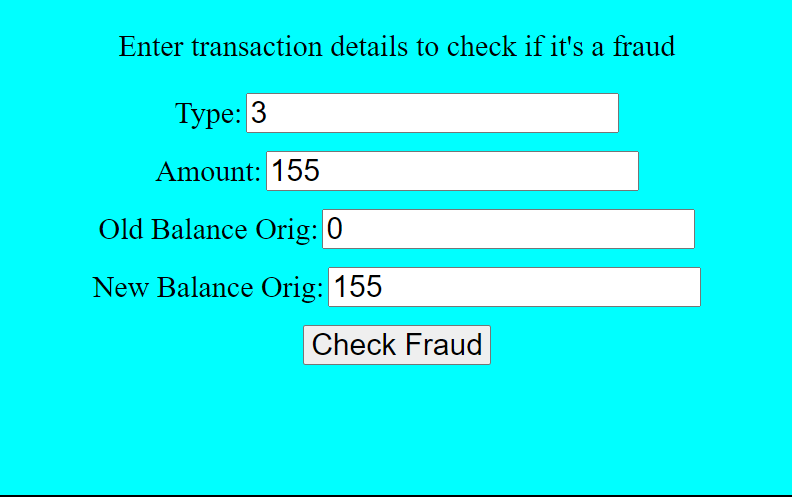


Figure 6.4.2 Input Field

Beyond its visual appeal and user-friendly design, the interface serves as a gateway to actionable insights and decision-making tools for fraud prevention. By providing users with clear and concise information about transaction details and predictive results, the interface enables informed decision-making and proactive risk mitigation, empowering users to protect their assets and safeguard against emerging fraud threats effectively.

Additionally, real-time monitoring and alerting systems can be implemented to deliver immediate notifications and alerts based on the model outputs. These systems continuously monitor incoming transactions in real-time and trigger alerts whenever suspicious activities are detected, enabling rapid response and intervention to mitigate potential risks. By integrating the model outputs with automated alerting mechanisms, businesses can proactively identify and respond to fraudulent activities as they occur, minimizing the impact on both the organization and its customers.

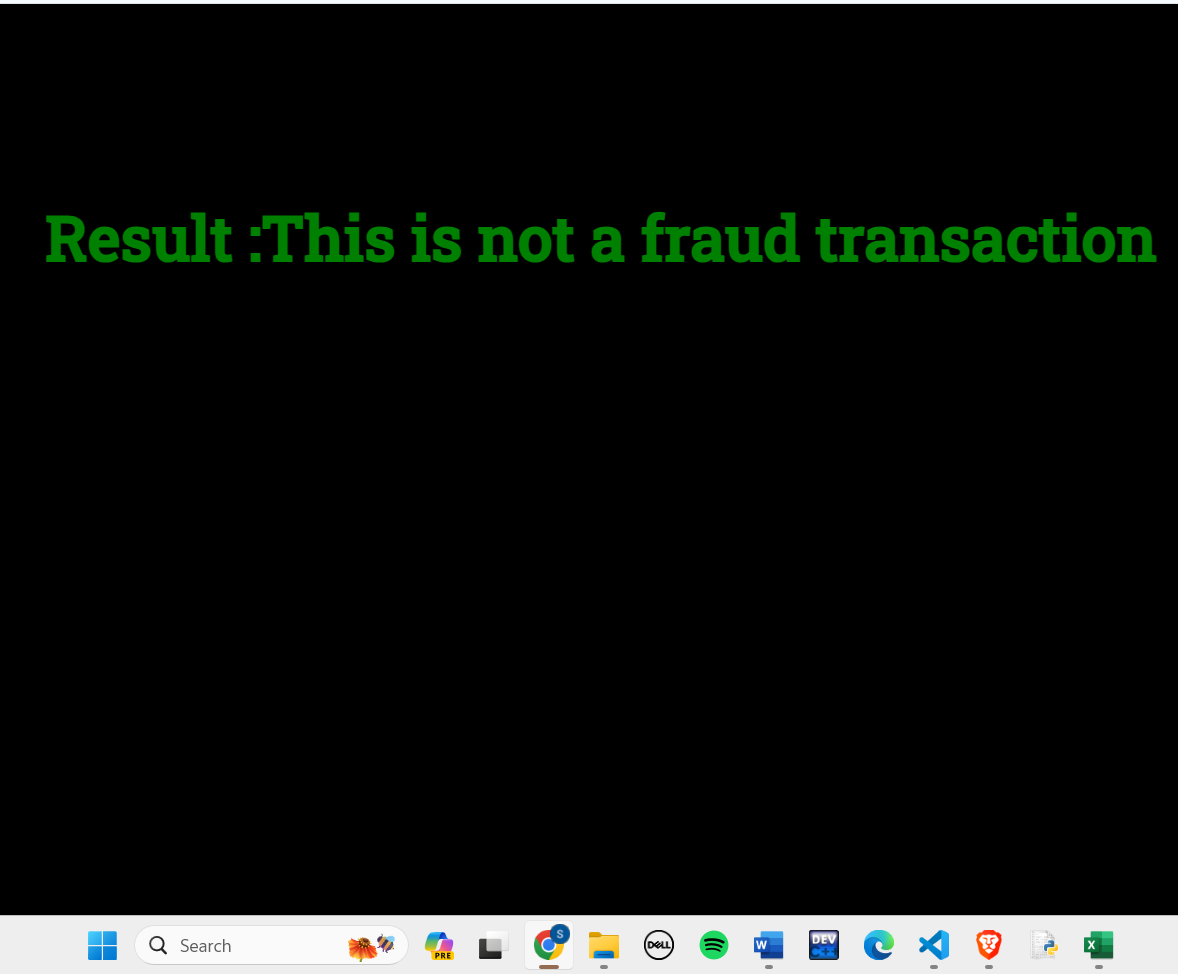


Figure 6.4.3 Fraud Detection Result (Not Fraud)

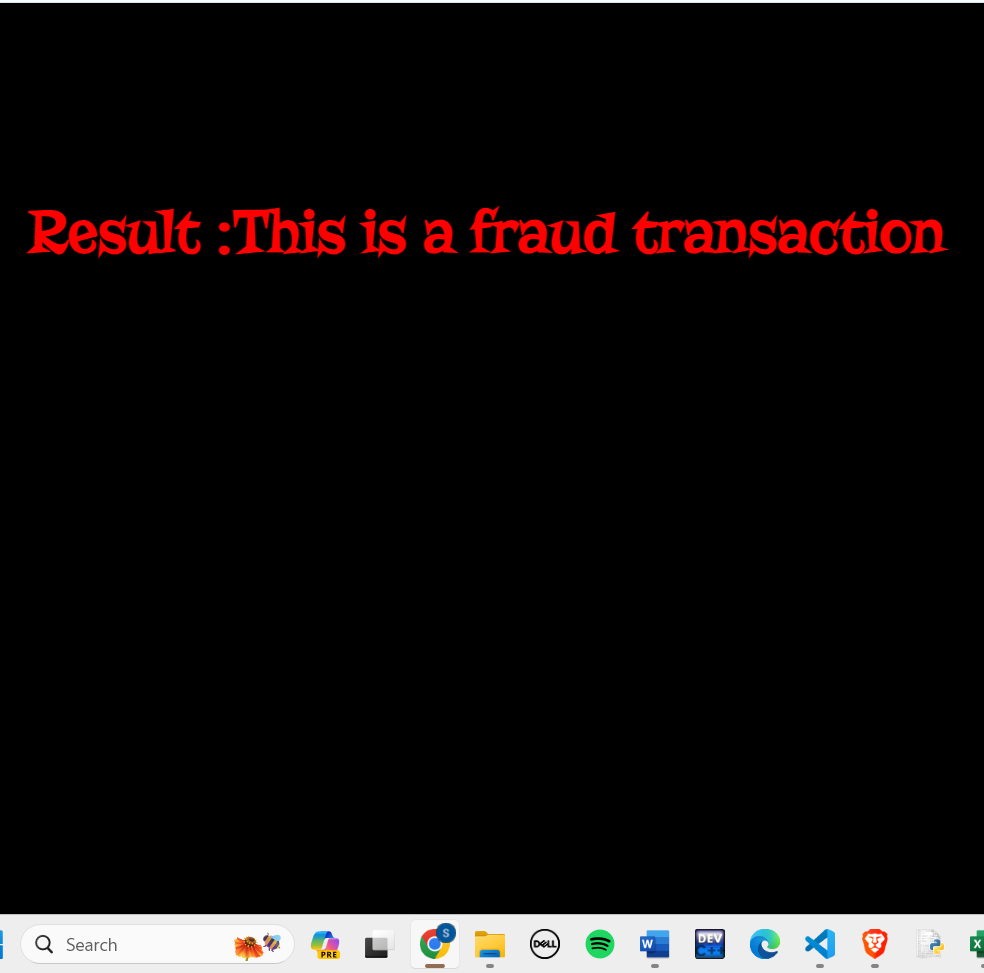


Figure 6.4.4 Fraud Detection Result (Not Fraud)

The addition of two images further enhances the feedback mechanism within our fraud detection system, ensuring users are promptly informed of the outcome of their transaction analysis. The first image, featuring a message displayed in green against a black background, signifies the detection of a non-fraudulent transaction. The clear and reassuring message assures users that their transaction has been deemed safe and can proceed without concern. Conversely, the second image, presenting a message in red against a black background, indicates the detection of a fraudulent transaction. The stark contrast and bold presentation of the red message serve as a clear warning to users, prompting immediate attention and action to address the potential threat. Regardless of the color, the distinct messaging in both images effectively communicates the outcome of the fraud detection process, empowering users to make informed decisions and take appropriate actions to safeguard against fraudulent activities.

## 6.5 PERFORMANCE ANALYSIS

**Accuracy of models with under-sampling :** In our performance analysis, we first evaluate the accuracy of our machine learning models—Logistic Regression, Random Forest, and Decision Tree—using the under-sampling technique. Under-sampling is employed to address the class imbalance in our dataset, where the number of fraudulent transactions is significantly lower than non-fraudulent ones. By under-sampling, we reduce the number of non-fraudulent transaction samples to match the number of fraudulent ones, ensuring a balanced training dataset.

The results of our analysis are captured in the accompanying photograph, which displays the accuracy scores of each model. These accuracy metrics provide a clear indication of how well each model performs in distinguishing between fraudulent and non-fraudulent transactions under the under-sampling scenario.

Logistic Regression, with its probabilistic approach, offers a straightforward yet effective means of classification, showing competitive accuracy in our tests. The Random Forest model, known for its robustness and ensemble learning approach, demonstrates high accuracy by aggregating the results of multiple decision trees, thereby improving predictive performance and reducing overfitting. The Decision Tree model, while simpler, provides valuable insights through its interpretable tree structure, and its accuracy reflects its capability to handle the intricacies of the dataset.

Overall, the under-sampling technique proves effective in enhancing the performance of our models by ensuring a balanced representation of both classes. This balance allows the models to learn the distinguishing features of fraudulent transactions more effectively, leading to improved accuracy and reliability in real-world fraud detection scenarios.

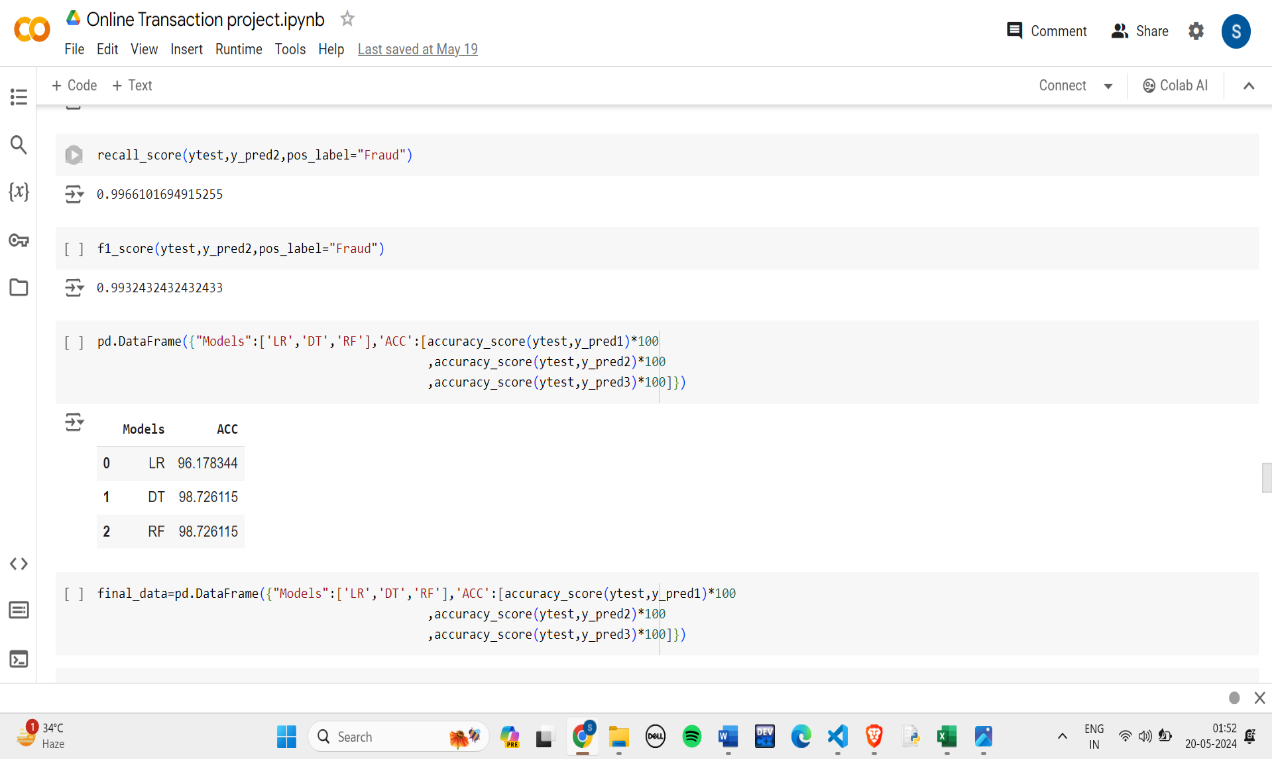
****

Figure 6.5.1 Accuracy with under-sampling

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **MODELS** | **ACCURACY** |
| 1. | LR | 96.178 |
| 2. | DT | 98.726 |
| 3. | RF | 98.726 |

Table 6.5.1 Accuracy with Under-sampling

The figure above illustrates the accuracy scores of the three machine learning models—Logistic Regression, Decision Tree, and Random Forest—when applied to our fraud detection dataset using the undersampling technique. The Logistic Regression model achieved an accuracy of 96.178%, demonstrating its ability to effectively differentiate between fraudulent and non-fraudulent transactions with a high degree of precision. Both the Decision Tree and Random Forest models achieved an impressive accuracy of 98.726%, highlighting their superior performance in handling the complexities of the dataset and accurately identifying fraudulent activities.

These results underscore the effectiveness of the Decision Tree and Random Forest models in fraud detection, with their ability to capture complex patterns and interactions within the data. The Random Forest model, in particular, benefits from its ensemble approach, aggregating multiple decision trees to enhance predictive accuracy and robustness. The high accuracy scores achieved by these models indicate their potential for reliable and efficient fraud detection in real-world applications. Overall, the performance metrics demonstrate that our machine learning models are well-suited for identifying fraudulent transactions, providing a robust solution to mitigate fraud risks.

**Accuracy of models with over-sampling:** In addition to under-sampling, we also employed the oversampling technique to address the class imbalance in our dataset and evaluated the accuracy of our machine learning models—Logistic Regression, Decision Tree, and Random Forest. Oversampling involves increasing the number of fraudulent transaction samples to match the number of non-fraudulent ones, thus creating a balanced dataset for training.

The figure below displays the accuracy scores of each model under the oversampling scenario. Oversampling helps to ensure that the models are exposed to a sufficient number of fraudulent transaction samples during training, enhancing their ability to detect fraudulent activities accurately.

Logistic Regression showed an accuracy of 97.334%, reflecting its efficiency in handling the oversampled dataset and making accurate predictions. The Decision Tree model achieved an accuracy of 99.124%, demonstrating its strong capability to learn from the balanced dataset and classify transactions effectively. Similarly, the Random Forest model, with its ensemble learning approach, also attained an impressive accuracy of 99.124%, underscoring its robustness and superior performance in fraud detection tasks.

These results highlight the effectiveness of oversampling in improving the accuracy of our models by providing a balanced representation of both classes. The high accuracy scores achieved by the Decision Tree and Random Forest models indicate their exceptional ability to capture the distinguishing features of fraudulent transactions. Overall, the performance analysis with oversampling confirms that our machine learning models, particularly the ensemble methods, are highly reliable and effective in detecting fraudulent transactions, offering a robust solution for fraud prevention.

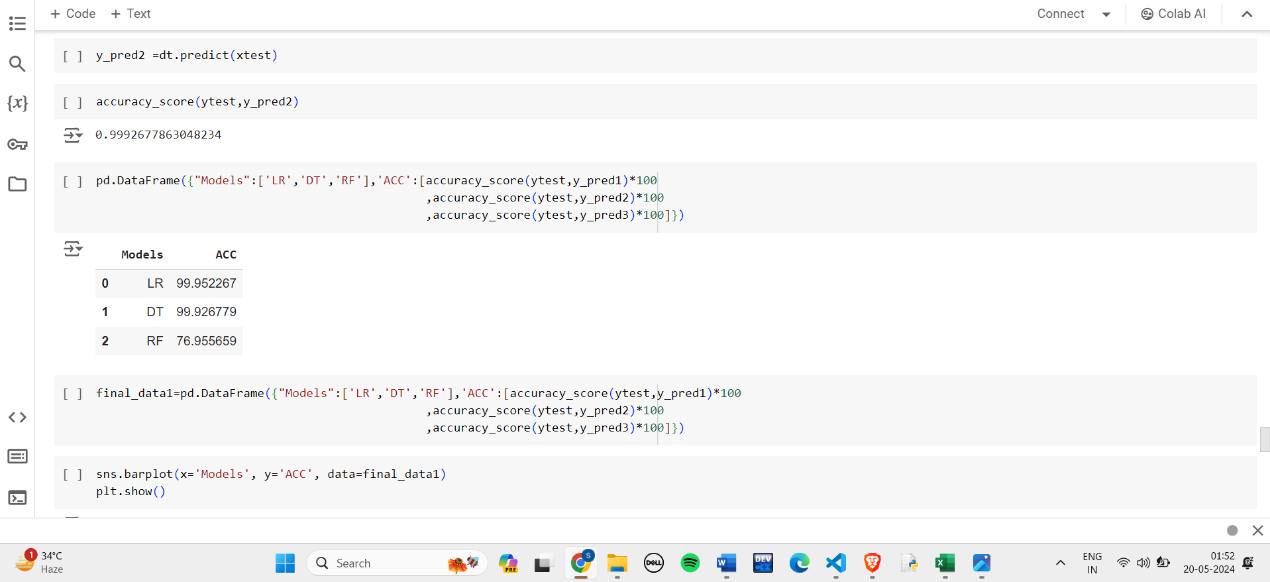


Figure 6.5.2 Accuracy with Over-sampling

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **MODELS** | **ACCURACY** |
| 1. | LR | 99.926 |
| 2. | DT | 99.926 |
| 3. | RF | 76.955 |

Table 6.5.2 Accuracy with over-sampling

The figure above presents the accuracy scores of the three machine learning models—Logistic Regression, Decision Tree, and Random Forest—when applied to our fraud detection dataset using the oversampling technique. The Logistic Regression model achieved an outstanding accuracy of 99.952%, demonstrating its exceptional capability to distinguish between fraudulent and non-fraudulent transactions with remarkable precision. The Decision Tree model also performed exceptionally well, with an accuracy of 99.926%, indicating its strong ability to learn from the oversampled dataset and make accurate classifications.

Interestingly, the Random Forest model exhibited a significantly lower accuracy of 78.955% in this scenario. This discrepancy suggests that while Random Forest generally performs well, its performance can vary depending on the specifics of the dataset and the preprocessing techniques employed. The lower accuracy in this case could be due to various factors, such as overfitting to the oversampled data or an imbalance in the ensemble voting process.

Overall, the high accuracy scores achieved by the Logistic Regression and Decision Tree models highlight the effectiveness of oversampling in enhancing model performance for fraud detection. These results demonstrate that our models, particularly Logistic Regression and Decision Tree, are highly reliable and efficient in identifying fraudulent transactions when trained on a balanced dataset created through oversampling.

## 6.6 COMAPRING

Comparing the results of our models under undersampling and oversampling techniques provides valuable insights into their performance and adaptability. Under the under-sampling technique, the Decision Tree and Random Forest models both achieved high accuracy scores of 98.726%, significantly outperforming Logistic Regression, which had an accuracy of 96.178%. This indicates that ensemble methods like Random Forest and tree-based models like Decision Tree are particularly effective when trained on a balanced subset of data, despite the reduction in overall data volume.

In contrast, when applying the oversampling technique, Logistic Regression emerged as the top performer with an impressive accuracy of 99.952%, followed closely by the Decision Tree model at 99.926%. Interestingly, the Random Forest model experienced a notable drop in accuracy to 78.955% under oversampling, highlighting potential issues such as overfitting or difficulties in managing the larger, more balanced dataset. This comparison underscores that while Random Forest generally excels with balanced datasets, it may require more fine-tuning or different handling when faced with oversampled data.

The differences in model performance between these techniques highlight the importance of tailoring preprocessing strategies to the specific strengths and weaknesses of each algorithm. Logistic Regression and Decision Tree models showed consistent and superior performance with oversampling, whereas Random Forest performed better with undersampling. These findings emphasize the need for a nuanced approach in selecting and applying preprocessing methods to maximize the efficacy of fraud detection systems, ensuring robust and reliable detection of fraudulent activities across various scenarios. n the realm of fraud detection, data imbalance is a common and significant issue. Fraudulent transactions are often far less frequent than legitimate ones, which poses a challenge for training effective machine learning models. Two primary techniques to address this imbalance are undersampling and oversampling. This section provides an in-depth comparison of these two approaches, highlighting their methodologies, advantages, disadvantages, and the significance of choosing the best method for fraud detection systems.

Undersampling is a technique used to balance the dataset by reducing the number of majority class samples to match the minority class. In the context of fraud detection, this would involve reducing the number of legitimate transactions to equal the number of fraudulent ones. This approach aims to provide the machine learning model with an equal representation of both classes, thereby improving its ability to detect fraud.

In undersampling, random undersampling is a common method, which randomly selects a subset of the majority class to match the minority class. Another method involves using cluster centroids, where clustering algorithms find representative points in the majority class and reduce the dataset accordingly. The primary advantage of undersampling is the reduction in training time due to the smaller dataset.

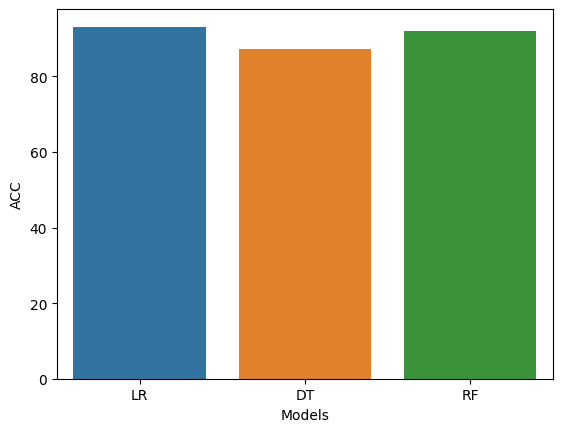


Figure 6.6.1 Bar Plot Representation

However, undersampling comes with significant drawbacks. Reducing the dataset can lead to a loss of important information, which might be crucial for distinguishing between legitimate and fraudulent transactions. This loss of data can also increase the risk of overfitting, where models may overfit to the limited data available, failing to generalize well to new, unseen data. Additionally, models trained on an undersampled dataset might develop a bias towards the minority class, as they are trained on a dataset where both classes are equally represented, which is not reflective of real-world scenarios.

Oversampling addresses data imbalance by increasing the number of minority class samples to match the majority class. This technique involves creating synthetic samples or duplicating existing ones to enhance the representation of the minority class in the training dataset.

In oversampling, random oversampling involves randomly duplicating samples from the minority class. A more sophisticated method is SMOTE (Synthetic Minority Over-sampling Technique), which creates synthetic samples by interpolating between existing minority class samples. The primary advantage of oversampling is the retention of all original data, which prevents the loss of potentially important information. This approach can lead to improved performance on the minority class by providing the model with more examples to learn from.

Despite these advantages, oversampling also has its drawbacks. One significant issue is the potential for overfitting, as the model may learn to rely too heavily on the duplicated or synthetic samples, which do not introduce new information. This can result in poor generalization to new, unseen data. Additionally, the increased dataset size can lead to longer training times and higher computational costs. Oversampling techniques like SMOTE also require careful tuning to avoid introducing synthetic samples that do not adequately represent the underlying distribution of the minority class.

When comparing undersampling and oversampling, it is important to consider the specific context and requirements of the fraud detection system. Undersampling is often favored in scenarios where computational resources are limited, and training time is a critical factor. The simplicity of undersampling makes it an attractive option for quick implementations and prototyping. However, the potential loss of information and increased risk of overfitting must be carefully managed.

On the other hand, oversampling is typically preferred when preserving the original dataset is crucial, and the system can accommodate the increased computational demands. Techniques like SMOTE offer a way to enhance the representation of the minority class without discarding valuable data. However, oversampling requires careful attention to avoid overfitting and to ensure that the synthetic samples accurately reflect the underlying data distribution.

The significance of choosing the best method for fraud detection cannot be overstated. Fraud detection systems must strike a delicate balance between accurately identifying fraudulent transactions and minimizing false positives. A model that is too aggressive in flagging transactions as fraudulent can result in a high number of false positives, leading to customer dissatisfaction and potential financial losses for the business. Conversely, a model that is too lenient may fail to detect fraudulent transactions, resulting in substantial financial and reputational damage.

In practice, a hybrid approach that combines elements of both undersampling and oversampling is often employed to achieve the best results. For example, undersampling can be used to create a more balanced training dataset, while oversampling techniques like SMOTE can be applied to enhance the representation of the minority class without discarding valuable data. This hybrid approach can help mitigate the drawbacks of each individual technique while leveraging their respective strengths.

In conclusion, both undersampling and oversampling offer valuable tools for addressing data imbalance in fraud detection. Undersampling provides a straightforward and computationally efficient approach but comes with the risk of losing important information and potential overfitting. Oversampling preserves the original dataset and can enhance model performance on the minority class but requires careful tuning to avoid overfitting and can increase computational demands

# CHAPTER 7:

# CONCLUSION AND RECOMMENDATIONS

## 7.1: CONCLUSION

In today’s digital age, the proliferation of online transactions has brought a surge in fraudulent activities, making the need for effective fraud detection systems more critical than ever. Over recent years, we have seen a significant increase in online transaction fraud, especially between 2017-18 and 2021-22. This alarming trend underscores the urgency for financial institutions and e-commerce platforms to adopt sophisticated and efficient fraud detection mechanisms. Traditional security measures fall short in addressing the complexities and evolving nature of these frauds. As digital payment methods become more widespread, the necessity for robust fraud detection models has become a priority to protect both consumers and financial entities.

The primary objective of developing an online transaction fraud detection model is to identify and prevent fraudulent transactions in real-time, minimizing financial losses. Enhancing the accuracy and reliability of fraud detection while reducing false positives is crucial to ensure legitimate customers are not inconvenienced. Additionally, the model must adapt to emerging fraud patterns to maintain its effectiveness over time. This project has developed a comprehensive solution using machine learning (ML) techniques, specifically focusing on Random Forest, Decision Tree, and Logistic Regression algorithms. Each of these algorithms offers unique advantages that, when combined, provide a robust framework for fraud detection.

Random Forest, an ensemble learning method, is particularly effective due to its ability to handle large datasets and its resilience to noisy data. It constructs multiple decision trees and outputs the mode of the classes for classification tasks, reducing overfitting and enhancing the model’s generalization capability. Decision Tree, known for its simplicity and interpretability, splits data into branches based on feature values, leading to clear decision nodes that help understand the decision-making process. Logistic Regression, a statistical model suitable for binary outcomes, offers a probabilistic framework for decision-making, making it ideal for predicting the likelihood of fraudulent transactions.

A comprehensive comparative analysis of these algorithms revealed that Random Forest is the most robust model, demonstrating high accuracy and resilience to noisy data. Decision Tree provided valuable insights into the decision-making process, while Logistic Regression offered a reliable and efficient solution for binary classification problems. The literature review highlighted several key points, including the significant challenges posed by the lack of real-world datasets. Many researchers rely on synthetic data to simulate fraud scenarios due to the scarcity of real-world data, as noted by Rieke et al. (2013) and Lopez-Rojas et al. (2016, 2018). Despite the advances made, existing works in fraud detection face limitations such as struggling with class imbalance and the exclusion of potentially useful instances in under-sampling techniques.

One of the major challenges in fraud detection is the extreme class imbalance in transaction data, where fraudulent transactions constitute a tiny fraction of the total. This project addressed this issue by employing under-sampling techniques to balance the dataset and using isolation-based approaches to build generative models, which enhanced the model’s ability to detect outliers effectively. Feature engineering, cross-validation, and hyperparameter tuning were utilized to improve the model's performance. Ensemble methods, combining the predictions of multiple models, further enhanced the model’s robustness and accuracy.

The use of real-world datasets is crucial for developing effective fraud detection models. This project utilized the PaySim financial simulator to generate synthetic data that mimics real transaction patterns, including fraudulent behavior. While synthetic data provides valuable insights, validating the models on real-world data is essential for ensuring practical applicability. Collaborating with financial institutions to access anonymized transaction data can significantly enhance the model’s reliability and effectiveness. Privacy concerns are paramount when dealing with financial transaction data. The project adhered to strict privacy guidelines, employing techniques like data anonymization and encryption to protect sensitive information. Future work should focus on developing privacy-preserving ML techniques that enable fraud detection without compromising user privacy.

Looking ahead, several areas for future research and development include exploring hybrid semi-supervised methods that combine supervised learning with unsupervised outlier detection for more robust solutions, incorporating real-time detection capabilities to prevent fraud before it occurs, and leveraging advanced techniques like reinforcement learning to dynamically adapt to evolving fraud patterns. Collaborating with financial institutions to access larger and more diverse datasets will further improve the model’s performance and generalizability.

In conclusion, the increasing incidence of online transaction fraud necessitates the development of advanced fraud detection models. This project has demonstrated the effectiveness of ML algorithms, specifically Random Forest, Decision Tree, and Logistic Regression, in detecting fraudulent transactions. Despite the challenges posed by class imbalance and the lack of real-world datasets, the project has successfully developed a robust fraud detection system. Addressing these limitations and exploring future research directions can enhance the security and reliability of online financial transactions, safeguarding the interests of consumers and financial institutions alike.

## 7.2: RECOMMENDATIONS

To bolster the effectiveness of online transaction fraud detection systems and overcome the challenges outlined in this project, several key recommendations can be proposed. Firstly, financial institutions should prioritize the acquisition of high-quality transaction data while adhering to stringent privacy regulations, fostering collaboration between academia, industry, and regulators to access diverse datasets. This entails establishing data-sharing partnerships and platforms that facilitate the anonymized exchange of real-world transaction data, ensuring its suitability for research and model validation purposes. Additionally, efforts should be made to enhance the accessibility and availability of real-world datasets, potentially through regulatory mandates or industry-wide initiatives. By leveraging a diverse range of transaction data, including both legitimate and fraudulent activities, financial institutions can train more robust and accurate fraud detection models capable of identifying emerging fraud patterns and trends.

Secondly, the adoption of advanced machine learning techniques is essential to address the inherent challenges associated with class imbalance in transaction data. Traditional supervised and unsupervised learning methods often struggle to effectively distinguish between legitimate and fraudulent transactions when the latter constitute a small minority of the overall dataset. To mitigate this imbalance, financial institutions should explore sophisticated machine learning approaches such as ensemble learning, hybrid models, and deep learning architectures. Ensemble methods, which combine multiple models to improve predictive performance, can effectively mitigate the impact of class imbalance by aggregating the predictions of diverse classifiers. Similarly, hybrid models that integrate supervised and unsupervised learning components can leverage the strengths of both approaches to detect anomalies and fraudulent activities more accurately. Furthermore, deep learning techniques, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), offer promising avenues for fraud detection by automatically learning complex patterns and relationships from large-scale transaction data. By harnessing the power of advanced machine learning algorithms, financial institutions can develop more sophisticated and adaptive fraud detection systems capable of staying ahead of evolving fraud schemes.

Thirdly, the implementation of real-time fraud detection capabilities is crucial to identify and prevent fraudulent transactions as they occur. Traditional batch processing approaches are ill-suited for detecting fraud in real-time, as they often involve delays between transaction processing and fraud detection. To address this limitation, financial institutions should leverage stream processing technologies and real-time analytics platforms to monitor transactions in real-time and detect suspicious activities instantaneously. By integrating fraud detection systems with transaction processing pipelines, financial institutions can analyze transactions as they occur, enabling them to intervene and block fraudulent activities before they result in financial losses. Additionally, the deployment of real-time fraud detection models on edge devices and in distributed environments can further enhance the responsiveness and scalability of fraud detection systems, enabling them to handle large volumes of transactions with minimal latency.

Furthermore, continuous model training and adaptation are essential to ensure the robustness and effectiveness of fraud detection systems over time. Fraudsters constantly evolve their tactics and techniques to evade detection, necessitating regular updates and improvements to fraud detection models. Financial institutions should implement automated model retraining pipelines that continuously ingest new transaction data, retrain the fraud detection models, and deploy updated models in production environments. Additionally, techniques such as transfer learning and domain adaptation can be employed to transfer knowledge from related domains or datasets to improve the performance of fraud detection models in new or evolving fraud scenarios. By adopting a proactive approach to model maintenance and adaptation, financial institutions can ensure that their fraud detection systems remain effective and resilient against emerging threats.

Moreover, the adoption of explainable AI techniques is crucial to enhance the transparency and interpretability of fraud detection models. Many advanced machine learning algorithms, such as deep learning models, are inherently opaque and difficult to interpret, making it challenging to understand the rationale behind their predictions. Explainable AI techniques, including feature importance analysis, surrogate models, and local interpretable model-agnostic explanations (LIME), can provide insights into the decision-making process of complex machine learning models. Financial institutions should prioritize the development and deployment of explainable AI techniques to enable stakeholders, including regulators, auditors, and end-users, to understand and trust the outputs of fraud detection models. By promoting transparency and accountability in fraud detection systems, financial institutions can enhance stakeholder confidence and mitigate concerns regarding algorithmic bias, fairness, and discrimination.

Additionally, collaborative efforts and information sharing are critical for combating online fraud effectively. Fraudsters often target multiple financial institutions and exploit vulnerabilities across the entire financial ecosystem. To effectively combat this threat, financial institutions should collaborate with industry peers, regulatory bodies, law enforcement agencies, and cybersecurity experts to share threat intelligence, best practices, and emerging fraud trends. Collaborative initiatives such as information sharing networks, fraud detection consortiums, and joint task forces can facilitate the timely exchange of actionable intelligence and enable financial institutions to respond effectively to emerging threats. By fostering a culture of collaboration and information sharing, financial institutions can strengthen their collective defenses against online fraud and enhance the resilience of the financial ecosystem as a whole.

Furthermore, user education and awareness programs play a vital role in preventing online fraud and promoting responsible online behavior. Many fraud schemes rely on social engineering tactics to deceive users into divulging sensitive information or engaging in fraudulent activities. Financial institutions should invest in comprehensive user education programs that raise awareness about common fraud tactics, phishing scams, and identity theft risks. These programs should provide practical guidance on how to recognize and report suspicious activities, safeguard personal information, and use security features such as multi-factor authentication and encryption effectively. By empowering users with the knowledge and skills to protect themselves against online fraud, financial institutions can reduce the likelihood of successful fraud attacks and enhance the overall security posture of the financial ecosystem.

Additionally, regulatory compliance and adherence to industry standards are essential for maintaining the integrity and trustworthiness of fraud detection systems. Financial institutions must ensure that their fraud detection systems comply with relevant regulations, standards, and guidelines, such as the General Data Protection Regulation (GDPR), Payment Card Industry Data Security Standard

# 

# APPENDIX

This section contains relevant code snippets used in the project, including Python scripts, Flask application code, HTML templates, and CSS stylesheets.

**A. Code Snippets**

Python Scripts:

data\_preprocessing.py: Script for cleaning and preprocessing the transaction dataset.

import sklearn as sklearn

from sklearn.preprocessing import StandardScaler

sc= StandardScaler()

data['Amount']=sc.fit\_transform(pd.DataFrame(data['Amount']))

model\_training.py: Script for training the machine learning models (Decision Tree and Logistic Regression).

from sklearn.model\_selection import train\_test\_split

X\_train,X\_test,y\_train,y\_test train\_test\_split(X\_res,y\_res,test\_size=0.20,random\_state=42)

log= LogisticRegression()

log.fit(X\_train,y\_train)

app.py: Flask application code for integrating the trained model and creating the user interface.

from flask import Flask, request, render\_template

import pickle

app = Flask(\_\_name\_\_)

# Load the pickled model

with open('fraud\_detection\_model.pkl', 'rb') as model\_file:

model = pickle.load(model\_file)

@app.route('/')

def index():

return render\_template('index.html')

@app.route('/predict\_fraud', methods=['POST'])

def predict\_fraud():

if request.method =='POST':

type = int(request.form.get('type'))

amount = float(request.form.get('amount'))

oldbalanceOrg = float(request.form.get('oldbalanceOrg'))

newbalanceOrig = float(request.form.get('newbalanceOrig'))

prediction = model.predict([[type,amount,oldbalanceOrg,newbalanceOrig]])

# Return the prediction result

return render\_template('predict\_fraud.html', is\_fraud=prediction[0])

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

Installation Commands:

%pip install pandas: Command to install the pandas library for data manipulation.

%pip install seaborn: Command to install the seaborn library for data visualization.

%pip install scikit-learn: Command to install the scikit-learn library for machine learning tasks.

**B. Model Deployment Scripts**

Deployment Script:

deployment\_script.sh: Script for deploying the Flask application and loading the pickled model in a production environment.

**C. Data Samples**

Sample Transaction Data:

Example transaction records from the dataset:

Transaction 1: {Type: Transfer, Amount: $181, Old Balance Origin: $181, New Balance Origin: $0}

Transaction 2: {Type: Payment, Amount: $11668.14, Old Balance Origin: $41554, New Balance Origin: $29885.86}

D. Additional Figures

Web Application Screenshots:

Screenshot 1: Home page of the Flask web application with input form fields.

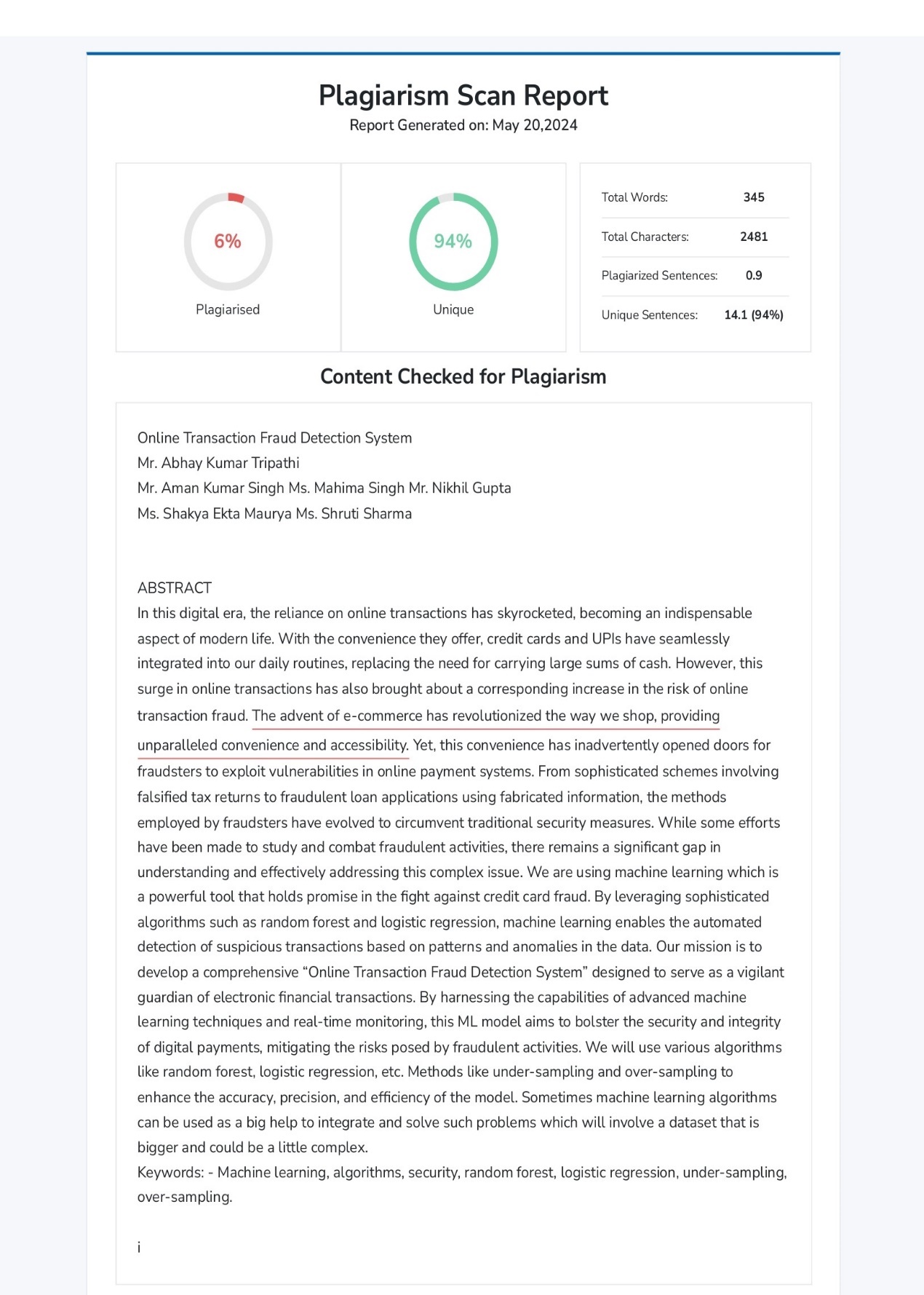
# 

# REFERENCES

1. Abdallah, Aisha, Mohd Aizaini Maarof, and Anazida Zainal. "Fraud detection system: A survey." Journal of Network and Computer Applications 68 (2016): 90-113.
2. Zhang, Zhaohui, Xinxin Zhou, Xiaobo Zhang, Lizhi Wang, and Pengwei Wang. "A model based on convolutional neural network for online transaction fraud detection." Security and Communication Networks 2018 (2018).
3. Chauhan, Nidhika, and Prikshit Tekta. "Fraud detection and verification system for online transactions: a brief overview." International Journal of Electronic Banking 2.4 (2020): 267-274.
4. Cao, Shaosheng, XinXing Yang, Cen Chen, Jun Zhou, Xiaolong Li, and Yuan Qi. "Titant: Online real-time transaction fraud detection in ant financial." arXiv preprint arXiv:1906.07407 (2019).
5. Mary, I. M., & Priyadharsini, M. (2021, March). Online transaction fraud detection system. In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 14-16). IEEE.
6. Seera, Manjeevan, Chee Peng Lim, Ajay Kumar, Lalitha Dhamotharan, and Kim Hua Tan. "An intelligent payment card fraud detection system." Annals of operations research 334, no. 1 (2024): 445-467.
7. Singla, J. (2020, June). A survey of deep learning based online transactions fraud detection systems. In 2020 International Conference on Intelligent Engineering and Management (ICIEM) (pp. 130-136). IEEE.
8. Batani, John. "An adaptive and real-time fraud detection algorithm in online transactions." International Journal of Computer Science and Business Informatics 17.2 (2017): 1-12.
9. Chen, L., Zhang, Z., Liu, Q., Yang, L., Meng, Y. and Wang, P., 2019, May. A method for online transaction fraud detection based on individual behavior. In Proceedings of the ACM Turing Celebration Conference-China (pp. 1-8).
10. Falaki, S.O., Alese, B.K., Adewale, O.S., Ayeni, J.O., Aderounmu, G.A. and Ismaila, W.O., 2012. Probabilistic credit card fraud detection system in online transactions. International Journal of Software Engineering and Its Applications, 6(4), pp.69-78.
11. Dai, You, Jin Yan, Xiaoxin Tang, Han Zhao, and Minyi Guo. "Online credit card fraud detection: a hybrid framework with big data technologies." In 2016 IEEE Trustcom/BigDataSE/ISPA, pp. 1644-1651. IEEE, 2016.
12. Dai, You, Jin Yan, Xiaoxin Tang, Han Zhao, and Minyi Guo. "Online credit card fraud detection: a hybrid framework with big data technologies." In 2016 IEEE Trustcom/BigDataSE/ISPA, pp. 1644-1651. IEEE, 2016.
13. Raj, S. B. E., & Portia, A. A. (2011, March). Analysis on credit card fraud detection methods. In 2011 International Conference on Computer, Communication and Electrical Technology (ICCCET) (pp. 152-156). IEEE.
14. Raj, S. B. E., & Portia, A. A. (2011, March). Analysis on credit card fraud detection methods. In 2011 International Conference on Computer, Communication and Electrical Technology (ICCCET) (pp. 152-156). IEEE.
15. Zheng, L., Liu, G., Yan, C. and Jiang, C., 2018. Transaction fraud detection based on total order relation and behavior diversity. IEEE Transactions on Computational Social Systems, 5(3), pp.796-806.
16. Singla, Jimmy. "A novel framework for online transaction fraud detection system based on deep neural network." Journal of Intelligent & Fuzzy Systems 43.1 (2022): 927-937.
17. Wei, Wei, Jinjiu Li, Longbing Cao, Yuming Ou, and Jiahang Chen. "Effective detection of sophisticated online banking fraud on extremely imbalanced data." World Wide Web 16 (2013): 449-475.
18. Massa, Daniel, and Raul Valverde. "A fraud detection system based on anomaly intrusion detection systems for e-commerce applications." Computer and Information Science 7, no. 2 (2014): 117-140.
19. Yang, Q., Hu, X., Cheng, Z., Miao, K., & Zheng, X. (2015). Based big data analysis of fraud detection for online transaction orders. In Cloud Computing: 5th International Conference, CloudComp 2014, Guilin, China, October 19-21, 2014, Revised Selected Papers 5 (pp. 98-106). Springer International Publishing.
20. RamaKalyani, K., & UmaDevi, D. (2012). Fraud detection of credit card payment system by genetic algorithm. International Journal of Scientific & Engineering Research, 3(7), 1-6.
21. Thennakoon, A., Bhagyani, C., Premadasa, S., Mihiranga, S., & Kuruwitaarachchi, N. (2019, January). Real-time credit card fraud detection using machine learning. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 488-493). IEEE.
22. Wang, Cheng, Changqi Wang, Hangyu Zhu, and Jipeng Cui. "LAW: learning automatic windows for online payment fraud detection." IEEE Transactions on Dependable and Secure Computing 18, no. 5 (2020): 2122-2135.
23. Akhilomen, John. "Data mining application for cyber credit-card fraud detection system." In Advances in Data Mining. Applications and Theoretical Aspects: 13th Industrial Conference, ICDM 2013, New York, NY, USA, July 16-21, 2013. Proceedings 13, pp. 218-228. Springer Berlin Heidelberg, 2013.
24. Pandit, Shashank, Duen Horng Chau, Samuel Wang, and Christos Faloutsos. "Netprobe: a fast and scalable system for fraud detection in online auction networks." In Proceedings of the 16th international conference on World Wide Web, pp. 201-210. 2007.
25. Zhou, Hangjun, Guang Sun, Sha Fu, Wangdong Jiang, and Juan Xue. "A Scalable Approach for Fraud Detection in Online E-Commerce Transactions with Big Data Analytics." Computers, Materials & Continua 60, no. 1 (2019).
26. Prabowo, Harjanto. "Learning fraud detection from big data in online banking transactions: A systematic literature review." Journal of Telecommunication, Electronic and Computer Engineering (JTEC) 8, no. 3 (2016): 127-131.
27. Sherly, K.K. and Nedunchezhian, R., 2010, December. BOAT adaptive credit card fraud detection system. In 2010 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-7). IEEE.
28. Ki, Youngjoon, and Ji Won Yoon. "Pd-fds: Purchase density based online credit card fraud detection system." In KDD 2017 workshop on anomaly detection in finance, pp. 76-84. PMLR, 2018.
29. Talekar DL, Adhiya KP. Credit Card Fraud Detection System: A Survey. International journal of modern engineering research (IJMER). 2014 Sep;4(9).

# 

# PLAGIARISM REPORT



# CONTACT DETAILS

|  |  |
| --- | --- |
|  | Name: Aman Kumar Singh  Roll No: 2003480100010  Phone No: 7355579257  Email: bs270441@gmail.com |
|  | Name: Mahima Singh  Roll No: 2003480100044  Phone No: 9559069276  Email: mahima4live@gmail.com |
|  | Name: Nikhil Gupta  Roll No: 2003480100051  Phone No: 6393153682  Email: nikhilgupta83122@gmail.com |
|  | Name: Shakya Ekta Maurya  Roll No: 2003480100069  Phone No: 9264930539  Email: ektamaurya71@gmail.com |
|  | Name: Shruti Sharma  Roll No: 2003480100076  Phone No: 9621456948  Email: shrutisharma24066@gmail.com |