Arab American University

Faculty of Graduate Studies

Department of Natural, Engineering and Technology Sciences

Master Program in Data Science and Business Analytics

International Traffic Bypass Detection in Telecom Using Machine Learning and Resampling Techniques on Imbalanced Data

Wael AbuNaeem

201911914

Supervision Committee:

Dr. Majdi Owda

Dr. Mohammed Hussien

Dr. Sami Sader

This Thesis Was Submitted in Partial Fulfilment
of the Requirements for the Master Degree in
Data Science & Business Analytics

Palestine, 1/2026

Arab American University

Faculty of Graduate Studies

Department of Natural, Engineering and Technology Sciences

Master Program in Data Science and Business Analytics

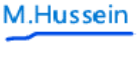
Thesis Approval

International Traffic Bypass Detection in Telecom Using Machine Learning and Resampling Techniques on Imbalanced Data

Wael AbuNaeem
201911914

This thesis was defended successfully on 31/1/2026 and approved by:

Thesis Committee Members:

Name	Title	Signature
1. Dr. Majdi Owda	Supervisor	
3. Dr. Sami Sader	Internal Examiner	
4. Dr. Mohammad Hussein	External Examiner	

Palestine, 1/2026

Declaration

I declare that, except where explicit reference is made to the contribution of others, this thesis is substantially my own work and has not been submitted for any other degree at the Arab American University or any other institution.

Student Name: Wael Rashed Abdelqader Naeem

Student ID: 201911914

Signature: . . . 

Date of Submitting the Final Version of the Thesis: 01 / 03 / 2026

Dedication

To my beloved Wife, Muna, whose unwavering support and endless encouragement have been my greatest source of strength and motivation. To my precious family, especially my father, my mother and my beloved children, for their love, patience, and continuous motivation.

Wael Rashed Abdelqader Naeem

Acknowledgements

First, I would like to thank all who have helped me with this project; My deepest thanks go to Dr. Majdi Owda who was my supervisors during this project, for all his support, and recommendations. I sincerely thank the Arab American University for providing the required facilities, equipment, technical support, and professional staff needed to complete this project successfully. To everyone who contributed to this work in any capacity, thank you.

International Traffic Bypass Detection in Telecom Using Machine Learning and Resampling Techniques on Imbalanced Data

Wael Abu Naeem

Supervision Committee:
Dr. Majdi Owda
Dr. Mohammed Hussien
Dr. Sami Sader

Abstract

Various fraud types are considered major threats to telecom operators due to the huge amount of revenue loss they cause and their effect on the credibility, customer satisfaction, and performance of telecom operators. One major type of fraud that is the subject of our study is International Traffic Bypass fraud, also known as SIMBOX, SIMBOX is a common and widely used type of fraud in the telecom industry that causes losses of billions of dollars yearly Howell, 2021. In this type of fraud, fraudsters usually use a special device to bypass international traffic and gain revenues on the account of telecom operators. The traditional approaches used for SIMBOX fraud detection is by using an expert system with a predefined set of fixed rules and statistical analysis conducted on traffic Event Detailed Record (EDRs) from source systems mainly Mobile Switch Center (MSC) during a given period, based on system rules and traffic analysis results including the normal calling hours, call duration, differently called numbers, different locations, etc., the subscriber will be classified as a suspected fraudster or not. The traditional approach in SIMBOX detection requires a continuous update of the rules and generates false positive cases frequently. This necessitates the need to develop intelligent and efficient models for detection using Machine Learning (ML) algorithms. The intricacies of employing machine learning techniques for SIMBOX detection were investigated in this research, with a primary focus on the application of Support Vector Machine (SVM) and Random Forest Classifier (RFC). A real telecom dataset was used for this purpose composed of of 23,017 cases of which 22,289 normal cases and 728 fraudulent cases. That dataset was generated from big raw data storage of millions of records. A comparison of classification accuracy was conducted both before and after the implementation of various resampling techniques. These strategies were utilized to address the prevalent issue of imbalanced datasets common in this domain. Ultimately, the most effective approach for managing data imbalance was identified through this evaluative process. The results show that oversampling is the best approach to solve the imbalance issue in both RF and SVM implementations, SVM accuracy is slightly better than RF, SVM gave an accuracy of 99.71% while RF gave 99.66%, SVM has a lower FPR and it takes three times less time than RF for model tuning, training, and testing.

Keywords: Telecom Fraud, International Traffic Bypass, SIMBOX Detection, Machine Learning, Support Vector Machine (SVM), Random Forest Classifier (RFC), Data Imbalance, Resampling, Oversampling, Event Detailed Records (EDRs).

Table of Contents

#	Title	Page
	Declaration	I
	Dedication	II
	Acknowledgements	III
	Abstract	IV
	List of Tables	VIII
	List of Figures	IX
	List of Definitions of Abbreviations	XI
	Chapter One: Introduction	1
1.1	Fraud in the Telecom Industry	1
1.2	Thesis Statement	2
1.3	Study Objective	2
1.4	Study Significance	3
1.5	Models Selection	4
1.6	Telecom Fraud Detection in Palestinian Context	4
1.7	Thesis layout	5
	Chapter Two: Literature Review	6
2.1	Background	6
2.2	Telecom Fraud Definitions and Classification	6
2.2.1	Subscription Fraud	7
2.2.2	System Access Fraud	7
2.2.3	Call selling	7
2.2.4	Premium-rate Service Fraud	7
2.2.5	Internal fraud	7
2.2.6	SIM card cloning	8

2.2.7 Interconnect Traffic Bypass, SIMBOX 8
2.3 Fraud Management Systems 9
2.3.1 Fraud Prevention Systems (FPS) 10
2.3.2 Fraud Detection Systems (FDS) 10
2.4 Fraud Detection Using ML 11
2.5 Handling Imbalance in Fraud Datasets through Data Sampling 20
2.6 Summary 23

Chapter Three: Research Methodology 25
3.1 Dataset Description 25
3.2 System Methodology 26
3.3 Extracting Data from Source System 27
3.4 Load The Data Using Python 28
3.5 Data Preparation and Preprocessing 29
3.5.1 Feature Extraction and Handling Missing Data 29
3.5.2 Identifying and Removing Outliers 30
3.5.3 Data Normalization 31
3.5.4 Dummy Variable Coding 33
3.6 Explanatory Data Analysis EDA 33
3.7 Data Sampling 40
3.8 Development of RF Model 42
3.9 Development of SVM Model 43
3.10 Summary 46

Chapter Four: Results and Discussion 48
4.1 RF Implementation Results 48
4.2 SVM Implementation Results 51
4.3 Models Performance Evaluation 52
4.4 RF and SVM FNR, FPR and RMSE Evaluation 54
4.5 Model Building Duration Evaluation 55
4.6 Study limitations 55
4.7 Summary 56

Chapter Five: Conclusion and Recommendations 58
5.1 Summary 58
5.2 Conclusion 58
5.3 Future Work 59

References 60

List of Tables

Table #	Title of Table	Page
Table 2.1	Examples of Fraud Prevention Systems and Their Key Features	10
Table 2.2	Scoring Supervised Learning Models Kilinc, 2022	13
Table 2.3	Comparison of different sampling techniques: Pros and Cons Awe et al., 2024	23
Table 3.1	Feature Descriptions of Fraud Dataset	26
Table 3.2	Categorization of Features	27
Table 3.3	Dataset Features Descriptive Statistics (Fraudulent Cases)	31
Table 3.4	Dataset Features Descriptive Statistics (Non-Fraudulent Cases)	31
Table 3.5	Outliers Analysis Results	32
Table 4.1	RF Performance Measures - Imbalanced Dataset	48
Table 4.2	RF Performance Measures - Effect of Sampling on Model Performance	49
Table 4.3	SVM Performance Measures - Different Kernels	51
Table 4.4	SVM Performance Measures - Different Kernels Sampled vs Non-Sampled	52
Table 4.5	Comparison of selected RF and SVM model	54

List of Figures

Figure #	Title of Figure	Page
Fig 2.1	On-net and Off-net SIMBOX bypass Krasić and Čelar, 2022	9
Fig 2.2	Proposed Machine Learning Model Kilinc, 2022	13
Fig 2.3	Comparison of SVM and ANN model training duration Sallehuddin et al., 2015	18
Fig 3.1	System Methodology	28
Fig 3.2	Distribution by Outgoing Minutes Before and After Remove Outliers	32
Fig 3.3	Dataset's Shape	33
Fig 3.4	Dataset's Messing Values Check	34
Fig 3.5	Feature Distribution By Status	35
Fig 3.6	By Status Distribution Across Different Categories	36
Fig 3.7	Features Heatmap	38
Fig 3.8	Call Count and Call Duration Distribution Across Different Categories	38
Fig 3.9	Numerical Features with High Correlation	40
Fig 3.10	Sampling Techniques Effect Comparison	41
Fig 3.11	Flow of RF implementation	44

Fig 3.12	Flow of SVM implementation	45
Fig 4.1	RF ROC - Effect of Sampling on Model Performance	49
Fig 4.2	RF Confusion Matrix - Sampled vs Non-Sampled	50
Fig 4.3	RF Features Importance	50
Fig 4.4	SVM Confusion Matrix - Different Kernels Sampled vs Non-Sampled	53
Fig 4.5	SVM ROC - Effect of Sampling on Model Performance	53
Fig 4.6	RF vs SVM Confusion Matrix	54

List of Definitions of Abbreviations

Abbreviations	Title
EDRs	Event Detailed Record
MSC	Mobile Switch Center
ANNs	Artificial Neural Networks
SVM	Support Vector Machine
RFC	Random Forest Classifier
ML	Machine Learning
DL	Deep Learning
FMS	Fraud Management Systems
FPS	Fraud Prevention Systems
FDS	Fraud Detection Systems
CNN	Convolutional Neural Network
KNN	K-Nearest Neighbors
CEP	Complex Event Processing
FNR	False Negative Rate
FPR	False Positive Rate
RMSE	Root Mean Square Error
PBX	Private Branch Exchange
SIM	Subscriber Identity Module
VOIP	Voice Over Internet Protocol
MLTs	Machine Learning Techniques
CDRs	Call Detail Records
NN	Neural Network
ANN	Artificial Neural Network
MCC	Matthews Correlation Coefficient
NB	Naive Bayes
EDRs	Event Detailed Record
MSC	Mobile Switch Center
ANNs	Artificial Neural Networks
SVM	Support Vector Machine
RFC	Random Forest Classifier

Abbreviations	Title
ML	Machine Learning
DL	Deep Learning
FMS	Fraud Management Systems
FPS	Fraud Prevention Systems
FDS	Fraud Detection Systems
CNN	Convolutional Neural Network
KNN	K-Nearest Neighbors
CEP	Complex Event Processing
FNR	False Negative Rate
FPR	False Positive Rate
RMSE	Root Mean Square Error
PBX	Private Branch Exchange
SIM	Subscriber Identity Module
VOIP	Voice Over Internet Protocol
MLTs	Machine Learning Techniques
CDRs	Call Detail Records
NN	Neural Network
ANN	Artificial Neural Network
MCC	Matthews Correlation Coefficient
NB	Naive Bayes
SMOTE	Synthetic Minority Over-sampling Technique
ADASYN	Adaptive Synthetic Sampling
GSM	Global System for Mobile Communications
NLP	Natural Language Processing
TFAM	Telecom Fraud Analytics Model
ETL	Extract, Transform, Load
EDA	Exploratory Data Analysis
RBF	Radial Basis Function

Chapter One: Introduction

Fraud widely prevalent in various business domains, including telecom, banking, insurance, e-commerce, and many others. Fraud is any action aimed at performing an illegal use of the infrastructure of any given industry domain via services and subscription misuse to gain profit and disrupt normal business flow Becker et al., 2010. SIMBOX is a common and widely used type of fraud in the telecom industry that causes losses of billions of dollars yearly through bypassing international traffic. This chapter addresses in detail fraud in the telecom Industry and the rapid growth of telecom technologies, number of subscribers and services and the associated challenges related to fraud risk, specially SIMBOX fraud. The impact fraud has on operator revenue, credibility, customer satisfaction and performance was discussed, making detection and prevention vital. Also, this chapter outlines the research gap and the need for improved fraud detection using machine learning and resampling techniques. Study objectives focus on applying SVM and Random Forest models, comparing their performance before and after resampling was addressed too. The Study Significance emphasizes the importance of data mining, sampling strategies, and real-world datasets in enhancing fraud detection accuracy was discussed. Finally, the Thesis Layout provides a roadmap of the chapters, covering literature review, methodology, results, and conclusions.

1.1 Fraud in the Telecom Industry

The new technologies and the fast growth in the telecom industry including fast growth in the number of subscribers, different types of services like telephony, SMS, Data, and Integrated services. In addition, such exapnsion created further challanges such as fraudsters ability to deploy inexpensive, sophisticated, and adaptive fraud solutions Krasić and Čelar, 2022. According to Howell, 2021 in 2021, total global telecom revenues are estimated to be approximately USD 1.8 trillion. The total amount of telecom revenue loss due to fraud is estimated to be 2.22% of revenues or $39.89 Billion from which SIMBOX caused $3.11 Billion of the total losses. Compared to 2019, fraud losses increased by 28% or approximately $11.6 Billion. As a result of this, and the fraud effect on the credibility, customer satisfaction, and performance of telecom operators, fraud detection and prevention are becoming of extreme importance not only for telecom operators but also for telecom solutions providers. Moreover, there is a lack of studies that address fraud in the telecom domain and handles the imbalance issue usually found in datasets related to this fraud domain. In different business domains and especially telecom, a large amount of data is collected due to the recent advancement in technology and storage capabilities.

Such a considerable amount of data needs to be analyzed in order to extract useful information and which is not be possible without the use of Data Mining. Data Mining is extremely helpful for analyzing business performance, customer behavior, extracting valuable information and hidden insights Ghorbani and Ghousi, 2020.

Customer behavior identification is the first step to identifying if the customer is suspicious, has any abnormalities, or is actually fraudulent. Identifying customer behavior poses a challenge in this field of study due to imbalanced data. Fraud datasets typically contain fewer fraudulent cases compared to normal non-fraudulent cases. Additionally, the absence of comparison studies for various resampling techniques further complicates the identification process. Different sampling techniques like Random Under-sampling, Random Over-sampling, K-Medoids Under-sampling, and SMOTE Over-sampling are used to handle the imbalanced data problem while predicting the fraudulent behavior using different ML algorithms including Support Vector Machine (SVM), and Random Forest Classifier to better check the performance of the different re-sampling techniques.

1.2 Thesis Statement

Fraud detection and prevention is a complex domain, it is very complex for electronic systems to automatically handle fraudulent cases and automatically prevent heavy losses due to several challenges related to this domain Abdallah et al., 2016. Research related to fraud detection in telecom and other business domains has some limitations, challenges, and motivations. Most of the existing models for SIMBOX detection focus on analyzing target subscribers raw EDRs for detecting suspicious behavior using different machine learning techniques. To the best of our knowledge reviewing existing research in this field, there is a lack of entity-based analysis and resampling comparisons, no studies have performed the analysis using the subscriber as an entity, an entity composed of subscriber behavior and engagement within a telecommunications network, subscriber-centric variables and Service acquisition details. Moreover, we have noticed a lack of studies that address the high imbalance issue usually found in datasets related to this fraud domain. Therefore, finding the optimal combination of ML approach and sampling technique is still an open question that needs further investigation and analysis. So, in this research work we aim to propose the use of ML and a sampling technique that provides the most accurate results on real dataset for SIMBOX fraud detection.

1.3 Study Objective

The goal of the study is to address in detail the approach of using Machine Learning (ML) techniques for SIMBOX detection using Support Vector Machine (SVM), and Random Forest Classifier (RFC) and compare the accuracy before and after applying different resampling techniques to handle the imbalanced dataset we usually have in this type of anal-

ysis to identify the best approach for handling imbalanced data issues. Also, the possibility of using reinforcement learning to enhance the model performance will be addressed based on the generated results quality and models accuracy. Accordingly, the objectives of the study are:

- Develop and implement Machine Learning (ML) models, specifically Support Vector Machine (SVM) and Random Forest Classifier (RFC), for SIMBOX fraud detection.
- Evaluate the impact of data imbalance on fraud detection accuracy and apply various resampling techniques (Random Under-sampling, Random Over-sampling, K-Medoids, SMOTE) to mitigate this issue.
- Compare model performance before and after resampling to identify the most effective approach for handling imbalanced datasets.
- Identifies the criteria for effective SIMBOX fraud detection by integrating ML algorithms with optimal sampling strategies on real-world telecom datasets.
- Enhance efficiency by reducing false positives cases and save human resources and increase business operation effectiveness.
- Investigate the potential of reinforcement learning to further enhance model adaptability and accuracy.

1.4 Study Significance

This study addresses the critical challenge of fraud detection within the evolving telecom industry, where significant revenue losses occur despite advanced services. It focuses on enhancing fraud detection accuracy through a combination of data mining techniques, various sampling strategies to mitigate data imbalance, and the application of machine learning algorithms on real-world telecom datasets. The research aims to evaluate and compare the effectiveness of different approaches, particularly focusing on resampling methods in conjunction with machine learning models like SVM and Random Forest, to identify the most accurate and reliable methods for detecting fraudulent behavior, including SIMBOX fraud. The following points state the significance of the study:

1. The adaptation of Data Mining techniques to prepare a dataset for machine learning.
2. The use of different sampling techniques to solve the imbalance issue in the dataset.
3. The use of a real telecom dataset to identify fraudulent behavior with high accuracy using multiple ML algorithms.

4. Compare the different ML algorithms' performance with different sampling techniques.

5. Identifying the main criteria for effective SIMBOX fraud detection.

1.5 Models Selection

In this study RF and SVM algorithms were selected to build the models and compare the accuracy and performance and this was due to several reasons. As the data set composed of mixed feature types (numerical behavioral metrics and categorical attributes) and shows high class imbalance. RF in terms of suitability to the data characteristics is well□suited for such data as it captures complex non linear interactions without requiring explicit feature transformation, robust to noise and outliers due to ensemble averaging and handles high-dimensional data and correlated variables effectively Aburbeian and Ashqar, 2023. While, SVM specifically with non-linear kernel is high accurate in situations when decision boundary between classes is complex and not linearly separable, number of features is relatively large compared to the number of fraudulent observations Alshawi, 2024. Also, each algorithm has different learning principles as RF is a tree based ensemble model, relying on bagging and majority voting, while SVM is a margin based classifier, optimizing a global objective function Wang et al., 2025. So, by using these two algorithms we are not only evaluating their results before and after sampling. We are evaluating an ensemble, data driven approach (RF) versus theoretically grounded, margin maximization approach (SVM). This strengthens the methodological accuracy of the study by avoiding reliance on a single modeling assumption. In addition to that, the proven effectiveness in fraud detection literature, was another reason for choosing these two algorithms Lee et al., 2025. RF and SVM are widely used benchmark models in fraud detection and anomaly detection research. RF provides feature importance measures, which help identify the key drivers of fraudulent behavior and support domain interpretation. SVM prioritizes predictive generalization and robustness serving as a strong benchmark for classification performance Lee et al., 2025.

1.6 Telecom Fraud Detection in Palestinian Context

Palestine's telecom sector is relatively small compared to regional markets with limited operators and infrastructure. Like any emerging markets in regions with regulatory constrained oversight and limited technological resources in which SIMBOX has been reported, Palestine particularly vulnerable Subex, 2025. Beyond the high financial losses SIMBOX fraud cause. In Palestine, telecom fraud has security implication as telecommunication channels are highly sensitive in politically complex environment. So, effective

fraud detection systems contributes not only to economic resilience but also to national security and customers trust.

1.7 Thesis layout

Chapter One: Introduction This chapter introduces the research problem of fraud in the telecom industry, outlines the thesis statement, and defines the study's objectives and significance, including a description of the real-world, imbalanced telecom dataset used.

Chapter Two: Literature Review This chapter provides a comprehensive review of the existing academic literature on telecom fraud, different types of fraudulent activities (especially SIMBOX), established fraud management systems, and the application of machine learning and data sampling techniques in fraud detection.

Chapter Three: Methodology This chapter details the systematic research process, including the steps for preparing and analyzing the dataset, handling missing data and outliers, applying normalization and dummy coding, and implementing various data sampling techniques to prepare for the development and evaluation of machine learning models.

Chapter Four: Results and Discussion This chapter presents the empirical results of implementing the Random Forest and Support Vector Machine models, evaluating their performance based on metrics like FNR, FPR, and RMSE, and comparing the efficiency of their model-building duration across different data sampling techniques.

Chapter Five: Conclusion This chapter summarizes the study's key findings regarding the effectiveness of different machine learning models and data sampling techniques for telecom fraud detection, discusses the implications of these results, identifies limitations, and suggests avenues for future research.

Chapter Two: Literature Review

This chapter explores the current trends and conducts an in-depth review of the literature concerning fraud detection using machine learning (ML), both in a general context and specifically in relation to International Traffic Bypass. It delves into various machine learning algorithms utilized for fraud detection, each of which exhibits distinct advantages, drawbacks, and constraints.

2.1 Background

Fraud detection and prevention is a complex domain; it is very complex for electronic systems to automatically handle fraudulent cases and automatically prevent heavy losses due to several challenges related to this domain Abdallah et al., 2016. Research related to fraud detection in telecom and other business domains has many limitations, challenges, and motivations.Becker et al., 2010; Chu et al., 2022; Maes et al., 2002 Outlines several limitations and motivations regarding fraud detection:

1. Datasets are highly unbalanced, and only a small portion of the available data are fraudulent cases, which makes producing an efficient model more difficult.
2. The dynamic nature of fraud keeps changing and so the model needs to keep changing to adapt to the change.
3. Overlapping patterns and noisy data.
4. Lack of publicly available datasets due to confidentiality, regulatory, and security limitations.
5. Large amount of data to process.
6. Real-time detection support.
7. The high impact that fraud actions have on operators' revenues and customer satisfaction makes a strong motivation for research.

2.2 Telecom Fraud Definitions and Classification

There are many types of fraud in the telecom industry; different researchers have categorized them differently based on the method used to gain unauthorized access and gain revenues. According to Estévez et al., 2006; Yelland, 2013 there are seven major types of fraud in the telecom industry:

2.2.1 Subscription Fraud

This type of fraud is also known as no intention to pay fraud, subscribers usually subscribe to the service, use the service and exploit the service and then default on the bill payment. This type is mainly common in post-paid offers, where subscribers subscribe and use the service prior to paying any fees. Fraudsters usually use a number of techniques to overcome the controls implemented by the mobile operators to overcome this type of fraud to create an impression that they are ordinary users, not fraudsters.

2.2.2 System Access Fraud

Also called Private Branch Exchange (PBX) hacking, unauthorized PBX systems access using the system maintainers' means, like freephone numbers or weak passwords and then after gaining total access to the system fraudster starts to make outgoing calls and generate revenue by providing services at a cheaper rate than through the legitimate operators.

2.2.3 Call selling

Call Selling is the practice of selling services to end consumers at a lower rate than the legitimate service providers, using another person's system and equipment. This fraud is usually used for selling international calls at lower rates for which the legitimate operators have to pay to their interconnect carriers for that calls and at the same time their customers, usually postpaid will experience bill shock and will ask for the option of waving part of their bill and possible churn which will have very bad impact on the operator revenue and customer base.

2.2.4 Premium-rate Service Fraud

Also called Wangiri Fraud, a Japanese word meaning 'one (ring) and cut'. This fraud relies on the curiosity of phone users when receiving missed calls from unknown international numbers. The fraudsters usually make millions of missed calls, 'one ring call', and when the target user calls back, the number it is usually connected with the premium-rate line of high cost from which the fraudster makes money as a share of the call revenue.

2.2.5 Internal fraud

Type of fraud that should be detected through the implementation of regular internal security audits and routine intelligence gathering. There are several fraud categories under this type of fraud:

1. Dealer's Sales Fraud, which occurs as a result of inflating sales figures to achieve sales targets and generate high commissions.

2. Packet Splitting, This is related to dealers selling shipped handsets and related accessories, usually with associated credit separately instead of selling them as a single product to gain more profit by selling them separately.

3. Unauthorized Credit Adjustments, occurs simply when credit adjustments are performed for accounts without request from the credit department officially.

4. Voucher Fraud occurs when the serial/PIN numbers of top-up voucher cards are disclosed.

5. Leaking Credit Card Information, occurs when information related to credit card payments is leaked to contacts outside the organization.

6. Selling Billing Information, related to selling billing records of individuals to any unauthorized persons, not to a law enforcement agency.

2.2.6 SIM card cloning

In this type of fraud, legitimate SIM cards are duplicated and subsequently inserted into secondary devices. These cloned units are then utilized to generate network traffic, with the resulting costs being billed back to the original subscriber. The primary detection methodology involves the execution of a velocity check on the suspect SIM. Through this process, the physical feasibility of travel between two distinct cell sites is evaluated, specifically within the time interval recorded between two consecutive calls one legitimate and the other potentially fraudulent. If the calculated speed required to traverse the distance between these coordinates exceeds realistic physical limits, a high probability of SIM cloning is established.

2.2.7 Interconnect Traffic Bypass, SIMBOX

This type of fraud involves alteration of the call flow of international traffic using a special device that receives international calls via VOIP techniques and bypasses the traffic to the destination called number and gains revenues on the account of telecom operators. In normal call flow, the international calls are terminated via legitimate routes between the source and destination operators. The source operator has to pay the destination operator a fixed fee per minute for the terminated traffic on their network. Using SIMBOX, the calls are not terminated via the legitimate routes, and the destination operator loses the termination fees of the traffic since the calls established to terminate the traffic to target called numbers are done by establishing on/off-net calls to destination subscribers benefiting from the low-cost on/off-net calls offers. As a result, the call termination fee, which should be paid by the originating operator to the destination operator, is lost and the

destination subscribers receive bad-quality calls and usually with the wrong national or unknown number. Figure 2.1 below illustrates the normal and fraudulent call scenarios.

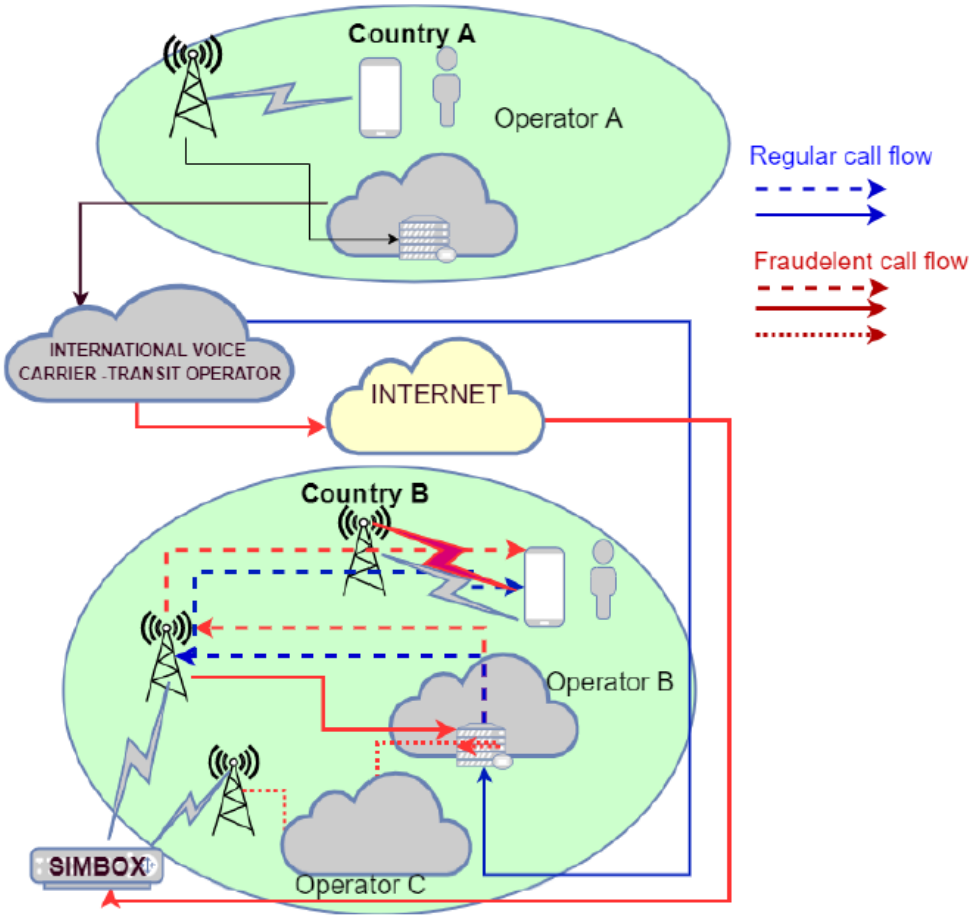


Figure 2.1: On-net and Off-net SIMBOX bypass Krasić and Čelar, 2022

A regular call from Operator A to Operator B is established via standard interconnection routes and gateways between both operators. Operator A pays fixed interconnection fees for each minute terminated by operator B, while in a fraudulent scenario the calls are converted to VOIP and transferred over the internet to SIMBOX device in the destination network country (Operator B). Then SIMBOX device using installed destination operator country local SIM cards can be Operator B SIM cards or any other local operator SIM card establishing normal local on/off-net calls to originally called destination numbers.

2.3 Fraud Management Systems

With modern technologies and global communications, fraud risk is rapidly increasing in various business domains. fraud management system (FMS) incorporates various activities for fraud detection and prevention. The following subsections will discuss in detail the various strategies:

2.3.1 Fraud Prevention Systems (FPS)

As the name implies, such systems' main goal is to prevent fraud from occurring in the first place as the first level of defense. Due to the known fact that fraudsters change their methods and adapt them easily with new prevention rules and security limitations, there is a lack of such systems designed to detect and stop fraudulent activities across industries such as banking, telecoms, e-commerce, and healthcare through combining rule-based engines with machine-learning models to analyze transactions in real time Kouam et al., 2021. Typical features analyzed include identity verification, anomaly detection, device fingerprinting, and behavioral analytics by monitoring large data streams; these systems identify suspicious patterns like unusual login locations or abnormal spending Erastov and Balytska, 2025. Table 2.1 shows some widely used Fraud Prevention Systems:

Table 2.1: Examples of Fraud Prevention Systems and Their Key Features

System	**Industry Focus**	**Distinctive Feature**
FICO Falcon F. COM, n.d.	Cards/Banking	ML scoring in milliseconds
SAS Fraud Management S. COM, n.d.	Banking/Insurance	Streaming analytics
ACI Proactive Risk Worldwider, n.d.	Payments/ATM	Neural networks
RSA Suite Suite, n.d.	Online Banking/E-commerce	Behavioral biometrics

2.3.2 Fraud Detection Systems (FDS)

The detection systems are considered the next level of defense which we will address in detail in this study. Its main goal is to detect fraud occurrence and identify fraudulent activities and report detected cases to fraud analysts to further investigate and take the needed actions, like suspending or disabling the fraudsters. FDSs continuously monitor transactions, user behavior, and network patterns to uncover anomalies that may indicate fraudulent behavior. Modern FDS platforms often integrate multiple data sources such as payment gateways, customer profiles, device fingerprints, and geolocation data to create a comprehensive risk assessment in real time. A major challenge for FDSs is the ability to flag high-risk activities within milliseconds to prevent financial loss before a transaction is finalized without affecting business flow. FDSs are generally divided into two main categories as per their working mechanism, rule-based systems that depend on predefined rules and thresholds that are used to identify fraudulent cases while the other using statistical or machine learning algorithms for detecting fraud using either supervised, unsupervised or Semi-Supervised algorithms Sallehuddin et al., 2015.

Rule-based systems remain valuable for their transparency and ease of auditing; they allow compliance teams to specify thresholds such as transaction limits, unusual login times, or geographic restrictions. However, rule-based systems usually generate a high number of false positives and require constant manual updates to remain effective against evolving tactics. These challenges are addressed by ML systems that learn from historical fraud patterns and continuously adapt to new schemes. While supervised algorithms train on labeled datasets of past fraud cases that are known, enabling precise classification of future events, unsupervised algorithms detect outliers in unlabeled data, which is useful when fraudulent examples are scarce and semi-supervised techniques blend both methods to maximize detection accuracy while minimizing the need for extensive labeled data Phua et al., 2010.

2.4 Fraud Detection Using ML

In ML, fraud detection is a simple classification problem; researchers usually us data mining and machine learning to detect and prevent fraud actions and possible revenue loss. They extract lots of features from large datasets of millions of records like numbers of calls, different numbers called, different calling locations, types of calls, frequency of calls, etc.Azad and Morla, 2012; Becker et al., 2010; Jiang et al., 2013 Over the past two decades, fraud detection methodologies have evolved from manual checks to advanced ML models, notably effective in recent times Raghavan and El Gayar, 2019. Several machine learning-based approaches have been used for fraud detection and prevention including supervised, non-supervised and semi-supervised Babaei et al., 2019.

The paper Omar et al., 2018 reviews fraud detection research, particularly focusing on machine learning techniques used within the last decade. It examines 80 relevant articles retrieved from various digital libraries, identifying common machine learning methods like Artificial Neural Networks (ANN), Decision Trees, Support Vector Machines (SVM), Naive Bayes, Random Forest, and K-Nearest Neighbors (K-NN) in fraud detection systems. The strengths and weaknesses of these various techniques are analyzed, with the prevalent usage of supervised approaches across diverse fraud domains being highlighted. It is noted in Omar et al., 2018 that current research is significantly hindered by the lack of generally available datasets for the purposes of testing and comparative analysis. To address these constraints, several future research directions are suggested. The importance of publicly shared datasets is emphasized as a means to enable more robust comparisons between detection methods. Furthermore, the combination of multiple classifiers is advocated as a potential avenue to enhance the performance of individual models. Finally, the necessity for broader reviews within fraud detection research is acknowledged in Omar et al., 2018, with the aim of advancing institutional understanding and encouraging the development of innovative approaches.

Authors in Kashir and Bashir, 2019 address the operators' struggle to innovate services to minimize costs and preventing revenue loss. Their research applied machine learning algorithms (MLTs), commonly used in diverse fields, to classify normal and fraudulent subscribers (SIMBOX). They used call detail records (CDRs) as input with 25 selected attributes per customer important for differentiating between normal and fraudulent subscribers using Neural Network (NN) and Support Vector Machine (SVM). Performing comparative analysis shows that SVM with Polynomial, Radial, and Sigmoid kernels achieve the best performance, with 99.24% accuracy. Meanwhile, SVM Linear kernel performs least with 95.18% accuracy. For NN, Bayesian Regularization and Resilient Back-Propagation exhibit the best (99.87%) and worst (99.53%) performance, respectively.

In Kouam et al., 2021 , one of the most comprehensive surveys on SIMBOX bypass fraud is presented. The vast scale of the problem is highlighted, with global telecommunications losses estimated at over $28 billion annually. The exploitation of the significant price gap between international and local call termination rates is explained in detail. The paper Kouam et al., 2021 describes how SIMBOX devices readily available through online channels allow international traffic to be rerouted via VoIP networks and terminated as local calls. Consequently, both government tax revenues and operator profits are diminished. Beyond financial depletion, the degradation of customer experience is emphasized, specifically regarding poor audio quality, call delays, and the suppression of caller IDs, all of which erode network trust. The paper Kouam et al., 2021 analyzes also geographic prevalence of this fraud which is noted to be highest in developing regions, particularly Africa and the Middle East, though its presence in Europe and the Americas is also acknowledged. A holistic view of the fraud ecosystem is offered by synthesizing insights from scientific literature, industry reports, and anti-fraud manufacturers. The architecture of these devices is explained, alongside the evolution of fraudster strategies designed to mimic human behavior, thereby making detection increasingly complex.

Detection methodologies are categorized into active and passive approaches; however, it is pointed out that many academic contributions remain limited or outdated due to restricted access to real world Call Detail Records (CDRs). Furthermore, a warning is issued regarding the emergence of 5G and 6G technologies, which may introduce novel vulnerabilities and facilitate more sophisticated schemes. In Kilinc, 2022, telecommunication fraud and anomalies are explored through the application of various machine learning tools. A dataset comprising 417,000 call records across 217 countries was analyzed. K-means clustering was employed for the segmentation of categorical variables, while a combination of supervised models (XGBoost, Extra Trees, Random Forest) and unsupervised models (Isolation Forest, Mixture of Experts) was utilized for anomaly detection.

Anomaly scores, which were generated from model predictions, resulted in the flagging of 1% of total calls as suspected fraud. These findings are noted to align closely with existing industry reports. The utilization of massive data sources, specifically Call Detail

Figure 2.2: Proposed Machine Learning Model Kilinc, 2022

Records (CDRs), is emphasized for the purpose of fraud detection through the application of diverse analytical and hybrid methodologies. While the challenges of extracting fraud signatures from vast datasets are acknowledged, the significance of clear, visualizable alerts is highlighted. Furthermore, the careful consideration of false positives and false negatives is presented as a critical factor in the refinement of detection systems. To further this objective, five distinct machine learning approaches were adopted, with the primary goal of reducing false alarms. These methods are deemed suitable for both real time and batch processing studies. It is suggested that future research be directed toward the development of behavior based, real time detection models to enhance the responsiveness of these systems.

Table 2.2: Scoring Supervised Learning Models Kilinc, 2022

Supervised Learning Classifiers	Precision	Recall	F1-Score	Accuracy
XGBoost	0.97	0.40	0.57	0.99
Random Forest	0.96	0.81	0.88	1.00
Extra Trees	0.95	0.85	0.90	1.00
The Mixture of Experts	0.98	0.79	0.88	1.00

The critical issue of fraud detection within telecommunications is tackled in Chu et al., 2022, where the evolving challenges posed by sophisticated fraudulent tactics often designed to evade conventional algorithms are addressed. An innovative detection model is introduced, founded upon interconnected spatial-temporal call patterns, an area that has been rarely explored in existing literature. Through the extension of statistical features into dynamic call patterns and the application of probabilistic models, user behaviors are effectively captured. Sequential and structural patterns are uncovered through the utilization of Hidden Markov Models and Graph-SAGE models, respectively, resulting in the generation of a specific fraud score for each user. The superiority of this model in accurately detecting fraud, compared to contemporary methods, is demonstrated through experiments involving real-world telecom datasets, thereby affirming its efficiency and in-

terpretability. Advanced fraud types are investigated, and a probabilistic behavior model is designed and validated through various experiments and case studies. By integrating spatial-temporal dimensions into the analysis of user behavior, the capability to identify sophisticated fraudsters who are often camouflaged as regular users is significantly enhanced. The classification of calls as either fraudulent or legitimate for the detection of SIMBOX fraud was addressed in Daka and Nyirenda, 2022 through the utilization of Artificial Neural Networks (ANN). Call attributes were extracted from Call Detail Records (CDRs), and a subset of 7,006 unique records was selected from a total of 13,398. To highlight patterns indicative of fraudulent behavior, feature selection was meticulously employed. The ANN was implemented via the Neural Network pattern recognition tool in Matlab. Due to the utilization of a balanced data representation, a classification performance of 100% was exhibited on the test dataset. It is asserted that ANN serves as a highly successful tool for the detection of SIMBOX fraud, with a demonstrated capability for identifying anomalies in calling patterns.Ultimately, it is recommended that future research be focused on specific monthly periods targeted by fraudsters. Furthermore, the combination of diverse data mining techniques and the employment of multiple models with varying algorithms are suggested to enhance the reliability of results and more effectively tackle fraudulent activities.The significance of Call Detail Records (CDR) as a pivotal source of information for the mobile operator industry and broader societal advancement is underscored in Geepalla et al., 2018. It is noted that valuable insights can be uncovered and irregular customer behaviors can be detected through the rigorous analysis of CDRs. Consequently, companies are aided in their strategic future planning. However, it is also acknowledged that the analysis is made increasingly complex by the massive volume of data inherent in these records. In Randhawa et al., 2018 researchers address the growing problem of credit card fraud, which causes billions of dollars in losses annually. As traditional fraud detection methods struggle with highly imbalanced datasets and evolving fraud techniques. Researchers propose using machine learning algorithms, enhanced with AdaBoost and majority voting ensembles, to improve fraud detection accuracy and robustness. Using a real-world dataset from a financial institution (3 months of transactions) and a benchmark dataset (European cardholders, 284,807 transactions, 492 fraud cases). Twelve models including Naïve Bayes, Decision Trees, Random Forest, Gradient Boosted Trees, Logistic Regression, SVM, Neural Networks, Deep Learning, and others. Several techniques have been applied, AdaBoost to strengthen weak learners, Majority voting to combine predictions from multiple models, Noise injection to test robustness, and Matthews Correlation Coefficient (MCC), which balances performance across imbalanced classes. MCC equation 2.1 which is used to measure the quality of two class problems results, it is a balanced measure even if the two classes are of different size, Imbalanced Dataset. MCC value ranges from -1 to 1, the closer to one is the better the model performance and the closer to -1 is the worst while 0 indicates complete random guessing. Their results show

that most individual models achieved high overall accuracy (99%), but fraud detection rates varied widely (e.g., 32.5% for Random Trees vs. 83% for Naïve Bayes). AdaBoost Improved fraud detection rates slightly (e.g., SVM fraud detection rose from 79.8% to 82.3%), while majority voting produced the most balanced and robust results. For example, NN+NB achieved the highest MCC score (0.823), outperforming individual models. The study demonstrates that ensemble methods (AdaBoost, majority voting) outperform single models in fraud detection. Provides one of the few studies using real-world credit card transaction data, not just benchmark datasets. Highlights the importance of handling imbalanced datasets and using robust evaluation metrics like MCC.

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{2.1}$$

In Mitiku et al., 2024 researchers address SIMBOX fraud as a major challenge in telecommunications that causes revenue loss, degraded service quality, and security risks for operators. They proposed a near real-time fraud detection model using machine learning, a study focusing on Ethio Telecom. Significant financial losses, undermines customer trust and limitation of traditional fraud management systems (rule-based/manual) against evolving fraud schemes were the main motivations for their study. The main research questions are which features can distinguish legitimate vs. fraudulent SIMBOX traffic and which machine learning methods best predict SIMBOX fraud patterns. In their methodology, they retrieve CDRs for voice, data, and SMS traffic logs from Ethio Telecom systems on hourly basis. Perform the required cleaning, integration, feature selection, and aggregation of subscriber-level attributes to extract subscriber behavioral indicators (e.g., call frequency, duration, and unique numbers). They split the datasets into 1-hour, 1-day, and 7-day samples. Researchers tested Random Forest (RF), Support Vector Machine (SVM), and Neural Networks (NN). The results show that Random Forest (RF) and Neural Networks (NN) achieved 100% accuracy across all datasets (1-hour, 1-day, 7-days) while SVM performed less effectively compared to RF and NN. Near real-time detection is feasible using hourly CDR aggregation, though limited by infrastructure constraints. A data-driven framework, specifically tailored to the infrastructure of Ethio Telecom, is provided within this study. It is demonstrated that through the rigorous analysis of Call Detail Records (CDRs) and the application of machine learning most notably Random Forest (RF) and Neural Networks (NN) highly accurate, near real-time detection of SIMBOX fraud can be achieved by telecom operators. Consequently, both revenue and service integrity are protected, even when challenges such as computational overhead and detection latency are encountered.

The escalating challenge of SIMBOX fraud within mobile networks was explored in

Bouguettoucha, 2024, where the focus was directed toward the resulting impact on telecom operators and the presentation of a practical solution. The mechanics of SIMBOX fraud involve the rerouting of international calls through devices containing multiple SIM cards, by which process calls are made to appear local. Significant revenue losses and degraded service quality are frequently incurred by operators as a result of this practice. An AI-driven approach was designed for the detection of such fraud, utilizing Call Detail Records (CDR) sourced from Djezzy, a major Algerian operator. A detailed analysis of the CDR data was conducted, with particular concentration on the final two days of activity for each SIM to ensure that behavioral shifts typical of fraud were captured. During the data preprocessing stage, records were cleaned, missing values were addressed, and outliers were removed. Furthermore, the feature space was expanded through the creation of 24 new metrics, including total call counts, unique cell visits, and call ratios. To mitigate the severe imbalance between fraudulent and legitimate instances, several balancing techniques were tested, such as SMOTE, ADASYN, and a hybrid method combining Tomek Links with Random Undersampling. The performance of various machine learning algorithms, including Artificial Neural Networks (ANN), SVM, Random Forest, and XGBoost, was evaluated. It was determined that the highest performance was achieved by the XGBoost model when combined with Tomek Links and Random Undersampling, resulting in accuracy, precision, recall, and F1-scores of 96%.Through this model, a robust balance was maintained between the identification of fraudulent SIM cards and the minimization of false positives. The work was concluded with the development of a Streamlit-based web application, through which CDR datasets can be uploaded, automated preprocessing can be executed, and fraud detection results can be obtained by telecom operators. Two distinct datasets are generated by the application: a comprehensive record where fraudulent entries are highlighted, and a separate file containing exclusively flagged numbers. It is demonstrated through this research that fraud detection within highly imbalanced datasets can be significantly improved when robust feature engineering is combined with effective data balancing and advanced algorithms such as XGBoost. Furthermore, the accessibility and utility of the solution for real-world telecom operations are ensured through its practical deployment as a web application.

The escalating challenge of SIM box fraud in mobile networks was also explored in Elmi et al., 2012, where the implementation of neural networks was proposed as a primary detection tool. Over 234,000 calls made by more than 6,000 users were analyzed using real GSM call detail records from prepaid subscribers. Within this dataset, roughly one-third of the calls were confirmed as fraudulent. Nine behavioral features were identified by the authors to distinguish fraudulent actors from legitimate customers. These metrics included the number of unique contacts dialed, call duration patterns, nighttime activity, and the ratio of outgoing to incoming calls. Using a supervised learning approach with a multi layer perceptron, they tested different network architectures and parameter set-

tings to find the most effective model. Their best configuration, with two hidden layers and carefully tuned learning rate and momentum values, achieved an impressive accuracy of 98.7 percent, with very few false positives or negatives. The study demonstrates that neural networks can capture subtle patterns in subscriber behavior and provide telecom operators with a reliable method for detecting SIM box activity. The authors conclude that while ANN models are highly effective, future work should also compare their performance with other machine learning techniques such as support vector machines.

In paper Bello et al., 2023 researchers addresses the practical challenges of real-time implementation and computational cost in AI-driven fraud detection systems. They found that there are two main types of challenge, practical challenges and computational cost challenges. Practical challenges include data quality and preprocessing, scalability, latency, integration with legacy systems and regulatory compliance. Real-time systems require clean, consistent, and complete data streams while processing millions of transactions per second to detect fraud instantly before transactions are finalized. Even small delays can result in financial losses or customer dissatisfaction. Advanced AI models required to integrate with legacy systems without disrupting business operations and must meet strict financial regulations while maintaining speed and accuracy. Computational cost challenges include high resource demand, execution time, infrastructure costs and optimization needs. Training and tuning advanced models like deep learning, ensemble methods requires significant computational power and memory. Also, continuously running real-time fraud detection and optimization techniques requires powerful servers or cloud resources, increasing operational expenses. This makes careful optimization, efficient infrastructure, and continuous data management essential to overcome these challenges.

In Zareapoor, Shamsolmoali, et al., 2015 Authors performs comparative analysis for fraud detection using KNN, SVM, Naïve Bayes, and Bagging Ensemble Classifier and they discussed the various concerns and challenges handling the fraud detection problem like non-availability of real-world data due to confidentiality, security, and regulatory concerns. Also, they addressed the highly unbalanced data issue as the number of fraudulent transactions usually is only 2% of the total number of samples. They also discussed the challenges related to big data analysis and computational time needed for such cases and the dynamic nature of fraud which makes regular updates for ML algorithms an essential step. They used the dataset of 100,000 instances of credit card transactions with 20 attributes from an e-commerce website with only 2,293 fraudulent instances/transactions. Their analysis shows that the True Positive Rate (fraud catching rate), True Negative Rate, False Positive Rate (false alarm rate), and False Negative rate reflect the performance more than the accuracy and the error rate. They also used 10-cross-fold validation for their experiment. Their results show that KNN had a much better false alarm rate and fraud catching rate as compared to SVM and Naïve Bayes Classifier.

While in Sallehuddin et al., 2015 they performed a comparative study for two machine learning algorithms, SVM and ANN using an experimental fraud dataset of subscribers EDRs. Their dataset contained 234,324 calls performed by 6415 different subscribers from one Cell only. The dataset is composed of 2,126 (33.14%) fraudulent subscribers and 4,289 (66.86%) non-fraudulent. They found that the accuracy of the SVM model is higher than ANN, it gives an accuracy rate of 99.06% compared with 98.71% accuracy for ANN, also they found that in addition to accuracy, SVM outperforms ANN in computational time for model building and training Figure 2.3.

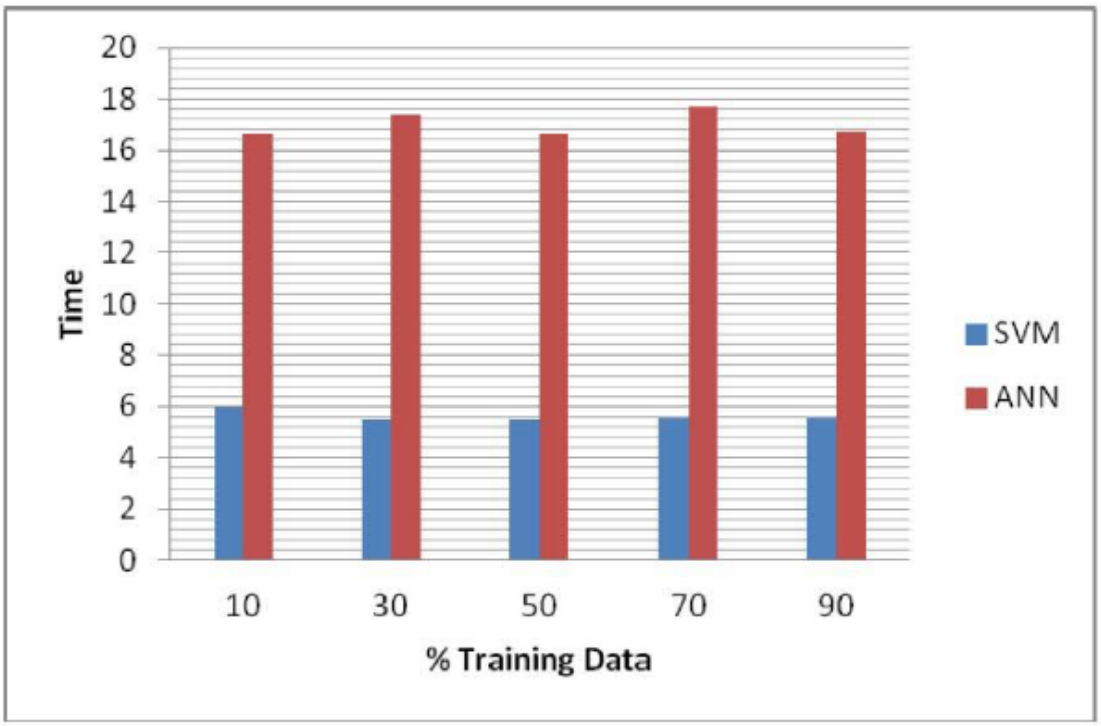


Figure 2.3: Comparison of SVM and ANN model training duration Sallehuddin et al., 2015

While in Raghavan and El Gayar, 2019 researchers used empirical study to compare ML and Deep Learning (DL) models, providing insights into their performance on varied datasets, aiming to guide practitioners and organizations seeking optimized fraud detection methods. They used three different datasets the Australian, European (EU), and German dataset to benchmark different ML and DL technique using empirical study comparing different generated models, their comparative analysis results shows that SVM gives the best results with large datasets and combining it with Convolutional Neural Network (CNN) gives even more reliable results while with small datasets Convolutional Neural Network (CNN) outperform other deep learning approaches. A limitation of their study is the dynamic nature of fraud which makes supervised ML algorithms like CNN and RF not work well in a dynamic environment and need to be retrained using newly collected datasets.

Paper Zhao et al., 2018 proposed a different approach for fraud calls detection using

NLP for analysis of the content of the call, they collected the textual dataset for the fraud telecommunication from the internet then they used NLP to perform feature extraction and build rules for fraudulent calls detection and finally they developed an application for fraud detection on Android platform that should be installed on the user's handset, their approach results show that it can be implemented and generalized in real-world applications.They introduce a novel approach by analyzing the content of calls rather than just metadata or usage patterns. However, textual datasets collected from the internet may raise concerns about representativeness and data quality. on the other hand, requiring installation on the user's handset could limit adoption due to privacy concerns, resource usage, or user resistance.

Another research paper Marah et al., 2015 addresses the widespread issue of fraud affecting telecommunication companies globally, focusing particularly on bypass fraud in international calls and its substantial revenue losses. The paper Marah et al., 2015 introduces a novel fraud detection technique using user profiling and fuzzy logic. They used a real database from a mobile operator company, they extracted five key features subscribers' mobility, call ratios, cell activity, irregular calls, and service type to establish detection patterns. User behavior is profiled through the analysis of mobile network traffic, by which detection patterns for SIM cards are established. Within this framework, fuzzy logic membership equations are utilized to assess the probability of fraudulent activity. However, it is noted that the verification of the program is hindered by a lack of testing data, thereby limiting the definitive confirmation of fraud occurrences. Despite these constraints, the application of fuzzy logic for profiling is recognized for its flexibility and reliability in the management of extensive datasets.

In Suryowibowo et al., 2019, the identification of SIMBOX usage by unauthorized entities is addressed using a spectrum analyzer and the K-Nearest Neighbor (K-NN) algorithm. Attributes such as location and strong frequency signal are utilized for identification, with a specific focus on GSM DCS 1800 MHz uplink frequency measurements. An accuracy of 81.25% was achieved by the K-NN algorithm through the use of 18 training data points and 32 testing points. It was concluded that K values of 3, 5, and 7 are effective for this purpose, with the highest accuracy observed at K=7. An innovative approach utilizing Neo4j for automated CDR analysis is introduced in Geepalla et al., 2018. The methodology is centered on the transformation of CDR data into a Neo4j format, after which the Cypher query language is employed for automated processing. To validate this proposed approach, a real dataset was utilized. The leverage of big data technologies is recommended for the modeling and analysis of extensive datasets, with particular emphasis placed on the contribution of graph-based methods in understanding user behavior and identifying anomalies. The feasibility of this approach was demonstrated through the application of a practical dataset, by which its potential in real-world scenarios was showcased. Abnormal behaviors were detected through the execution of Cypher queries;

specifically, excessively long call durations, call activity restricted to a single geographic location, and high volumes of outgoing calls coupled with a total absence of incoming traffic were identified. Furthermore, the reliance solely on call services without service diversity was flagged. Through graph-based analysis, patterns of abnormal behavior consistent with fraudulent SIMBOX activity were revealed most notably, accounts were identified making thousands of outgoing calls without any reciprocal contact. It was observed that approximately 0.002% of users conducted all calls from a single location, which is considered a strong indicator of potential fraud. Additionally, calls exceeding 30 minutes in duration were made by 0.6% of users, marking them for further investigation.

In Chang et al., 2017 they proposed a fraud detection model using social network analysis techniques to analyze group members and their influence on each other using a real telecom dataset with 113 fraudsters. This study addresses the growing challenge of telecommunication fraud mainly the significant financial losses worldwide. The study focuses on analyzing the behavior of highly organized, transnational criminal groups using a proposed model called Fraud Analysis Model (TFAM) based on Social Network Analysis (SNA). This is to reveal the hidden structure of telecom fraud organizations and to overcome the limitation of traditional investigation approaches trying to dismantle such groups effectively. The idea behind the proposed TFAM model was to capture real world collaboration and mobility patterns among fraudsters by linking suspects using flight records and co offending data. They found that telecom fraud groups are structured organizations composed of three main departments with distinct responsibilities: telecom, operations, financial and all coordinated by a central leader (boss). Degree, Closeness, Eigenvector, Betweenness, and Structural Holes are identified as the main influential actors within fraud networks. The study results show that the Degree and Eigenvector centrality are the most effective measures for identifying key roles such as the boss, caller chief, and money laundering chief, outperforming closeness and betweenness centrality. The study confirms that by leveraging relational data rather than individual behavior alone, social network analysis can be used for effective fraud investigation and detection and even provides law enforcement agencies with a powerful analytical tool to better understand fraud organizations and design more effective intervention strategies.

2.5 Handling Imbalance in Fraud Datasets through Data Sampling

In general, the datasets used for studying fraud detection in various business domains are usually imbalanced, and the number of fraudulent cases is usually minimal compared with normal non-fraudulent cases. In Awe et al., 2024; Dubey et al., 2014; He et al., 2008 researchers analyzed five data sampling techniques below in addition to the no-sampling approach:

1. No Sampling: The dataset is used without implementing any sampling approaches.

All data points from the majority and minority training sets are used. using this approach we preserves original dataset integrity and avoid artificial bias but cause model bias toward majority class, Poor recall and precision for minority class and Limited generalization.

2. Random Under-sampling: In this approach, all the data points from the minority class are used and instances from the majority class are randomly removed from the dataset until the desired balance is achieved. This approach is simple and straight forward but it has one major disadvantage as some useful information might be lost from the majority class.

3. Random Over-sampling: a technique used to handle imbalanced datasets by increasing the number of samples in the minority class. In this approach, all the data points from the majority and minority classes are used. To achieve the desired balance using this approach instances are randomly selected from the minority class with replacement. Using this approach no new information is added only duplicates existing data and might cause an over-fitting problem in the generated classifier but it is very simple and quick to apply,helps the model give more attention to the minority class and no information is lost from the majority class.

4. K-Medoids Under-sampling: Reduces the size of the majority class by selecting representative samples using the K-Medoids clustering algorithm. This approach is based on an unsupervised clustering algorithm in which the majority class data points are clustered where the number of clusters equals the number of minority training class data points. The cluster centers are the actual data points that are chosen randomly, the process is repeated and the best result (the one with the minimum cost) is selected. The final balanced dataset is a combination of all data points from the minority class and the cluster centers from the majority class. it keeps only the most informative and diverse samples and avoid random deletion (which could remove important data) but it is slower than random under-sampling as it is computationally heavy.

5. SMOTE Over-sampling: SMOTE stands for “Synthetic Minority Over-sampling Technique”, its approach is to generate new synthetic data points for the minority class by randomly generating pairs of nearest neighbors, the minority class is over-sampled. The resulting dataset is a combination of all data points from both classes without elimination and, additionally, the newly generated synthetic minority data points so that the final training dataset is balanced Chawla et al., 2002.This approach is inspired by a technique that proved successful in handwritten character recognition Ha and Bunke, 1997, researchers created extra training data by performing certain operations on real data, and as a result the minority class is over-sampled

by taking each minority class sample and introducing synthetic examples along the line segments joining any/all of the k minority class nearest neighbors. This approach Use k nearest neighbors.No duplicates which reduces overfitting risk but it is computationally heavy, may generate synthetic noise and can create ambiguous samples.

6. ADASYN ADAptive SYNthetic Sampling: an over-sampling technique that adaptively creates synthetic data points for the minority class, it focuses more on the harder-to-learn samples that are more likely to be misclassified using k nearest neighbors (usually k = 5). It measures how many of its neighbors belong to the majority class, if most of their neighbors are majority class it's hard to learn.If most neighbors are minority class it's easy to learn. Generate more synthetic samples for hard-to-learn points and fewer for easy ones. Unlike SMOTE methodology treating all minority points equally, ADASYN adapts more new points where the model needs help, fewer new points where the data is already well represented. While ADASYN focuses on difficult cases, creates new data (not just duplicates) and achieves better class balance with better decision boundaries Smarter than Random Over-sampling or plain SMOTE it is more complex and computationally heavier, requires careful tuning and might introduce noise if not tuned well.

Table 2.3: Comparison of different sampling techniques: Pros and Cons Awe et al., 2024

Sampling Technique	**Pros**	**Cons**
Random Under-sampling	• Simple and straightforward • Very quick to apply • Helps the model give more attention to the minority	• Useful information might be lost from the majority class
Random Over-sampling	• No information is lost	• Might cause an over-fitting problem
K-Medoids Under-sampling	• Keeps only the most informative and diverse samples • Avoids random deletion	• Slower than random under-sampling as it is computationally heavy
SMOTE Over-sampling	• Uses k-nearest neighbors • Generates synthetic samples (not duplicates), reducing overfitting risk	• May generate synthetic noise • Can create ambiguous samples • Computationally heavy
ADAptive SYNthetic Sampling	• Uses k-nearest neighbors • Focuses on difficult cases • Creates new data (not just duplicates)	• More complex and computationally heavier • Requires careful tuning • Might introduce noise if not tuned well

2.6 Summary

In this chapter, an extensive review of the literature regarding fraud detection via machine learning (ML) was presented, with particular attention paid to the bypass of international traffic. The various hurdles inherent in this field such as the prevalence of noisy data, the lack of accessible public datasets, and the necessity for real-time processing are thoroughly examined. It is noted that supervised approaches are predominantly favored, with techniques such as Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Random Forest being frequently utilized within these systems. A variety of methodologies

are discussed, ranging from social network analysis and user profiling to fuzzy logic and anomaly detection. To address the persistent issue of data imbalance, several sampling strategies are explored, including SMOTE over-sampling and K-medoids under-sampling. The advantages and inherent limitations of each technique are weighed to demonstrate how a scarcity of fraudulent instances can be effectively managed. Ultimately, the significance of combining classifiers and utilizing shared datasets is emphasized as a primary direction for future advancements in the field. A wide array of methodologies, including NLP analysis, user profiling, and fuzzy logic, is employed to identify fraudulent patterns within telecom networks. Furthermore, social network analysis and anomaly detection are utilized to further refine these systems. The critical issue of dataset imbalance is addressed through the exploration of various sampling techniques, such as random under-sampling and SMOTE over-sampling. Each of these methods is meticulously described, with particular attention paid to the trade-offs and benefits encountered when fraudulent instances are infrequent. Through this analysis, light is shed on the innovative advancements in machine learning across multiple domains. Finally, the necessity of shared datasets and the integration of combined classifiers are emphasized as essential strategies for the enhancement of future detection performance. It has been found that most of the existing models for SIMBOX detection focus on analyzing target subscribers' raw EDRs for detecting suspicious behavior using different machine-learning techniques. As noted, reviewing existing research in this field, no studies have performed the analysis using the subscriber as an entity constructed from usage patterns over time using data mining techniques in addition to subscriber-level information. Moreover, we have noticed a lack of studies that address the high imbalance issue usually found in datasets related to this fraud domain. Therefore, finding the optimal combination of the ML approach and sampling technique is still an open question that needs further investigation and analysis. So, in this research work, we aim to propose the use of ML and a sampling technique that provides the most accurate results on a real dataset for SIMBOX fraud detection.

Chapter Three: Research Methodology

This study methodology is divided into four different parts, Dataset preparation, and analysis, data sampling, development of the SVM model, and development of the RF model each using an imbalanced and balanced dataset using both oversampling and under-sampling techniques.

3.1 Dataset Description

This research work focuses on performing the required analysis to build a SIMBOX fraud detection model using a real telecom dataset, Table 3.1. The dataset obtained from one of the Palestinians telecom operator. Both Support Vector Machine (SVM) and Random Forest classifier with implementing the various sampling techniques was built to identify the best combination in term of accuracy and performance. The dataset includes the usage patterns for a portion of subscribers who have been detected using traditional fraud system as a suspected fraudster and has been marked by the fraud expert as fraudulent or no-fraudulent during the period 27/03/2020 - 02/11/2022 in addition to some subscriber-level features that are of high value for Identifying possible fraudsters, Table 3.1.

The dataset used for the study composed of 23,017 cases of which 22,289 normal cases and 728 fraudulent cases. The dataset has been generated from big raw data storage of millions of records after applying various data mining techniques. Each entry in the dataset represents a unique case detected and the columns represent the features related to that case. The feature selection is a critical task to provide the best description for the case that is at the same time useful for identifying whether it is fraudulent or not which is the target of the study. Table 3.2, each case is represented by 20 features categorized as follows, customer behavior and engagement within the telecommunications network features, ,customer-centric features and Service acquisition features. A comprehensive view of customer behavior and engagement within a telecommunications network is provided by this dataset, with data captured on a case-by-case basis. Within each record, a unique identifier (CASE_ID) is included, alongside customer-centric variables such as the subscription product (PRODUCT), marketing category (MARKETING_CATEGORY), and geographical region (SUB_REGION). Additionally, a classification flag (CUSTOMER_SEGMENT) is incorporated to categorize the subscriber base.Service acquisition details are also tracked, including the specific dealer and sales channel (DEALER_NAME, SALES_CHANNEL) through which the subscription was facilitated. Furthermore, detailed telecommunication usage metrics are incorporated into the dataset, by which valuable insights into call patterns are offered. This includes

total call count and duration (CALLS_COUNT, TOTAL_DURATION), a breakdown of incoming versus outgoing call activity (COUNT_OUTGOING, DUR_OUTGOING, COUNT_INCOMING, DUR_INCOMING), and the diversity of calling behavior. Network usage is also monitored through metrics such as the number of different cells utilized (COUNT_UNIQUE_CELLS).Furthermore, the data includes performance indicators related to call success, like the total number of failed calls and unique numbers associated with those failures (NUMBER_OF_FAILED_CALLS, FAILED_DIST_CALLED _NUMS). Finally, the dataset offers a case history with detection details, such as the subscriber's age on the network at the time of detection (AGE_ON_DETECTION), the number and frequency of detections (DETECTION_COUNT, DETECTION_FREQUENCE), and the current status of the case via a descriptive name and code (STATUS_NAME, STATUS_CODE).

Table 3.1: Feature Descriptions of Fraud Dataset

Feature Name	**Description**
CASE_ID	Autogenerated Unique Identifier for each case.
PRODUCT	Subscription Product of the subscriber, prepaid, postpaid, hybrid..etc.
MARKETING_CATEGORY	Category of the Subscriber, internal classification based on the subscriber subscription type and payment method.
SUB_REGION	Region of the Subscriber.
CALLS_COUNT	Number of calls performed – all types incoming and outgoing.
TOTAL_DURATION	Total Calls Duration – all types incoming and outgoing.
COUNT_OUTGOING	Number of outgoing calls.
DUR_OUTGOING	Duration of outgoing calls.
COUNT_UNIQUE_NUMBERS_CALLED	Different numbers called.
COUNT_INCOMING	Number of incoming calls.
DUR_INCOMING	Duration of incoming calls.
COUNT_UNIQUE_CELLS	Different Cells used to make calls. Mobility indicator of the case.
NUMBER_OF_FAILED_CALLS	Number of failed call attempts. Calls performed by the subscriber and failed.
FAILED_DIST_CALLED_NUMS	Number of different numbers called, and call has been failed.
CNT_OUT_VS_DIST_OUT	Ratio of outgoing calls count vs the different called destinations.
AGE_ON_DETECTION	Subscriber age on network on detection date.
DETECTION_COUNT	Number of times the subscriber has been detected by the rule-based system.
DETECTION_FREQUENCE	Case Detection frequency, it represents how frequent the case has been detected as suspected fraudulent.
CUSTOMER_SEGMENT	Subscriber classification flag, Internal Classification depending on subscriber profile and spending behaviour.
DEALER_NAME	Source of getting the SIM used, activation channel name.
SALES_CHANNEL	Type of sales channel, activation channel type.
STATUS_NAME	Status name of the case, textual representation of the case classification.
STATUS_CODE	Status code of the case, numerical representation of the case classification.

3.2 System Methodology

In this research, after extracting the data from the source system to be used to build the models. Data pre-processing actions that include handling missing data, removing outliers, normalizing data, and feature extraction was performed. Below is a step-by-step

Table 3.2: Categorization of Features

Feature Name	Category
PRODUCT	Customer-centric Feature
MARKETING_CATEGORY	Customer-centric Feature
SUB_REGION	Customer Behavior Feature
CALLS_COUNT	Customer Behavior Feature
TOTAL_DURATION	Customer Behavior Feature
COUNT_OUTGOING	Customer Behavior Feature
DUR_OUTGOING	Customer Behavior Feature
COUNT_UNIQUE_NUMBERS_CALLED	Customer Behavior Feature
COUNT_INCOMING	Customer Behavior Feature
DUR_INCOMING	Customer Behavior Feature
COUNT_UNIQUE_CELLS	Customer Behavior Feature
NUMBER_OF_FAILED_CALLS	Customer Behavior Feature
FAILED_DIST_CALLED_NUMS	Customer Behavior Feature
CNT_OUT_VS_DIST_OUT	Customer Behavior Feature
AGE_ON_DETECTION	Customer Behavior Feature
DETECTION_COUNT	Customer Behavior Feature
DETECTION_FREQUENCE	Customer Behavior Feature
CUSTOMER_SEGMENT	Customer-centric Feature
DEALER_NAME	Service Acquisition
SALES_CHANNEL	Service Acquisition

summary of the adopted methodology in this study Figure 3.1:

1. Extract the data from the source system and load the data using Python.
2. Prepare the data by making the required per-processing and cleaning to fix any issues related to messing, wrong, and impossible or outlier values.
3. Perform statistical descriptive analysis to extract the required input attributes for classification.
4. Generate data visualizations for various data dimensions.
5. Implements over and undersampling techniques to solve the imbalance issue.
6. Divide the dataset into training, and testing datasets.
7. Performing the classification using Support Vector Machine (SVM), and Random Forest Classifier.
8. Compare the results of the analysis and evaluate the model's performance.

3.3 Extracting Data from Source System

The data extraction and loading process involved raw subscriber data, in the form of EDRs that was extracted from source Mobile Switching Center (MSC) files. This raw data was

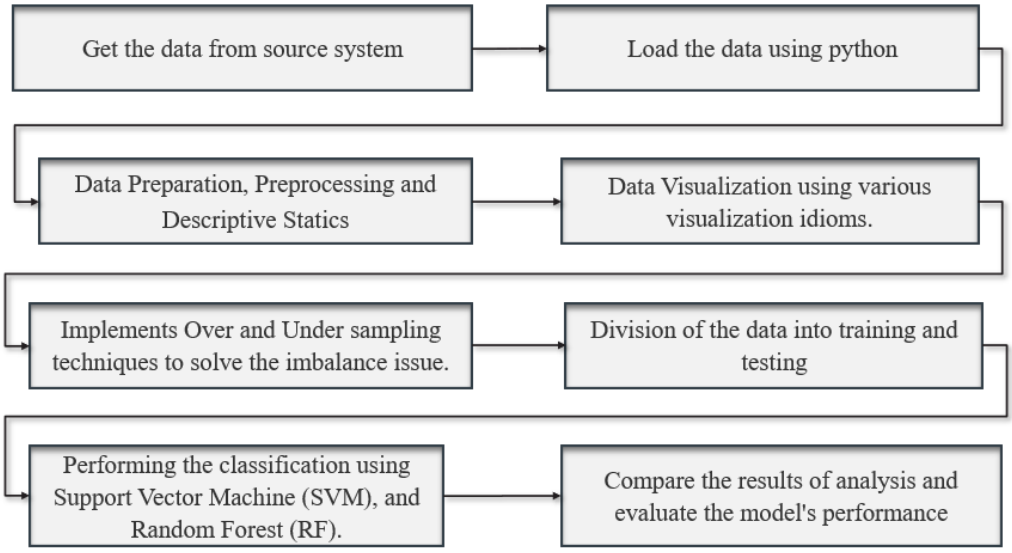


Figure 3.1: System Methodology

then moved to a staging area as part of an Extract, Transform, Load (ETL) pipeline in its raw format. Subscriber EDRs represent the individual events performed by the subscribers; such detailed data is not useful in its raw format for identifying fraudulent behavior at the EDR level. Therefore, the very first step is to use the raw EDRs to build an ensemble of the data on the subscriber level. So, after the loading phase, a series of transformations were applied to the data before it was finally extracted from Oracle database. These transformations enriched the data by deriving meaningful attributes, such as subscriber product, marketing category, and regional information, while also aggregating call details like CALLS_COUNT and TOTAL_DURATION. The result is a refined dataset containing detailed subscriber information and key performance metrics, which is ready for analysis.

3.4 Load The Data Using Python

To facilitate the analysis of subscriber data, a critical intermediate step was undertaken in which the refined dataset was extracted from the oracle database into an excel spreadsheet. Through this process, data was effectively offloaded from the production environment into a more accessible format. Once stored within the Excel file, the data was loaded directly into a DataFrame for analysis via Python's pandas library.This common workflow is streamlined through the application of the pandas.read_excel() function, by which structured data from the excel grid is transformed into a flexible, tabular DataFrame Figure 3.3. A local, file-based copy of the data is thus provided, which is considered ideal for offline processing, collaborative sharing, and the execution of detailed analysis within

Python's extensive data science ecosystem. Alternatively, a method typically utilized in production systems involves the establishment of a direct connection between Python and the Oracle database, allowing the refined subscriber data to be pulled directly for in-depth investigation. Leveraging libraries like oracledb and pandas, a connection to the oracle database is established using the appropriate credentials and a connect string Vajja, 2016. A SQL query is then executed to select the transformed data, such as CALLS_COUNT, TOTAL_DURATION, and CUSTOMER_SEGMENT. The pandas.read_sql function is the most common and convenient method for this, as it directly converts the SQL query results into a structured DataFrame. This process effectively extracts the data from its relational database table and brings it into the Python environment.

3.5 Data Preparation and Preprocessing

3.5.1 Feature Extraction and Handling Missing Data

The EDRs related to each subscriber are summarized into one unique record that describes the subscriber's calling behavior and usage pattern in addition to some subscriber-level features. A total of 20 features has been selected and identified as useful information for detecting SIMBOX fraud. The features have been selected based on the literature studied about the characteristics of SIMBOX fraud and the experience of working staff in the field of telecom fraud.Feature extraction could involve creating new, more informative variables from the existing raw attributes to better capture customer behavior Bouguettoucha, 2024. For example, a new feature calculating the "Call Duration per Unique Number" could be derived by dividing TOTAL_DURATION by COUNT_UNIQUE_NUMBERS_CALLED to provide a normalized measure of engagement. Similarly, an "Engagement Ratio" might be engineered from the proportion of incoming calls relative to total calls using COUNT_INCOMING and CALLS_COUNT.

Handling missing data is another critical preprocessing step, which for this dataset might involve using imputation techniques for variables with missing values. For instance, the DEALER_NAME or SALES_CHANNEL could be populated by filling missing entries with the most frequent category using mode imputation, as these are categorical features. For numerical fields like AGE_ON_DETECTION, median imputation would be more robust to potential outliers than the mean. A more advanced technique could use a K-Nearest Neighbors (K-NN) model to impute missing values by leveraging the data of similar, complete records. For crucial variables where a high volume of data is missing, a specific analysis of the missingness pattern is warranted to avoid introducing bias. Ultimately, a combination of these techniques ensures the data is clean, complete, and contains the most predictive features for downstream analysis and model building.The dataset used for this study did not include that much of messing values; only one feature that represents the subscriber age on the network included empty values and was treated by filling it with

the average value Figure 3.4.

3.5.2 Identifying and Removing Outliers

Outliers are unusual or abnormal data values that can be errors or normal values not consistent with the other observations, or they are against the trend of the data, or they are near the extreme limits of the data ranges. Such outliers are usually removed from the dataset to eliminate any unwanted effects they could have on the accuracy and performance of the model. In this study dataset, to identify outlier values descriptive statistics was performed for the dataset features as shown in table 3.3 below and also used a visual representation for the various features as shown in Figure 3.2. Most of the variables' distributions do not show evidence of a symmetric distribution, which indicates the presence of outliers and extreme values. For example, the total duration feature has a mean of 267 while the maximum value is 1,094, which is way larger than average and is considered an extreme value compared to the average call duration. The same observation for calls count, the mean of the call count is 89, while the maximum call count is 396 which is an extreme value and can be considered an outlier, it is more than 4 times greater than the mean. In addition to descriptive statistics and visual representation for identifying outliers, Z-score standardization is also used. The Z-score of a data point represents the difference between the point and the mean of all the data points divided by the standard deviation of that point, equation 3.1, represents the number of standard deviations the data point is from the mean, a data value is considered an outlier if it is ±3 standard deviations from the mean.

$$Z - Score = \frac{X - Mean(X)}{SD(X)} \tag{3.1}$$

Applying the Z-Score Standardization on call duration feature identifies 131 subscribers as outliers, subscribers whose total calls duration is greater than 810 minutes.

As shown in Figure 3.2, the distribution by call duration looks better after removing the outliers compared to before removing them. The majority of the data points are distributed on the left extreme tail and only a few points are distributed on the right extreme tail, right-skewness. Despite removing the outliers the distribution is still right-skewed but this might be due to the fact that the dataset contains an unequal proportion of fraudulent and non-fraudulent cases, normal subscribers usually have low call duration compared with fraudulent subscribers. It has been found that for some features, the reason for having outliers is due to the imbalanced issue in the dataset since most of the observations that are considered statistical outliers belong to the target fraudulent class which means that if we delete them we will lose a very important part of our dataset and will increase the imbalanced issue we already have between the two classes. Table 3.5.

Table 3.3: Dataset Features Descriptive Statistics (Fraudulent Cases)

Feature	Count	Mean	Std	Min	25%	50%	75%	Max
CALLS_COUNT	728	65.38	11.26	51.00	58.00	62.00	70.00	136.00
TOTAL_DURATION	728	320.91	79.03	146.53	264.01	307.87	366.86	579.28
COUNT_OUTGOING	728	62.05	11.21	50.00	54.00	59.00	66.00	136.00
DUR_OUTGOING	728	320.34	79.04	146.45	263.85	307.21	366.64	579.28
COUNT_UNIQUE_NUMBERS_CALLED	728	59.45	10.13	50.00	52.00	56.00	63.00	116.00
COUNT_INCOMING	728	3.33	2.92	–	1.00	3.00	5.00	18.00
DUR_INCOMING	728	0.57	0.64	–	0.08	0.38	0.83	4.35
COUNT_UNIQUE_CELLS	728	2.00	0.65	1.00	2.00	2.00	2.00	3.00
NUMBER_OF_FAILED_CALLS	728	48.87	41.42	–	–	51.00	73.00	309.00
FAILED_DIST_CALLED_NUMS	728	43.85	35.04	–	–	48.00	65.00	224.00
CNT_OUT_VS_DIST_OUT	728	0.96	0.03	0.84	0.94	0.96	0.98	1.00
AGE_ON_DETECTION	712	5.24	3.97	1.01	2.50	3.69	6.38	25.58
DETECTION_COUNT	728	1.65	0.52	1.00	1.00	2.00	2.00	3.00
DETECTION_FREQUENCE	728	1.62	0.49	1.00	1.00	2.00	2.00	2.00

Table 3.4: Dataset Features Descriptive Statistics (Non-Fraudulent Cases)

Feature	Count	Mean	Std	Min	25%	50%	75%	Max
CALLS_COUNT	22,289	90.33	29.51	51.00	70.00	83.00	102.00	396.00
TOTAL_DURATION	22,289	265.36	180.26	30.03	99.93	236.20	406.75	1,094.43
COUNT_OUTGOING	22,289	84.75	27.62	50.00	65.00	79.00	96.00	382.00
DUR_OUTGOING	22,289	257.66	180.01	30.02	91.28	228.92	400.13	1,073.32
COUNT_UNIQUE_NUMBERS_CALLED	22,289	69.01	19.26	50.00	55.00	63.00	77.00	265.00
COUNT_INCOMING	22,289	5.58	7.59	–	1.00	3.00	7.00	75.00
DUR_INCOMING	22,289	7.69	8.53	0.12	1.00	4.40	13.13	29.98
COUNT_UNIQUE_CELLS	22,289	1.76	0.77	1.00	1.00	2.00	2.00	3.00
NUMBER_OF_FAILED_CALLS	22,289	67.98	164.78	–	32.00	65.00	44.00	2,195.00
FAILED_DIST_CALLED_NUMS	22,289	45.50	102.12	–	22.00	65.00	44.00	1,368.00
CNT_OUT_VS_DIST_OUT	22,289	0.83	0.12	0.50	0.76	0.85	0.94	1.00
AGE_ON_DETECTION	22,252	647.60	770.74	–	53.01	514.01	917.91	4,727.48
DETECTION_COUNT	22,289	271.59	252.25	50.00	201.00	453.00	871.00	–
DETECTION_FREQUENCY	22,289	0.57	0.64	0.01	0.26	0.69	0.69	7.00

3.5.3 Data Normalization

Both SVM and Neural Networks require data to be normalized before being used Kurani et al., 2023. This is to prevent features that contain large values from impacting the algorithm's processing power simply because it contains large values. By performing the normalization using Min-Max Normalization, all the numerical values are compressed in 0 to 1 scale as shown in equation 3.2.

$$NormalizedValue = \frac{X - MIN(X)}{MAX(X) - MIN(X)} \tag{3.2}$$

Implementing min-max normalization equation 3.2 on all the numerical dataset features, the data value that represents the min value in the given feature will have a min-max normalization value of 0 and the value that represents the max value will have min-max normalization value of 1 and all the other values will simply range from 0 to 1.

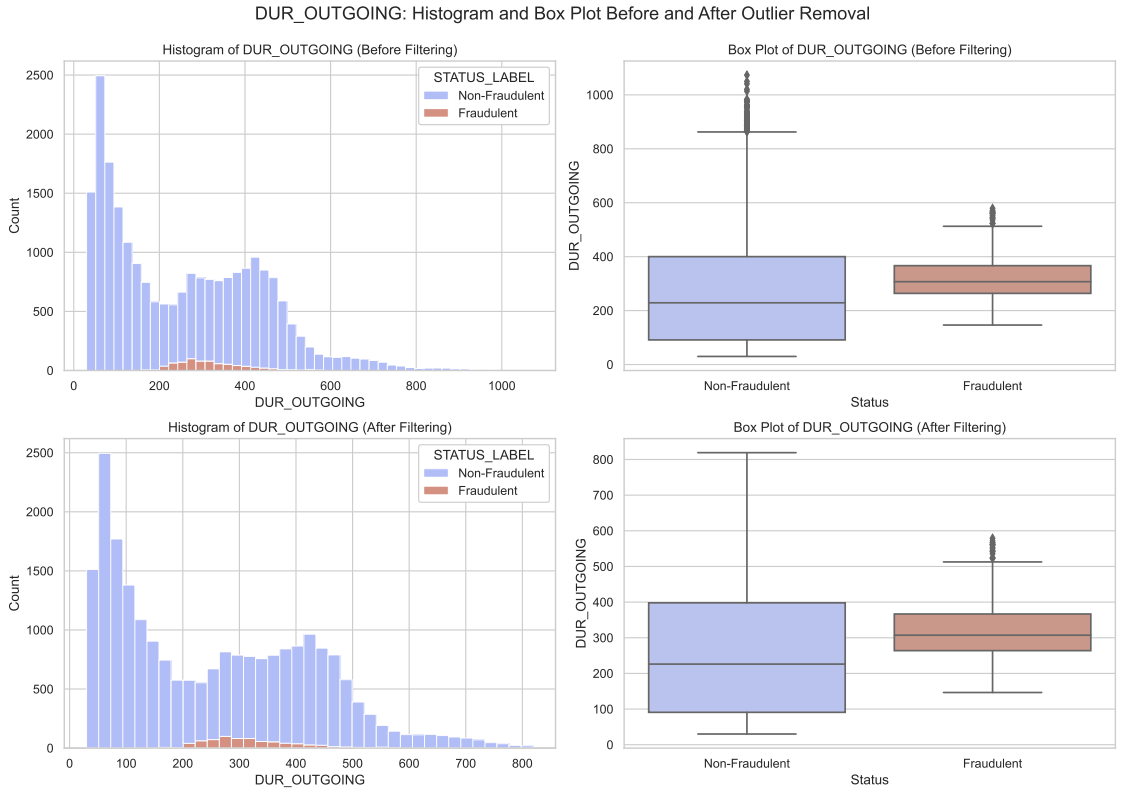


Figure 3.2: Distribution by Outgoing Minutes Before and After Remove Outliers

Table 3.5: Outliers Analysis Results

Feature	Outlier Limit	Description	Action
CALLS_COUNT	More Than 140	1561 Non Fraudulent Observations	Drop Observations
TOTAL_DURATION	More Than 820	123 Non Fraudulent Observations	Drop Observations
COUNT_OUTGOING	More Than 150	750 Non Fraudulent Observations	Drop Observations
DUR_OUTGOING	More Than 820	113 Non Fraudulent Observations	Drop Observations
COUNT_UNIQUE_NUMBERS_CALLED	More Than 120	585 Non Fraudulent Observations	Drop Observations
COUNT_INCOMING	More Than 22	920 Non Fraudulent Observations	Drop Observations
NUMBER_OF_FAILED_CALLS	More Than 200	460 Non Fraudulent Observations, 3 Fraudulent	Drop Observations
FAILED_DIST_CALLED_NUMS	More Than 200	377 Non Fraudulent Observations, 2 Fraudulent	Drop Observations
AGE_ON_DETECTION	More Than 500	11814 Non Fraudulent Observations, 16 Fraudulent	Keep Observations
DETECTION_FREQUENCE	More Than 1.45	803 Non Fraudulent Observations, 454 Fraudulent	Keep Observations

3.5.4 Dummy Variable Coding

For the discrete variables it is required to perform variable coding to be represented using 0 and 1 values. For one discrete feature X that includes 3 different values it will be represented using 0 and 1 by creating 3 different dummy variables with 0 or 1 depending on the original value. In fact, for a given feature with 3 different values, it is enough to have two dummies to represent it correctly, since [0,1] represents the first value, [1,0] represents the second value, and [0,0] represents the third value. So, it is always recommended to eliminate one of the generated dummy variables for better model performance and low complexity.

3.6 Explanatory Data Analysis EDA

For the purpose of investigating calling behaviors and subscriber patterns, detailed explanatory data analysis was performed for the study dataset with the goal of investigating SIMBOX fraud. To Initially check the dataset's shape, understand its scale and assess its structure, a comprehensive data inspection and initial profiling was performed after loading the data from Excel into a Pandas DataFrame figure 3.3, df.info() was used to verify data types, especially ensuring numerical features like AGE_ON_DETECTION and CALLS_COUNT are correctly cast, figure 3.4. And to perform messing data analysis df.isnull().sum() was used to identify any other potential messing values or gaps in the dataset, figure 3.4.

```
fraud_ds = pd.read_excel("onnet_simbox_ds.xlsx", sheet_name='onnet_simbox_ds')
fraud_ds
```

	CASE_ID	PRODUCT	MARKETING_CATEGORY	SUB_REGION	CUSTOMER_SEGMENT	DEALER_NAME	SALES_CHANNEL	STATUS_NAME	STATUS_CODE
0	1	PROD_14035	MKT_3788	REG_8274	CUST_SEG193	DEALER_13046	CHANNEL_2182	Normal	1
1	2	PROD_14035	MKT_3788	REG_8274	CUST_SEG193	DEALER_13046	CHANNEL_2182	Normal	1
2	3	PROD_14035	MKT_3788	REG_8274	CUST_SEG193	DEALER_13046	CHANNEL_2182	Normal	1
3	4	PROD_14035	MKT_3788	REG_8274	CUST_SEG193	DEALER_13046	CHANNEL_2182	Normal	1

Figure 3.3: Dataset's Shape

The second phase was univariate analysis to understand individual features. For numerical variables, descriptive statistics was generated using df.describe() to provide insight into their central tendency, spread, and the presence of outliers, Table 3.3 shows the descriptive statistics for fraudulent cases while, Table 3.4 shows the descriptive statistics for non-fraudulent cases. Histograms and density plots visualizations generated using python seaborn and matplotlib libraries for TOTAL_DURATION and CALLS_COUNT reveal their distribution, highlighting any heavy skewness or multiple peaks that could signal distinct groups of subscribers. Box plots are instrumental in visualizing outliers, which in this context could represent highly unusual and potentially fraudulent calling activity, Figure 3.2 , Table 3.5. Fraudulent cases tend

```
fraud_ds.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 23017 entries, 1921 to 1215
Data columns (total 23 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   CASE_ID                      23017 non-null  int64
 1   PRODUCT                      23017 non-null  object
 2   MARKETING_CATEGORY           23017 non-null  object
 3   SUB_REGION                   23017 non-null  object
 4   CUSTOMER_SEGMENT             23017 non-null  object
 5   DEALER_NAME                  23017 non-null  object
 6   SALES_CHANNEL                23017 non-null  object
 7   STATUS_NAME                  23017 non-null  object
 8   STATUS_CODE                  23017 non-null  int64
 9   CALLS_COUNT                  23017 non-null  int64
 10  TOTAL_DURATION               23017 non-null  float64
 11  COUNT_OUTGOING               23017 non-null  int64
 12  DUR_OUTGOING                 23017 non-null  float64
 13  COUNT_UNIQUE_NUMBERS_CALLED  23017 non-null  int64
 14  COUNT_INCOMING               23017 non-null  int64
 15  DUR_INCOMING                 23017 non-null  float64
 16  COUNT_UNIQUE_CELLS           23017 non-null  int64
 17  NUMBER_OF_FAILED_CALLS       23017 non-null  int64
 18  FAILED_DIST_CALLED_NUMS      23017 non-null  int64
 19  CNT_OUT_VS_DIST_OUT          23017 non-null  float64
 20  AGE_ON_DETECTION             22964 non-null  float64
 21  DETECTION_COUNT              23017 non-null  int64
 22  DETECTION_FREQUENCE          23017 non-null  float64
dtypes: float64(6), int64(10), object(7)
memory usage: 4.7+ MB
```

```
#Check Columns with null values
fraud_ds.isnull().sum()
```

```
CASE_ID                       0
PRODUCT                       0
MARKETING_CATEGORY            0
SUB_REGION                    0
CUSTOMER_SEGMENT              0
DEALER_NAME                   0
SALES_CHANNEL                 0
STATUS_NAME                   0
STATUS_CODE                   0
CALLS_COUNT                   0
TOTAL_DURATION                0
COUNT_OUTGOING                0
DUR_OUTGOING                  0
COUNT_UNIQUE_NUMBERS_CALLED   0
COUNT_INCOMING                0
DUR_INCOMING                  0
COUNT_UNIQUE_CELLS            0
NUMBER_OF_FAILED_CALLS        0
FAILED_DIST_CALLED_NUMS       0
CNT_OUT_VS_DIST_OUT           0
AGE_ON_DETECTION             53
DETECTION_COUNT               0
DETECTION_FREQUENCE           0
```

Figure 3.4: Dataset's Messing Values Check

to have a lower and tighter distribution of call counts. While, Non-Fraudulent cases show a wider spread, indicating more variability and generally higher call volumes. This suggests that fraudulent SIMs may be used in more controlled or limited ways, possibly to avoid detection.The total duration of calls for Fraudulent cases range between 200 and 400, with a more concentrated distribution. While, Non-Fraudulent cases exhibit a broader range, including longer durations.This supports the idea that legitimate users engage in more varied and extended calling behavior.The distribution of CNT_OUT_VS_DIST_OUT, which representing the ratio or comparison between outgoing call count and distinct numbers called, shows a distinct separation,fraudulent cases show greater diversity and have a narrower distribution,indicating non-repetitive calling patterns to many different numbers. Non-Fraudulent cases show lower diversity, suggesting more natural communication behavior, Figure 3.5.

To visually understand the distribution of subscribers across different categories like product, marketing category, and sales channel bar charts have been used to reveal any imbalance or segment-specific characteristics. The generated graphs highlighted several key characteristic's regarding the distribution of fraudulent activity. Analyzing product distribution shows that certain products are heavily associated with fraudulent activity while others are mainly non-fraudulent, which provide an insight of differences in how products are marketed or used. Marketing category distribution also shows that certain marketing categories are heavily associated with fraudulent activity while others are mainly non-fraudulent indicating that fraudsters may exploit specific marketing channels more

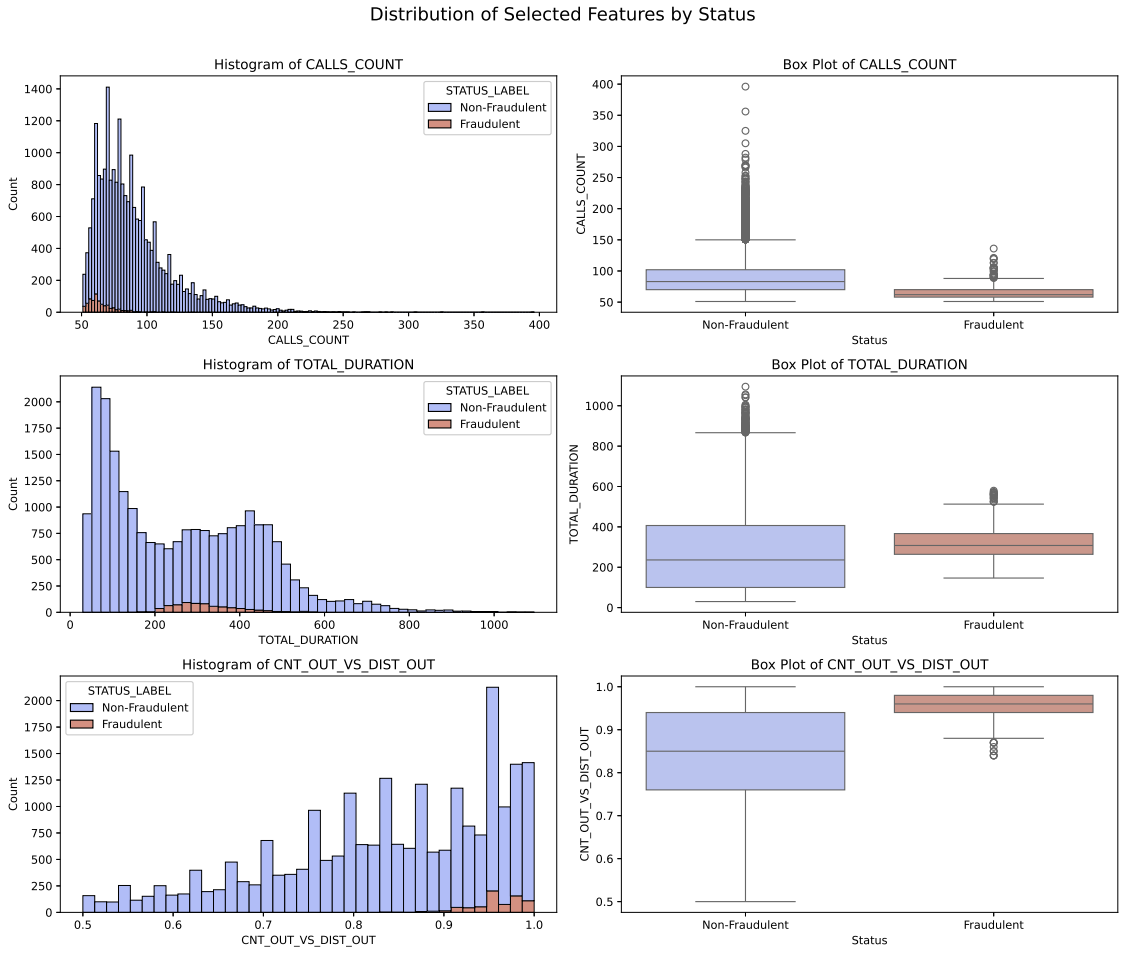


Figure 3.5: Feature Distribution By Status

frequently. Exploring sub-regions distribution showed that fraudsters concentrated only in one region, this might be due to regional vulnerabilities or enforcement gaps. Customer segment distribution shows that certain customer segments are more prone to fraud, which might be due to marketing strategies or usage behavior, while non-fraudulent cases distributed over a broader or more general segments. Sales channels distribution shows that fraudulent cases are more prevalent in specific sales channels signifying channels specific risks. There might be direct or less regulated channels that is more vulnerable to fraud sales. Finally, the analysis of the STATUS_NAME feature an internal operational variable, illustrating the different stages of fraud detection and labeling within the company's process. Figure 3.6.

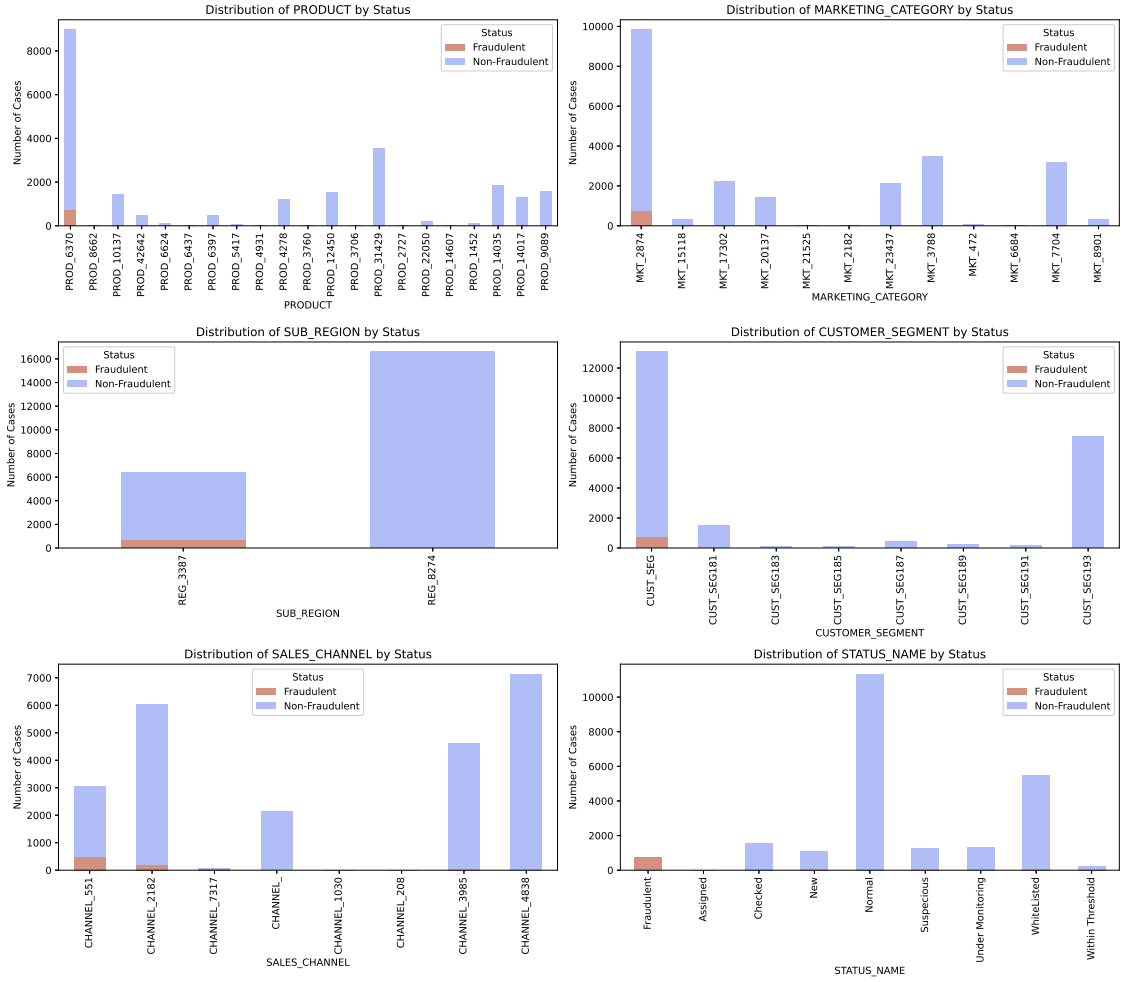


Figure 3.6: By Status Distribution Across Different Categories

To understanding the relationships between multiple variables, bivariate and multivariate analysis has been performed. A correlation heatmap of numerical features visually represent the strength and direction of their relationships, as expected high positive correlations between variables like CALLS_COUNT and COUNT_OUTGOING, COUNT_UNIQUE_NUMBERS_CALLED and COUNT_OUTGOING, while no or negative correlations between variables like

DUR_INCOMING and DUR_OUTGOING,
COUNT_UNIQUE_CELLS and TOTAL_DURATION warrant for further investigation, Figure 3.7. To analyze the interplay between categorical and numerical data, box plots or violin plots was used to compare the distribution of call durations across different categories, fraudulent or non-fraudulent, Figure 3.8.

Violin plots analysis provided valuable insights of the behavioral differences between fraudulent and non-fraudulent cases vs key communication features. Analyzing CALLS_CO
UNT distribution shows that non-fraudulent cases exhibit a wider and higher distribution, indicating more frequent and variable call activity that aligns with normal communication behavior. In contrast, fraudulent cases appear tightly clustered with lower call counts, suggesting limited, restricted, or potentially automated calling patterns rather than natural human interaction. The distribution of TOTAL_DURATION is observed to vary significantly between user classes; non-fraudulent activity is characterized by longer and more diverse call durations, which reflects the variability and engagement typical of human conversation. Conversely, shorter and more consistent durations are exhibited in fraudulent cases. It is suggested that such patterns imply scripted, transactional, or automated behavior, wherein calls are brief and lack natural variation. Further distinctions are identified through the CNT_OUT_VS_DIST_OUT feature distribution, which represents the ratio of outgoing calls to distinct numbers. A narrower and lower distribution is shown by fraudulent users, by which repetitive calling patterns and a high diversity of contacts are suggested. In summary, clear behavioral demarcations between fraudulent and legitimate subscriber activities are revealed through the application of violin plots. While fraudulent behavior is characterized as uniform and consistent in its low volume and high contact diversity, non-fraudulent behavior is found to be more varied, dynamic, and reflective of genuine communication habits.

To visualize the relationships between key numerical pairs, such as TOTAL_DURATION versus COUNT_UNIQUE_NUMBERS_CALLED, scatter plots was created to uncover hidden patterns, trends or unusual clusters, Figure 3.9. The analysis of the figures generated reveals several important relationships between call-related features. For example, A very strong positive relation between CALLS_COUNT and COUNT_OUTGOING, indicating that subscribers who make more calls also tend to have higher outgoing activity which suggests similar behavioral patterns and possibly clear distinctions between fraud and non-fraud clusters. CALLS_COUNT and COUNT_UNIQUE_NUMBERS_CALLED also shows a very strong correlation which indicates that users who make more calls usually contact a greater variety of numbers while fraudulent users might display repetitive patterns. CALLS_COUNT and CNT_OUT_VS_DIST_OUT visualization show a moderate

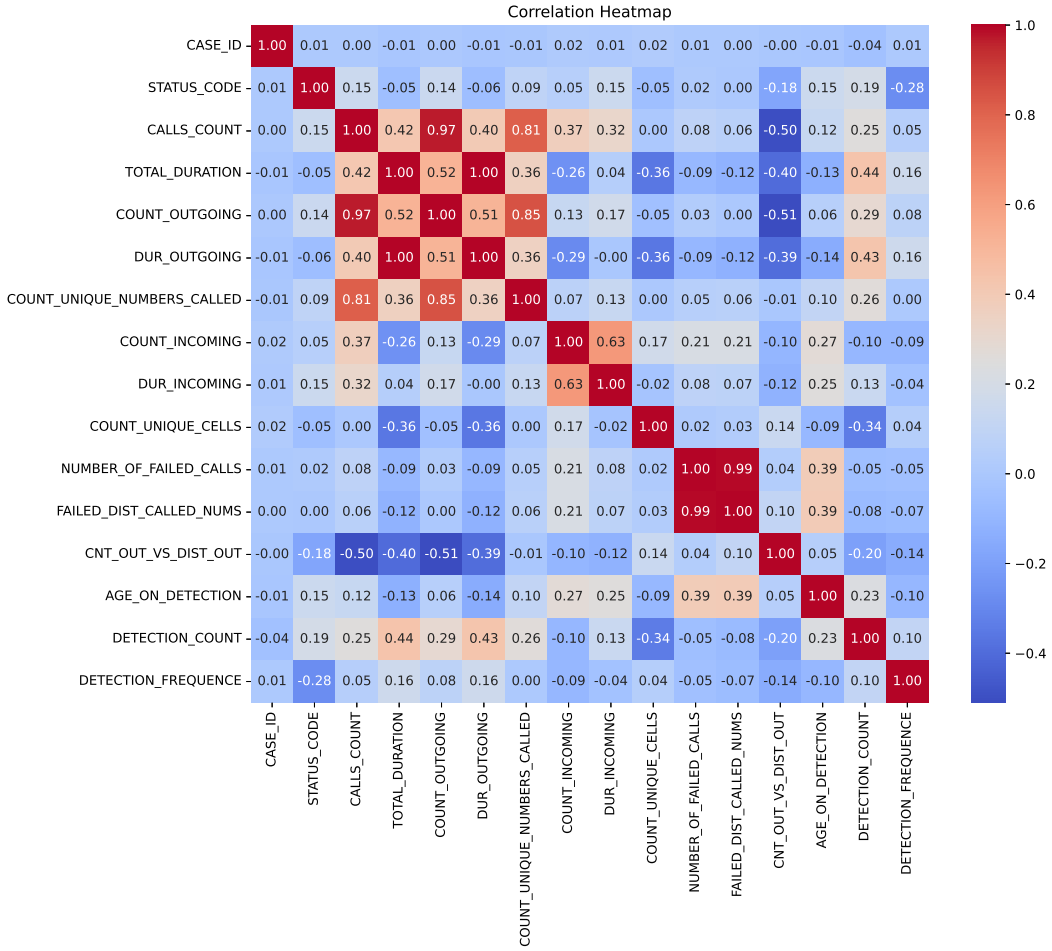


Figure 3.7: Features Heatmap

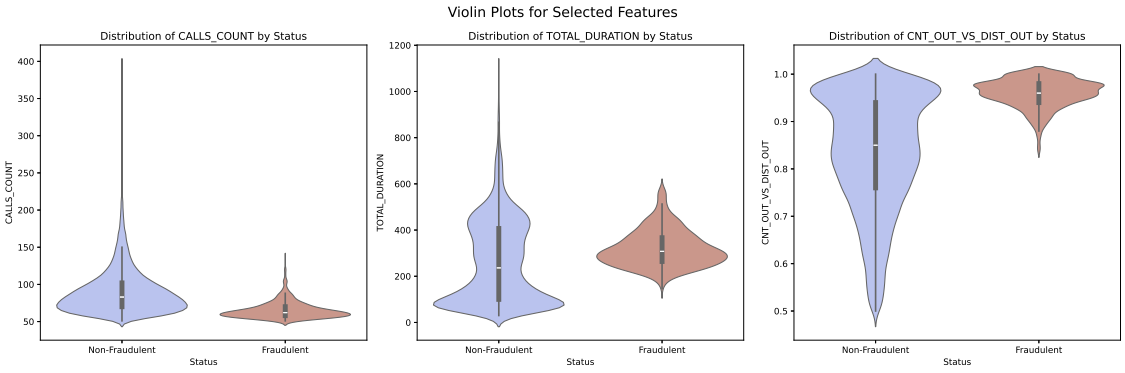


Figure 3.8: Call Count and Call Duration Distribution Across Different Categories

negative relation, as the total number of calls increases, the ratio of outgoing to distinct numbers decreases which could point to repetitive or suspicious calling behavior. Analyzing call duration patterns, TOTAL_DURATION and COUNT_OUTGOING show a moderate positive correlation as the more outgoing calls are generally associated with longer total talk time, typical of legitimate users while for fraudulent it is not necessary to show the same behavior. Furthermore, TOTAL_DURATION and DUR_OUTGOING have a high positive correlation confirming that these two variables are derived from the same metric and ensuring dataset integrity and consistency. Additionally, COUNT_OUTGOING and DUR_OUTGOING shows moderately positive relation, indicating that customers with more outgoing calls tend to accumulate longer call durations, while fraudulent cases might deviate by having many short calls. Another correlation that further supports the idea that frequent callers engage with a wider contact base is the strong positive correlation between COUNT_OUTGOING and COUNT_UNIQUE_NUMBERS_CALLED, while fraudsters likely focus on fewer repeated targets. On the other hand, the visualized correlation between COUNT_OUTGOING and CNT_OUT_VS_DIST_OUT that shows moderate negative correlation indicates that the higher the call counts might reduce the diversity of the dialed numbers, which can be a useful fraud indicator. Analyzing incoming calls mainly COUNT_INCOMING and DUR_INCOMING , shows that users receiving more calls typically have more incoming duration, a pattern that distinguishes genuine communication from automated or fraudulent activity that typically have few incoming calls count and duration. Finally, the representation of the correlation between NUMBER_OF_FAILED_CALLS and FAILED_DIST_CALLED
_NUMS shows a very strong correlation forming another indicator to fraudulent call attempts or possibly linked to technical issues but most probably linked to fraudulent as fraudsters tends to repeatedly try to call destination numbers in case of call failure.

Finally, the main objective of the EDA in this research is to identify anomalies and outliers, which are the main indicators of SIMBOX fraud. Previous EDA results and analysis zoom in on the suspicious behavior that could be used in identifying subscribers with fraudulent behavior. Subscribers with extremely high call counts but low average duration, subscribers with high number of failed call attempts (NUMBER_OF_FAILED_CALLS) indicates a potentially strong predictor of fraudulent calls. Ratios, like CNT_OUT_VS_DIST_OUT when analyzed for unusually high or low values could indicate irregular calling patterns. Visual inspection of outliers identified in box plots and potentially more sophisticated methods for identifying anomalies helps uncovers hidden fraudulent activities. All findings from this comprehensive EDA would then guide the feature engineering process for the development of a robust predictive model to detect SIMBOX fraud.

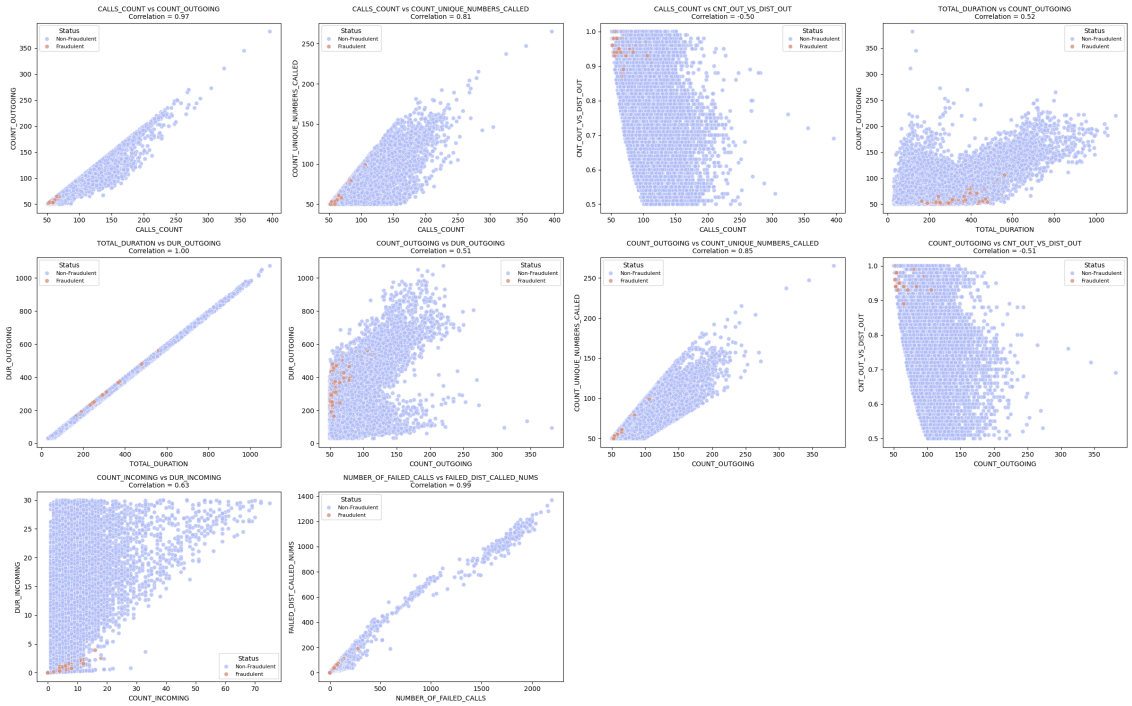


Figure 3.9: Numerical Features with High Correlation

3.7 Data Sampling

In general, the datasets used for studying fraud detection in various business domains are usually imbalanced, the number of fraudulent cases is usually minimal compared with normal non-fraudulent cases. The dataset we used in this study is not an exception to this; it is highly imbalanced, and the fraudulent cases are only 3% of the whole dataset. In our analysis, we processed our dataset implementing over-sampling and under-sampling techniques to balance our dataset in addition to the no-sampling.

- **No Sampling (Baseline):** This approach preserves the original distribution. While it reflects real-world conditions, it often leads to models that are insensitive to minority class patterns.
- **Under-Sampling:** This technique reduces the number of non-fraudulent cases to match the fraudulent ones. It helps balance the dataset but risks losing valuable information from legitimate behavior, which could affect generalization.
- **Over-Sampling:** Synthetically increase the number of fraudulent cases using techniques like SMOTE or random duplication. This enhances the model's ability to learn fraud patterns but may introduce noise or overfitting if not carefully tuned.

Each method was evaluated to understand its impact on model accuracy, precision, recall, and F1-score. The top-left plot, Figure 3.10, shows the original fraudulent data, which serves as the baseline. In this dataset, the fraudulent cases are tightly clustered

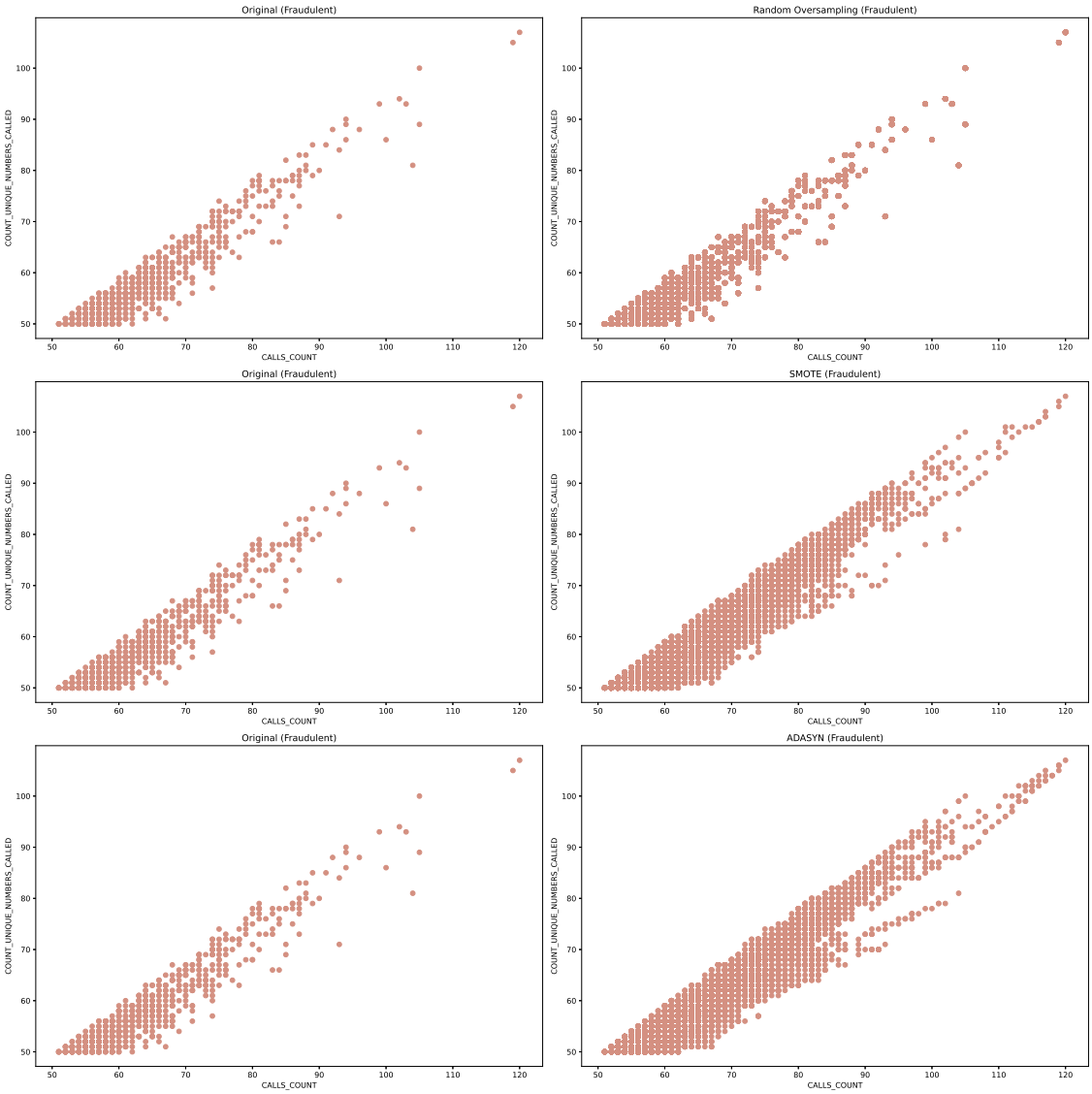


Figure 3.10: Sampling Techniques Effect Comparison

in the lower-left corner, indicating that most fraudsters make few calls and contact few unique numbers. The distribution is quite sparse, making it difficult for a model to learn meaningful patterns.In the top-right plot, we apply random oversampling. This technique simply duplicates existing fraudulent cases to match the number of non-fraudulent ones. While this increases the data density, it doesn't change the overall shape of the distribution. In essence, it's like copying and pasting the same points useful for balancing the dataset but risky because it can lead to overfitting.The middle-right plot illustrates SMOTE (Synthetic Minority Oversampling Technique). SMOTE generates synthetic samples by interpolating between existing data points. This results in a more natural spread of data, filling the gaps between clusters. The model can then learn from a wider variety of patterns without merely relying on duplicated data.
Finally, the bottom-right plot demonstrates ADASYN (Adaptive Synthetic Sampling). This method builds on SMOTE by focusing more on regions where fraudulent cases are underrepresented. As a result, more synthetic points appear in sparse areas, particularly where the original data was limited. This allows the model to better handle challenging, hard-to-classify instances. In summary, random oversampling provides more data but keeps the same distribution shape, SMOTE improves generalization through smooth interpolation, and ADASYN enhances dataset balance by generating targeted synthetic samples from minority class in underrepresented zones.

3.8 Development of RF Model

Random forest is an ensemble of decision trees to minimize the high variance in the model since decision trees have high variance and low bias which makes decision trees unstable Ali et al., 2012. In RF only a subset of the features is selected to split each node in a tree depending on Gini index calculation results. It operates based on the principle of bootstrap aggregation (bagging), where each tree is trained on a random subset of the data, and only a random subset of features is considered for splitting at each node. This randomization reduces overfitting and ensures that individual trees are less correlated, making the overall ensemble more stable and accurate Breiman, 2001. Decision trees alone typically have high variance and low bias, which can make them unstable. RF mitigates this issue by averaging multiple trees, thereby reducing variance without significantly increasing bias.Thanh Noi and Kappas, 2017 In RF, a subset of features is selected at each split based on impurity measures such as the Gini index or entropy. The main hyper-parameters influencing model performance include:

1. Number of trees (n_estimators), the default value is 1000.

2. The Maximum depth each decision tree can grow (max_depth), default value is 100.

3. Minimum observations required at each node to be eligible for splitting (min_samples_split), default value is 3.

4. Minimum number of observations in tree node (leaf node) (min_samples_leaf), the default value is 2.

5. Criterion of slitting (criterion), the default value is Gini, possible values {“entropy”, “log_loss”}.

To further enhance the model’s accuracy and control the bias–variance trade-off, the Grid Search method was employed to optimize these hyper-parameters, as illustrated in Figure 3.11. The optimized RF model was then evaluated using accuracy, precision, recall, F1-score, and AUC to ensure balanced performance across both classes.

Additionally, feature importance analysis was performed to identify the most significant variables contributing to fraud detection Figure 4.3. Despite its computational cost, RF proved to be a robust and reliable classifier for this dataset, capable of handling categorical and imbalanced data while maintaining high predictive stability

3.9 Development of SVM Model

SVM is a discriminatory supervised machine learning classifier that takes training data and outputs an optimal hyperplane that classifies new samples. Figure 3.11 illustrates the implementation of the SVM model, there are three tuning parameters of the SVM model Syarif et al., 2016:

1. Kernel: Kernels are transformation functions used for transforming the original data using a series of operations that can often be computed more efficiently. The kernel is given the original feature vectors and returns the same value as the dot product of its corresponding mapped feature vectors. In SVM, the Kernel function is used to implicitly map the input vector into a high-dimensional feature space. Different SVM algorithms use different types of kernel functions [Linear kernel, Non - linear kernel, Radial basis function (RBF), Sigmoid, Polynomial, Exponential], the most commonly used Kernels are radial basis function (RBF), linear kernel, and polynomial kernel.

2. Regularization or C parameter: Used to optimize the SVM classifier to avoid misclassification of data points, it controls the perpendicular margin distance between the closest data points and the Hyperplane from both sides, large value of C parameter decreases the margin distance, and increases the chance of overfitting while the small value of C parameter increase the margin distance and increase the chance of underfitting.

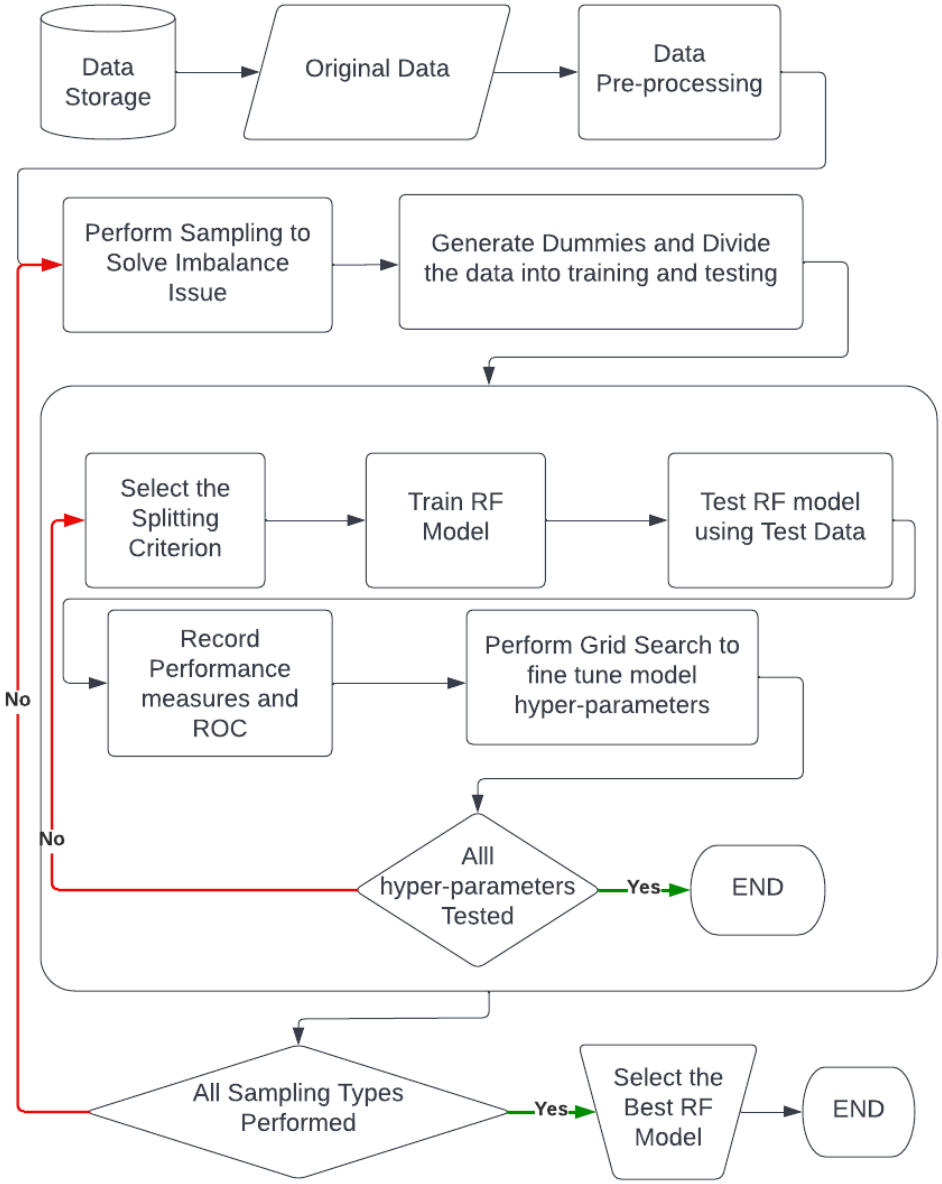


Figure 3.11: Flow of RF implementation

3. Gamma: Defines how far the data points that influence the calculation of the hyper-plane, using low value for the Gamma parameter data points far from the hyperplane will be considered for calculation while using a high value only close points will be considered for calculation. Therefore, very high gamma values tend to increase the chance of overfitting.

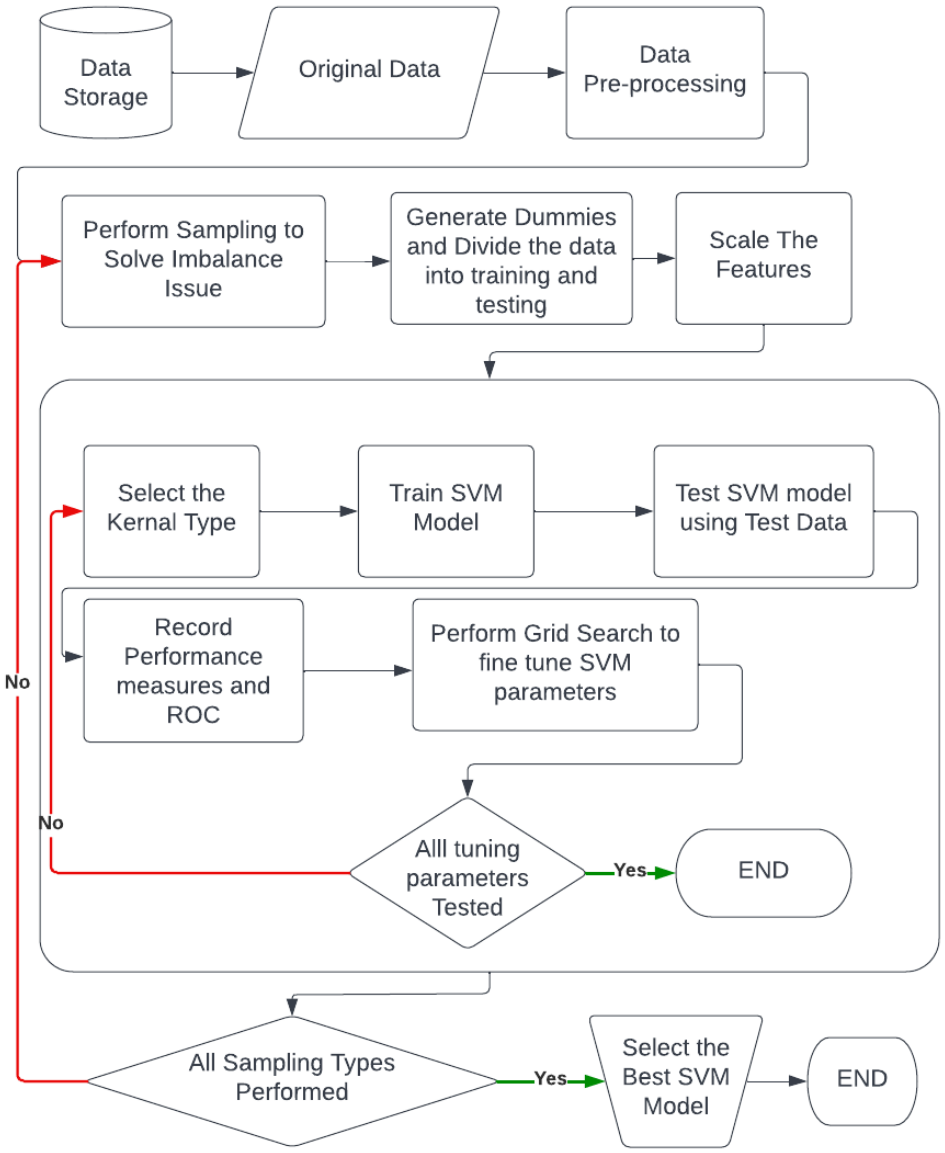


Figure 3.12: Flow of SVM implementation

To optimize these parameters to get the best performance parameters combinations, 5-fold cross-validation was applied to the dataset and all three kernels has been tested and the performance of each model has been recorded.

3.10 Summary

The methodology adopted in this study was designed to construct a robust SIMBOX fraud detection model using a real telecommunications dataset. The dataset used consists of 23,017 cases, of which 22,289 are normal and 728 are fraudulent, identified through expert labeling and traditional fraud detection systems. Each case is represented by a unique identifier and enriched with subscriber-level attributes, including product type, marketing category, sub-region, and customer segment, as well as acquisition details such as dealer name and sales channel. In addition, the dataset incorporates detailed usage metrics such as call counts, durations of incoming and outgoing calls, the number of unique cells accessed, and failed call statistics. Features such as age on detection, detection frequency, and case status codes that represent the case history have been included to provide a comprehensive view of subscriber behavior and enhance the machine learning model ability to distinguish fraudulent from legitimate cases. Starting with data extraction from raw Event Data Records (EDRs) stored in oracle databases, the dataset was extracted from the EDRs repository after being enriched with subscriber and product attributes and exported into Excel for accessibility. Using Python's pandas library, the dataset was loaded into a DataFrame for analysis, analysis included handling missing values, removing outliers, normalization, and feature engineering. Based on prior literature and expert knowledge twenty features were selected including derived features such as "Call Duration per Unique Number" and "Unique Called Numbers Ratio" engineered to capture fraud-specific behaviors. Data loading and preprocessing showed that missing values were minimal and were imputed using mean values. Descriptive statistics, visualizations, and Z-score standardization used for identifying and eliminating outliers but with careful consideration given to retaining fraudulent cases that appeared as statistical anomalies to avoid increasing the class imbalance. To ensure fair processing by algorithms sensitive to magnitude differences such as Support Vector Machine numerical features were scaled using Min-Max normalization to compress values into a [0,1] range, while categorical variables were encoded into dummy variables, with one dummy per feature eliminated to reduce redundancy and improve efficiency. Exploratory Data Analysis (EDA) was conducted in three phases, the first phase involved inspecting dataset shape, data types, missing values and performing descriptive statistics and histograms visualization to check for skewness and potential anomalies. The second phase involved performing univariate analysis for the individual features which showed that fraudulent cases tended to have lower call counts, shorter durations, and more diverse outgoing numbers, while non-fraudulent cases showed higher variability and longer calls. Categorical features showed that fraud cases concentrate in specific products, regions, and sales channels, highlighting areas of vulnerability. The third phase is the bivariate and multivariate analysis. During this phase the relationships between variables through correlation heatmaps, violin plots, and scatter plots has been

explored and analyzed. Analysis results showed strong positive correlations between call count and outgoing calls (r = 0.97), a uniform calling pattern for fraudulent users compared to the diverse and natural communication behaviors of legitimate users. Suspicious clusters observed in scatter plots, such as high call counts with short durations, and failed call metrics showed near-perfect correlation (r = 0.99), serving as strong fraud indicators. These analyses provided a valuable insight into the behavior of the subscribers, such insights that help in the process of distinctions between fraudulent and non-fraudulent subscribers and guided feature engineering and model development. The final stage of the methodology involved addressing class imbalance, oversampling and undersampling techniques has been used, followed by splitting the dataset into training and testing subsets for the purpose of training and testing SVM and RF models. Generated models performance has been evaluated through comparative analysis of accuracy and robustness. Overall, the study methodology integrates data engineering, statistical analysis, and machine learning to construct a comprehensive fraud detection model combining SVM and RF classifiers with balanced and imbalanced datasets. The study identifies optimal approaches for detecting SIMBOX fraud.

Chapter Four: Results and Discussion

The dataset has been analyzed using Random Forest and SVM implementations. The dataset is highly imbalanced, so we have performed the analysis using balanced and imbalanced datasets by using oversampling and undersampling techniques to solve the imbalance issue. This chapter presents detailed results of implementing Random Forest and SVM models for SIMBOX fraud detection. Discusses tuning, sampling and the models accuracy before and after implementing sampling techniques. It addresses also the different evaluation creteria used and concludes by noting limitations such as data imbalance, privacy concerns, and evolving fraud patterns, while emphasizing the strong potential of ML models in telecom fraud detection.

4.1 RF Implementation Results

This section presents in detail the results of implementing the Random Forest model using our dataset, and the following points address the main model outcomes:

1. Results using an imbalanced dataset show high model accuracy 99.5% and precision as shown in table 4.1 below.

   Table 4.1: RF Performance Measures - Imbalanced Dataset

Test Accuracy:	99.5%	The Percent of cases predicted correctly (fraudulent and non-fraudulent)
Precision (P):	87.8%	Among the predicted fraudulent what % is predicted correctly
Recall (R)/sensitivity/TPR:	97.1%	Among the actual fraudulent what fraction was predicted correctly
Specificity:	99.6%	Among the actual non-fraudulent what fraction was predicted correctly
FPR:	0.4%	Among the actual non-fraudulent what fraction was predicted wrongly

2. Using Grid Search, the best performance has been scored using the below tuned parameters:

   (a) Number of trees is 1000

   (b) Criterion of slitting is Gini [Gini, entropy, and log_loss has been tested]

   (c) Maximum depth each decision tree can grow is 70

   (d) Minimum observations required at each node to be eligible for splitting is 2.

   (e) Minimum number of observations in tree node should be 1

   Results after Using the optimal Parameters, the accuracy remains the same but the precision increased by around 1% to 88%.

3. Implementing under-sampling on our dataset the model performance remains the same which is already high, but the precision increased significantly from 87% to 98% which is an excellent improvement, while the FPR increased slightly by 1.5%. While, with over sampling the model performance remains the same, but the precision increased by 1.1% while the FPR decreased from 1.9% to 0.7% which is an excellent additional improvement, Table 4.2 illustrates the comparative results obtained before and after sampling, Figure 4.1 and Figure 4.2 show the ROC curve and confusion matrix for each model.

Table 4.2: RF Performance Measures - Effect of Sampling on Model Performance

Without Sampling		With OverSampling		With UnderSampling	
Test Accuracy:	99.5%	Test Accuracy:	99.7%	Test Accuracy:	99.1%
Precision (P):	87.8%	Precision (P):	99.4%	Precision (P):	98.2%
Recall (R)/sensitivity/TPR:	97.1%	Recall (R)/sensitivity/TPR:	100.0%	Recall (R)/sensitivity/TPR:	100.0%
Specificity:	99.6%	Specificity:	99.4%	Specificity:	98.1%
FPR:	0.4%	FPR:	0.6%	FPR:	1.9%

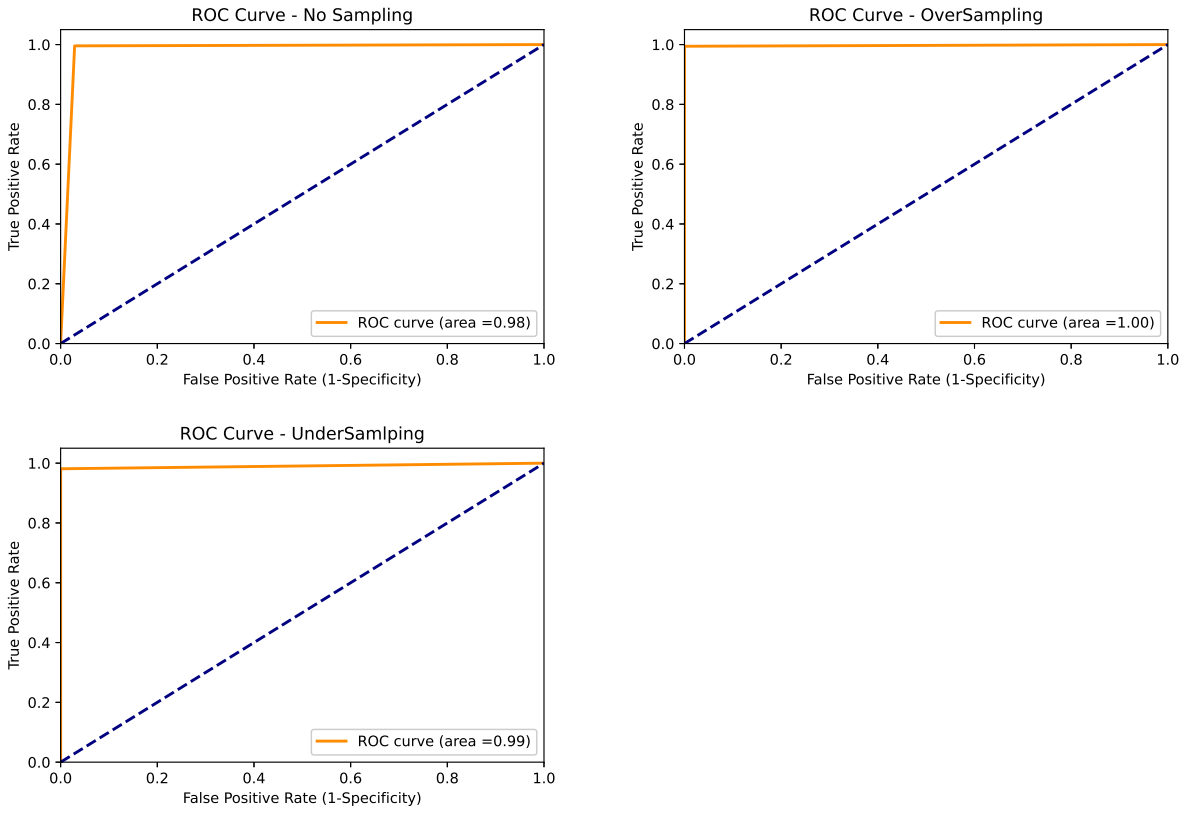


Figure 4.1: RF ROC - Effect of Sampling on Model Performance

4. Figure 4.3 illustrates the features importance using RF, the features that highly contribute in the final model decision have the highest importance like DETECTION_FREQUENCE, DETECTION_COUNT and DEALER_NAME

Without Sampling			With OverSampling			With UnderSampling		
	Predicted			Predicted			Predicted	
Actual	Fraud	Normal	Actual	Fraud	Normal	Actual	Fraud	Normal
Fraud	201	6	Fraud	5983	0	Fraud	224	0
Normal	28	6671	Normal	40	5893	Normal	4	209

Figure 4.2: RF Confusion Matrix - Sampled vs Non-Sampled

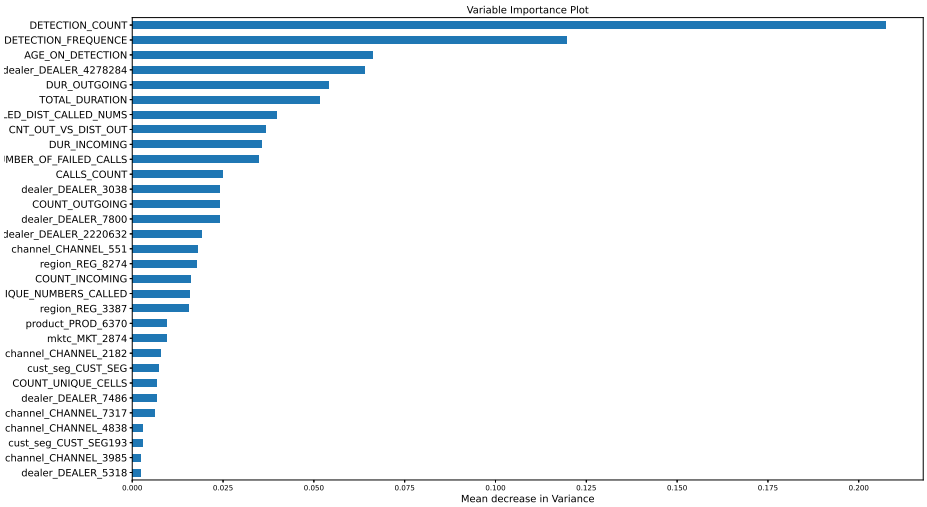


Figure 4.3: RF Features Importance

4.2 SVM Implementation Results

This section address in detail the SVM implementation results summarized below:

1. Using SVM it is recommended to scale the features and it is very important when using RBF Kernel.

2. The results of performing a grid search to tune the model parameters show that the best performance parameter combination is using RBF kernel with Gamma=0.25 and C=50, Polynomial Kernel with Gamma=0.2 and C=50. Table 4.3 illustrates the various performance measures using each kernel. The high performance and low precision are due to the imbalance issue in the dataset, so the analysis has been performed using both oversampling and under-sampling techniques since such results are not actual and do not reflect the actual model performance.

Table 4.3: SVM Performance Measures - Different Kernels

Measure	Linear Kernel	RBF Kernel	Polynomial Kernel
Test Accuracy:	99.1%	99.3%	99.3%
Precision (P):	79.8%	83.6%	83.7%
Recall (R)/sensitivity/TPR:	99.5%	99.0%	99.5%
Specificity:	99.1%	99.3%	99.3%
FPR:	0.9%	0.7%	0.7%

3. RBF kernel with Gamma=0.25 and C=50, Polynomial Kernel with Gamma=0.2 and C=50 are considered the optimal parameters for the SVM model despite that the model overall accuracy does not differ too much due to the fact that they produce the highest prediction accuracy among the fraudulent cases which is the main target of the model. Among the predicted fraudulent cases 84% is predicted correctly, while using Linear kernel it is less then 80%.

4. Trying to tune the parameters using Grid Search required high execution time and resources without scaling the features. Using a polynomial kernel the execution lasts for more than 24 hours and failed while using Linear and RBF kernels execution takes 2 hours fitting 5 folds for 36 candidates, a total of 180 fits. However, after scaling the features the required execution time decreased significantly to less than 10 minutes for each kernel.

5. To Solve the imbalance issue in the dataset and measure the effect on model performance, oversampling and undersampling have been implemented and SVM has been trained and tested on both oversampled and undersampled datasets. Table 4.4

illustrates the effect of sampling on model performance using different SVM kernels using the optimum parameters. It clearly shows that despite the fact that under-sampling increases FPR by 1.2%, both over and under-sampling have a significant effect on enhancing the model precision from 84% max using RBF and polynomial kernels to 99.4%.

Table 4.4: SVM Performance Measures - Different Kernels Sampled vs Non-Sampled

Linear Kernel	NoSampling	UnderSampling	OverSampling
Test Accuracy:	99.1%	99.1%	99.5%
Precision (P):	79.8%	98.2%	99.0%
Recall (R)/sensitivity/TPR:	99.5%	100.0%	100.0%
Specificity:	99.1%	98.1%	98.9%
FPR:	0.9%	1.9%	1.1%
RBF Kernel	NoSampling	UnderSampling	OverSampling
Test Accuracy:	99.3%	99.1%	99.7%
Precision (P):	83.6%	98.2%	99.4%
Recall (R)/sensitivity/TPR:	99.0%	100.0%	100.0%
Specificity:	99.3%	98.1%	99.4%
FPR:	0.7%	1.9%	0.6%
Polynomial Kernel	NoSampling	UnderSampling	OverSampling
Test Accuracy:	99.3%	99.1%	99.7%
Precision (P):	83.7%	98.2%	99.4%
Recall (R)/sensitivity/TPR:	99.5%	100.0%	100.0%
Specificity:	99.3%	98.1%	99.4%
FPR:	0.7%	1.9%	0.6%

4.3 Models Performance Evaluation

To compare the accuracy of RF and SVM models, two comparison criteria were chosen which are 1) FNR, FPR, and RMSE Evaluation to measure the classification accuracy, and 2) model building duration evaluation to evaluate the time needed for model building and tuning. In the first evaluation, a comparison between FNR and FPR for both RF and SVM best models was conducted.

	Linear Kernel			RBF Kernel			Polynomial Kernel		
		Predicted			Predicted			Predicted	
No Sampling	**Actual**	Fraud	Normal	**Actual**	Fraud	Normal	**Actual**	Fraud	Normal
	Fraud	205	1	Fraud	204	2	Fraud	205	1
	Normal	52	5918	Normal	40	5930	Normal	40	5930

		Predicted			Predicted			Predicted	
UnderSampling	**Actual**	Fraud	Normal	Actual	Fraud	Normal	**Actual**	Fraud	Normal
	Fraud	223	0	Fraud	223	0	Fraud	223	0
	Normal	4	209	Normal	4	209	Normal	4	209

		Predicted			Predicted			Predicted	
OverSampling	**Actual**	0	Normal	**Actual**	0	Normal	**Actual**	Fraud	Normal
	Fraud	5969	0	Fraud	5969	0	Fraud	5969	0
	Normal	63	5884	Normal	35	5912	Normal	37	5910

Figure 4.4: SVM Confusion Matrix - Different Kernels Sampled vs Non-Sampled

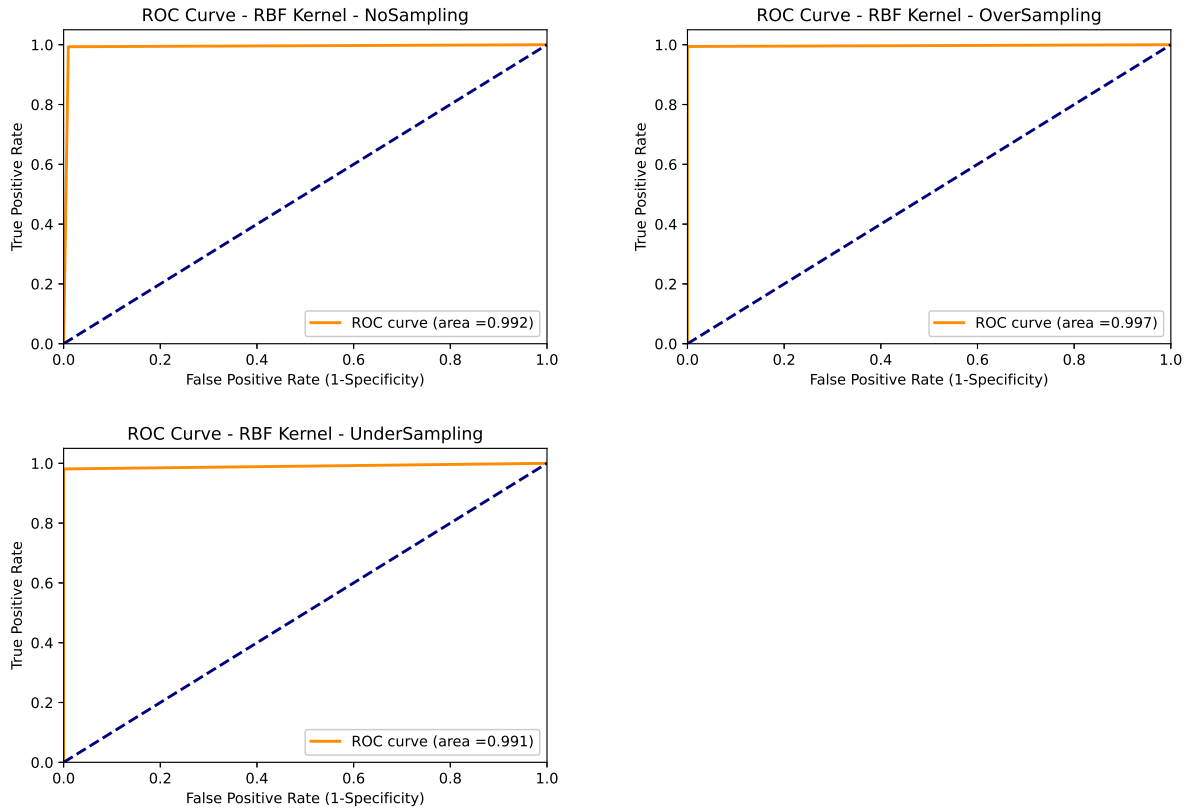


Figure 4.5: SVM ROC - Effect of Sampling on Model Performance

4.4 RF and SVM FNR, FPR and RMSE Evaluation

Comparing the performance of RF and SVM models in terms of FNR, FPR, and RMSE to identify which model has the best performance in identifying the normal and fraudulent cases. FPR represents the number of normal cases that are wrongly classified as fraudulent, while FNR represents the number of fraudulent subscribers that are wrongly classified as normal. Figure 4.6 shows the confusion matrix of both RF and SVM best models, the rows represent the actual values while the columns represent the predicted values.

RF			SVM		
	Predicted			Predicted	
Actual	Fraud	Normal	**Actual**	Fraud	Normal
Fraud	5983	0	Fraud	5969	0
Normal	40	5893	Normal	35	5912

Figure 4.6: RF vs SVM Confusion Matrix

In summary, RF has an RMSE of 0.0579, the model was able to correctly classify 5,983 out of 6,023 fraudulent cases and 5,893 out of 5,893 normal cases, while it failed to predict correctly 40 cases classified as fraud while they are actually normal. SVM has an RMSE of 0.0542, the model was able to correctly classify 5,969 out of 6,004 fraudulent cases and 5,912 out of 5,912 normal cases, while it failed to predict correctly 35 cases classified as fraud while they are actually normal, which means that it outperforms RF by 5 correct cases predicted correctly. Implementing feature scaling for RF has decreased the wrongly predicted cases from 40 to 36 which is almost the same performance as the SVM model. Table 4.5 illustrates the comparative results obtained from both models.

Table 4.5: Comparison of selected RF and SVM model

Features	RF Best Model	SVM Best Model
RMSE	0.0579	0.0542
Accuracy	99.66%	99.71%
Time - Sec.	52	20h
ROC Area	1.00	1.000
Precision	0.993	0.994
Recall/TPR	1.000	1.000
Tuning Time - Sec.	1,740	676

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y - \hat{y})^2} \tag{4.1}$$

$$FPR = \frac{FP}{TN + FP} \tag{4.2}$$

$$FNR = \frac{FN}{FN + TP} \tag{4.3}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.4}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.5}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.6}$$

4.5 Model Building Duration Evaluation

Model building and training duration are important criteria for evaluating and selecting the best model for a given task. Model building duration is the required time for the training and testing process to finish. The model-building duration of RF and SVM mode was compared in Table 4.5. From Table 4.5 it can be seen that SVM takes less time in model training and testing. It can be clearly seen that RF time is more than two times higher than the time taken by the SVM model in building and testing the model when 70% of the training dataset was used to train the model and 30% to test, it has taken only 20 seconds for SVM, while it has taken 51 seconds for RF model which is more than two times the duration taken by SVM. SVM also takes less time in tuning the model parameters using grid search to find the optimal parameter combinations that produce the best prediction results. As shown in Table 4.5, the time required for the RF model for fitting 5 folds for 36 candidates, a total of 180 fits is around three times higher than the SVM model, RF required 29 minutes while SVM required 11 minutes.

4.6 Study limitations

While valuable insights were gained, this research faced limitations inherent to the field, including securing data access from telecom providers and the complex nature of real-world datasets. Issues such as data quality, class imbalance, and the evolving nature of fraud over time affected the scope and potential for generalization. These factors inform the interpretation of the results and point to areas for future investigation.

Data Privacy and Security: Telecom datasets often contain sensitive customer information. Ensuring data privacy and security while collecting and handling such data can be a significant challenge.

Data Availability: Access to a comprehensive and representative telecom dataset that includes instances of SIMBOX fraud can be limited. Telecom companies may be hesitant to share their data due to security concerns, making it challenging to obtain enough relevant data.

Data Quality: Telecom data can be noisy and contain errors. Inaccurate or incomplete data can lead to biased results and affect the performance of machine learning models. Cleaning and preprocessing the data can be time-consuming and resource intensive.

Class Imbalance: Fraudulent activities like SIMBOX fraud are often rare events compared to legitimate transactions. Class imbalance in the dataset can make it difficult to train accurate machine learning models. Data sampling techniques, as mentioned in our study, are used to address this issue.

Data Labeling: Accurately labeling instances of SIMBOX fraud in the dataset can be challenging. Fraudulent activities might go undetected, leading to mislabeled data. This can impact the model's ability to learn and generalize.

Temporal Changes: Telecom fraud patterns can change over time. Data collected during one period may not be representative of future fraud activities. Continuous data collection and model updating may be required to adapt to evolving fraud tactics.

Collaboration with Telecom Providers: Establishing collaboration with telecom providers to access their data can be a complex and time-consuming process. Negotiating data-sharing agreements and obtaining the necessary permissions can be a barrier.

Limited Feature Availability: Telecom datasets may not always include all the relevant features needed for fraud detection. Additional data sources or feature engineering may be necessary, adding complexity to the data collection process.

4.7 Summary

In this chapter the results of Random Forest (RF) and Support Vector Machine (SVM) models implementation for SIMBOX fraud detection using real telecom dataset was presented. As the dataset is highly imbalanced, fraudulent cases representing only a small portion of the total cases, this study analyses were conducted on both imbalanced and balanced versions. To balance the dataset several oversampling and undersampling techniques were applied to address class imbalance and evaluate their effect on model performance. Results showed that on the imbalanced dataset Random Forest model initially achieved very high performance, with accuracy of 99.5%, precision of 87.8%, recall of 97.1%, and specificity of 99.6%. To further optimize the model performance, specially

the precision grid search optimization has been used. Optimization showed that best parameters as 1000 trees, gini splitting criterion, maximum depth of 70, and minimum node size of one. With these parameters, accuracy remained stable, but precision improved slightly to 88%. Using sampling techniques precision increased to 98.2% but at the cost of a slightly higher false positive rate when using undersampling. When using oversampling precision of 99.4% was achieved and false positive rate decreased to 0.6%. These results prove that oversampling could significantly enhance the model's ability to correctly identify fraudulent cases and enhance model overall performance. Analysis of feature importance revealed that features like detection frequency, detection count, and dealer name were the most important variables in detecting SIMBOX fraud. Also, SVM model results show strong performance that varies across kernels. Grid search tuning has been used to identify the optimal parameters as RBF kernel with Gamma = 0.25 and C = 50, and a polynomial kernel with Gamma = 0.2 and C = 50. Using imbalanced dataset without sampling, SVM achieved an accuracy of 99.3% with precision around 83–84%, after applying oversampling and undersampling, precision improved dramatically to 99.4%, with recall reaching 100% across different kernels. Execution time was a notable consideration, results show that SVM takes much less time in model training and testing compared to RF as RF time is more than two times higher than the time taken by the SVM model in building and testing the model. RF and SVM comparative analysis in terms of false negative rate (FNR), false positive rate (FPR), root mean square error (RMSE), and model building duration, both models achieved near-perfect classification, but SVM slightly outperformed RF in terms of RMSE (0.0542 vs.0.0579) and correctly classified five additional fraudulent cases, and in terms of computational efficiency, SVM required less time for both training and parameter tuning. So, as findings indicate that both models are highly effective, SVM offers marginally better predictive performance and efficiency. In summary, while RF achieved high accuracy, SVM offered slightly superior precision, recall, and efficiency, making it the more effective model overall. The results highlight the importance of preprocessing, feature selection, and sampling in fraud detection, while also acknowledging the practical challenges of data access, quality, and evolving fraud behaviors. These findings contribute to the advancement of machine learning applications in telecommunications fraud detection and highlight avenues for future research and collaboration.

Chapter Five: Conclusion and Recommendations

5.1 Summary

The aim of this research work was to identify the most significant features that can be used to effectively identify fraudulent subscribers. Subscribers who use SIMs to originate illegal fraud calls from SIMBOX devices to bypass international calls. Twenty meaningful features were extracted from subscriber EDRs and activation data, reflecting usage patterns such as call frequency, destination diversity and activation trends.To develop a classification model that can predict such cases with high accuracy and less computational power. Accuracy before and after applying oversampling and under-sampling techniques to handle the imbalanced dataset was evaluated. Two algorithms, Random Forest (RF) and Support Vector Machine (SVM) learning potential for the detection of SIM box fraud were tested and investigated.

5.2 Conclusion

Random Forest (RF) and Support Vector Machine (SVM) were selected for experimentation due to their proven capabilities in classification tasks. Both models were tested under two conditions: with raw imbalanced dataset and with balances dataset after applying sampling techniques to mitigate class imbalance. The dataset used in this study comprised subscriber event detail records (EDRs) and activation information, from which twenty representative features were extracted. These features extracted from raw subscribers' EDRs and activation information were found to be useful for identifying SIM box fraud. The selected features as shown in table 3.1. The dataset includes features represent the usage behavior of the subscribers such as call frequency, destination diversity, and activation trends, which are indicative of SIMBOX activity, in addition to Customer-centric and Service acquisition features.

All features have been explored, and 5-fold cross-validation has been used to test and evaluate the performance of both developed models and to ensure reliable performance estimates and reduce overfitting risk. The best model based on accuracy, execution time and precision is selected from 180 RF and SVM-developed models. The best RF and SVM models selected are then compared by applying 70% training and 30% testing datasets providing a clear comparison of their predictive capabilities under realistic conditions. Experimental results show that oversampling is the best approach to solve the imbalance issue in both RF and SVM implementations; by increasing the representation of fraudulent cases, oversampling enabled the models to learn more discriminative patterns without sacrificing overall accuracy. SVM accuracy is slightly better than RF, SVM gave an accuracy

of 99.71% while RF gave 99.66%, SVM has a lower (FPR), which is critical for reducing unnecessary investigations and minimizing operational costs. Also, SVM takes three times less time than RF for model tuning, training, and testing, making it more suitable for real-time or large-scale deployment. Therefore, in terms of accuracy and performance SVM with the oversampling approach is considered the more appropriate for building a prediction model for SIM BOX fraud detection. Its superior accuracy, reduced FPR, and faster execution time make it the preferred choice for telecom operators seeking a scalable and reliable solution.

5.3 Future Work

For future work, it is recommended to further implement other scaling and sampling techniques to further decrease the number of wrongly classified cases in both algorithms. Periodically retrain and monitor models to address evolving fraud patterns and monitoring for concept drift to maintain accuracy as fraud tactics adapt over time. Investigate the wrongly classified cases and trace how the algorithm came to a result. Deploy the model to be used online and offline to detect SIMBOX fraud.

References

Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications*, *68*, 90–113.

Aburbeian, A. M., & Ashqar, H. I. (2023). Credit card fraud detection using enhanced random forest classifier for imbalanced data. *International conference on advances in computing research*, 605–616.

Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012). Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, *9*(5), 272.

Alshawi, B. (2024). Comparison of svm kernels in credit card fraud detection using gans. *International Journal of Advanced Computer Science & Applications*, *15*(1).

Awe, O. O., Ojumu, J. B., Ayanwoye, G. A., Ojumoola, J. S., & Dias, R. (2024). Machine learning approaches for handling imbalances in health data classification. In *Sustainable statistical and data science methods and practices: Reports from lisa 2020 global network, ghana, 2022* (pp. 375–391). Springer.

Azad, M. A., & Morla, R. (2012). Mitigating spit with social strength. *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 1393–1398.

Babaei, K., Chen, Z., & Maul, T. (2019). A study of fraud types, challenges and detection approaches in telecommunication. *Journal of Information Systems and Telecommunication, October-December*, *7*(4), 248261.

Becker, R. A., Volinsky, C., & Wilks, A. R. (2010). Fraud detection in telecommunications: History and lessons learned. *Technometrics*, *52*(1), 20–33.

Bello, O. A., Ogundipe, A., Mohammed, D., Adebola, F., Alonge, O. A., et al. (2023). Ai-driven approaches for real-time fraud detection in us financial transactions: Challenges and opportunities. *European Journal of Computer Science and Information Technology*, *11*(6), 84–102.

Bouguettoucha, A. R. (2024). Simbox fraud detection in telecommunication operators: A case study on" djezzy".

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Chang, Y.-C., Lai, K.-T., Chou, S.-C. T., & Chen, M.-S. (2017). Mining the networks of telecommunication fraud groups using social network analysis. *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, 1128–1131.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321–357.

Chu, G., Wang, J., Qi, Q., Sun, H., Tao, S., Yang, H., Liao, J., & Han, Z. (2022). Exploiting spatial-temporal behavior patterns for fraud detection in telecom networks. *IEEE Transactions on Dependable and Secure Computing*.

COM, F. (n.d.). Fico falcon fraud manage [Accessed: 2025-09-18].

COM, S. (n.d.). Sas fraud management [Accessed: 2025-09-18].

Daka, J. C., & Nyirenda, M. (2022). Smart mobile telecommunication network fraud detection system using call traffic pattern analysis and artificial neural network. *American Journal of Intelligent Systems*, *12*(2), 43–50.

Dubey, R., Zhou, J., Wang, Y., Thompson, P. M., Ye, J., Initiative, A. D. N., et al. (2014). Analysis of sampling techniques for imbalanced data: An n= 648 adni study. *NeuroImage*, *87*, 220–241.

Elmi, A. H., Ibrahim, S., & Sallehuddin, R. (2012). Detecting sim box fraud using neural network. In *It convergence and security 2012* (pp. 575–582). Springer.

Erastov, V., & Balytska, M. (2025). Artificial intelligence in tax fraud detection and prevention. *Modern Tools for Fraud Detection: Insights from the V4 and Ukraine*, 8.

Estévez, P. A., Held, C. M., & Perez, C. A. (2006). Subscription fraud prevention in telecommunications using fuzzy rules and neural networks. *Expert Systems with Applications*, *31*(2), 337–344.

Geepalla, E., Abuhamoud, N., & Abouda, A. (2018). Analysis of call detail records for understanding users behavior and anomaly detection using neo4j. *5th International Symposium on Data Mining Applications*, 74–83.

Ghorbani, R., & Ghousi, R. (2020). Comparing different resampling methods in predicting students' performance using machine learning techniques. *IEEE Access*, *8*, 67899–67911.

Ha, T. M., & Bunke, H. (1997). Off-line, handwritten numeral recognition by perturbation method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(5), 535–539.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, 1322–1328.

Howell, A. (2021, May). Telecom fraud on the rise: 2021 cfca global telecommunications fraud loss survey.

Jiang, N., Jin, Y., Skudlark, A., & Zhang, Z.-L. (2013). Greystar: Fast and accurate detection of {sms} spam numbers in large cellular networks using gray phone space. *22nd USENIX Security Symposium (USENIX Security 13)*, 1–16.

Kashir, M., & Bashir, S. (2019). Machine learning techniques for sim box fraud detection. *2019 International Conference on Communication Technologies (ComTech)*, 4–8.

Kilinc, H. H. (2022). Anomaly pattern analysis based on machine learning on real telecommunication data. *2022 7th International Conference on Computer Science and Engineering (UBMK)*, 43–48.

Kouam, A. J., Viana, A. C., & Tchana, A. (2021). Simbox bypass frauds in cellular networks: Strategies, evolution, detection, and future directions. *IEEE Communications Surveys & Tutorials*.

Krasić, I., & Čelar, S. (2022). Telecom fraud detection with machine learning on imbalanced dataset. *2022 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 1–6.

Kurani, A., Doshi, P., Vakharia, A., & Shah, M. (2023). A comprehensive comparative study of artificial neural network (ann) and support vector machines (svm) on stock forecasting. *Annals of Data Science*, *10*(1), 183–208.

Lee, C.-W., Fu, M.-W., Wang, C.-C., & Azis, M. I. (2025). Evaluating machine learning algorithms for financial fraud detection: Insights from indonesia. *Mathematics*, *13*(4), 600.

Maes, S., Tuyls, K., Vanschoenwinkel, B., & Manderick, B. (2002). Credit card fraud detection using bayesian and neural networks. *Proceedings of the 1st international naiso congress on neuro fuzzy technologies*, *261*, 270.

Marah, H. M., Elrajubi, O. M., & Abouda, A. A. (2015). Fraud detection in international calls using fuzzy logic. *International Conference on Computer Vision and Image Analysis Applications*, 1–6.

Mitiku, G. A., Gemeda, K. A., & Ramulu, P. J. (2024). Detection and classification of bypass (simbox) fraud using deep learning: A case study on ethio-telecom of ethiopia. *Pan African Conference on Artificial Intelligence*, 32–57.

Omar, S. J., Fred, K., & Swaib, K. K. (2018). A state-of-the-art review of machine learning techniques for fraud detection research. *Proceedings of the 2018 International Conference on Software Engineering in Africa*, 11–19.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *arXiv preprint arXiv:1009.6119*.

Raghavan, P., & El Gayar, N. (2019). Fraud detection using machine learning and deep learning. *2019 international conference on computational intelligence and knowledge economy (ICCIKE)*, 334–339.

Randhawa, K., Loo, C. K., Seera, M., Lim, C. P., & Nandi, A. K. (2018). Credit card fraud detection using adaboost and majority voting. *IEEE access*, *6*, 14277–14284.

Sallehuddin, R., Ibrahim, S., Zain, A. M., & Elmi, A. H. (2015). Detecting sim box fraud by using support vector machine and artificial neural network. *Jurnal Teknologi*, *74*(1).

Subex. (2025). Simbox fraud: Challenges and ai-powered solutions for telecom operators [Accessed: February 4, 2026].

Suite, R. F. R. I. (n.d.). Rsa [Accessed: 2025-09-18].

Suryowibowo, A., Subroto, I. M. I., & Susila, E. N. B. (2019). Simbox identification using k-nearest neighbor based on spectrum analyzer. *Journal of Telematics and Informatics (JTI)*, *7*(3), 158–165.

Syarif, I., Prugel-Bennett, A., & Wills, G. (2016). Svm parameter optimization using grid search and genetic algorithm to improve classification performance. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *14*(4), 1502–1509.

Thanh Noi, P., & Kappas, M. (2017). Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. *Sensors*, *18*(1), 18.

Vajja, R. (2016). *Python oracle integration and data analysis* [Master's thesis, San Diego State University].

Wang, C., Nie, C., & Liu, Y. (2025). Evaluating supervised learning models for fraud detection: A comparative study of classical and deep architectures on imbalanced transaction data. *arXiv preprint arXiv:2505.22521*.

Worldwider, A. (n.d.). Aci [Accessed: 2025-09-18].

Yelland, M. (2013). Fraud in mobile networks. *Computer Fraud & Security*, *2013*(3), 5–9.

Zareapoor, M., Shamsolmoali, P., et al. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia computer science*, *48*(2015), 679–685.

Zhao, Q., Chen, K., Li, T., Yang, Y., & Wang, X. (2018). Detecting telecommunication fraud by understanding the contents of a call. *Cybersecurity*, *1*(1), 1–12.

الكشف عن التجاوزات في اجراء المكالمات الدوليه في قطاع الاتصالات باستخدام تقنيات التعلم الالي وتقنيه موازنه العينات على البيانات غير المتوازنه.

إعداد: وائل أبو نعيم

لجنة الإشراف:

د. مجدي عودة

د. محمد حسين

د. سامي سدر

الملخص

تُمثل أنماط الاحتيال المتعددة تهديداً جوهرياً لمشغلي قطاع الاتصالات، نظراً لما تكبده من هدر جسيم في الإيرادات، وما تخلّفه من آثار سلبية على الموثوقية المؤسسية ومستويات رضا المشتركين. وتتمحور هذه الدراسة حول أحد أبرز هذه الأنماط وهو احتيال تجاوز حركة المرور الدولية. يُصنف هذا النشاط كأحد أكثر الممارسات الاحتيالية انتشاراً، حيث تُعزى إليه خسائر سنوية تُقدر بمليارات الدولارات وفقاً للتقارير الصناعية. وتعتمد آلية هذا الاحتيال على توظيف معدات متخصصة لإعادة توجيه المكالمات الدولية بعيداً عن البوابات الرسمية، مما يتيح للمحتالين تحصيل عوائد مالية غير مشروعة على حساب مقدمي الخدمة. وتستند المنهجيات التقليدية في رصد هذا النوع من الاحتيال إلى الأنظمة الخبيرة القائمة على قواعد إحصائية ثابتة ومحددة سلفاً، حيث يتم تحليل سجلات تفاصيل الأحداث المستقاة من مراكز تبديل الهاتف المحمول حيث يجري تصنيف المشتركين كحالات مشتبه بها بناءً على معايير محددة تشمل: الكثافة الزمنية للمكالمات، مدة الاتصال، تنوع الأرقام المطلوبة، والتوزيع الجغرافي للنشاط. بيد أن هذه المنهجيات التقليدية تفرض تحديات تشغيلية، إذ تتطلب تحديثاً دورياً مستمراً للقواعد، فضلاً عن ارتفاع معدلات "الإيجابيات الخاطئة" ومن هنا تبرز الضرورة العلمية لتطوير نماذج ذكية تعتمد على خوارزميات التعلم الآلي لتعزيز دقة الرصد. وقد استقصى هذا البحث تعقيدات توظيف هذه الخوارزميات، مع التركيز بشكل محوري على أداء الخوارزميتين SVM و RFC. وقد شملت الدراسة إجراء مقارنات دقيقة لتقييم مستويات الدقة قبل وبعد تطبيق تقنيات إعادة أخذ العينات (Resampling)، وذلك بهدف معالجة معضلة "عدم توازن البيانات" (Data Imbalance) المتأصلة في هذا النطاق. وقد أفضت العملية التقييمية إلى أن منهجية "فرط أخذ العينات" (Oversampling) تُمثل الحل الأمثل لضبط توازن البيانات في كلا الخوارزميتين. كما أظهرت النتائج تفوقاً طفيفاً لخوارزمية SVM بدقة بلغت 99.71%، مقابل 99.66% لخوارزمية RF. وتميزت خوارزمية SVM بكفاءة زمنية أعلى، حيث استغرقت تلث الوقت المطلوب لعمليات التدريب والضبط مقارنةً بـ RF، مع تحقيق معدل أدنى في الإيجابيات الخاطئة.

الكلمات المفتاحيه: احتيال الاتصالات، تجاوز حركة المرور الدولية، التعلم الآلي، آلات المتجهات الداعمة (SVM)، تصنيف الغابة العشوائية (RFC)، عدم توازن البيانات، إعادة أخذ العينات، فرط أخذ العينات، سجلات تفاصيل الأحداث.