SIM-Box Fraud Detection Using Data Mining Techniques: The Case of ethio telecom

BY: KAHSU HAGOS

ADVISER: EPHREM TESHALE (PHD)

A Thesis submitted to

School of Electrical and Computer Engineering

Addis Ababa Institute of Technology

In Partial Fulfillment of the Requirements for the Degree of Master of Science

(Telecommunication Engineering)

November, 2018

Declaration

I, the undersigned, declare that the thesis comprises my own work in compliance with internationally accepted practices; I have fully acknowledged and referred all materials used in this thesis work.

Kahsu Hagos

Name	Signature

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

This is to certify that the thesis prepared by **Kahsu Hagos**, entitled *SIM-Box Fraud Detection Using Data Mining Techniques: The Case of ethio telecom* and submitted in partial fulfillment of the requirements for the degree of master of Science (Telecommunication Engineering) complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Committee:

Internal Examiner	_____	Signature	_____	Date	_____
External Examiner	_____	Signature	_____	Date	_____
Adviser	Ephrem Teshale (PhD)	Signature	_____	Date	_____
Director of Postgraduate Program	_____	Signature	_____	Date	_____

_____

Dean, School of Electrical and Computer Engineering

ABSTRACT

Telecommunication fraud is one of the threat of telecom operators as it drives telecom operators to loose a portion of their annual revenue. Bypass fraud is most worrying fraud type in today's telecom business. The advent of new technologies provided fraudsters new techniques to device bypass fraud. Subscriber Identity Module box (SIM box) fraud is the popular type of bypass fraud, that has emerged with the use of Voice Over Internet Protocol (VoIP) technologies. SIM box is used to terminate international calls by diverting away from the legitimate interconnect gateway route. SIM box fraud is more common in the operators where their tariff of international call termination is much higher than the local call tariff. This high tariff is a common method of subsidizing telecom infrastructure in the developing world. However, it creates strong motivation for fraudsters.

Among various fraud prevention approaches, the use of monitoring call patterns and profiles through Fraud Management Systems and Test Call Generators are common one. Yet, both approaches have drawbacks which make them insufficient because they are easily overcome by fraudsters. Therefore, the need for more sophisticated techniques is inevitable. In recent years, datamining techniques have gained popularity in fraud detection.

In this research, models were developed to classify Call Detail Records (CDRs) to propose a model that differentiate fraudulent from legitimate subscribers with better performance. Three classification techniques, Random Forest (RF), Artificial Neural Network (ANN) and Support Vector Machine (SVM), and three user profiling datasets, 4 hour, daily and monthly aggregated were proposed. These three algorithms along with the three datasets were applied in building the models. Results of the work show that RF performed better among the three algorithms with accuracy of 95.99% and a lesser false-positive on the 4 hour aggregated dataset.

KEYWORDS

SIM Box Fraud, telecom fraud, Bypass Fraud, Fraud Detection, Data Mining, Machine learning, classification, Artificial Neural Network, Multi-layer perceptron, Support Vector Machine, Random Forest

ACKNOWLEDGMENTS

I would like to express my gratitude towards

- My advisor Ephrem Teshale (PhD), for his constrictive expertise comments and advice.
- ethio telecom management and staff, for providing data, resources, and valuable expertise support.
- My family and friends, for their encouragement and support.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

CDR	Call Detail Record
GSM	Global System for Mobile Communications
SIM	Subscriber Identity Module
VoIP	Voice over Internet Protocol
FMS	Fraud Management System
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
ROC	Receiver Operating Characteristic
SMS	Short Message Service
SIM-Box	Subscriber Identity Module Box
ML	Machine Learning
ANN	Artificial Neural Network
RF	Random Forest
SVM	Support Vector Machine
CFS	Correlation based Feature Selection
CRISP-DM	Cross Industry Process for Data Mining
MLP	Multilayer Perceptron
TCG	Test Call Generation
IRSF	International Revenue Share Fraud

1

INTRODUCTION

Telecommunications industry has expanded dramatically as the result of communication technology advancements. The number of mobile service subscribers increased with the development of affordable technologies. The expansion in telecommunication industries provides certain characteristics that motivate fraudsters. Fraud method and techniques are increased in parallel to this dramatic expansion [1]. This increase in fraudulent activity brings tough challenges to telecom operators [2]. However, it is a must task to telecom operators to protect their customers and their revenue. Out of the frauds that attempted on telecom operators, International Call Frauds are the expensive calls. International Call Frauds harm the company as well as the customers [1].

Currently, subscribers demand is endless and the technology is moving fast and the fraudsters are advancing in new techniques and tools. Serving customers what they require, while protected from attack encounters operator to sustain in business. This demands telecom operators to move fast and cope up with the changes. However, operators which are using similar methods of protecting their network for several years costs them a lot. Making their telecom ecosystem harder to fraudsters minimizes fraud attempt. This can be achieved by making the cost of doing business for fraudsters to be expensive, through implementing better fraud detection and protection techniques [3]. Detecting frauds prematurely brings time to take corrective action early.

Telecommunication fraud occurs whenever a fraudsters uses deception to receive services free of charge or at a reduced rate. Telecommunication fraud is a worldwide problem and causes substantial annual revenue losses for telecommunication companies [3, 4]. Telecom fraud has always been a center of attraction to fraudsters as it is easy to get a subscription via fake identifications and the mobile terminal is not bound to a physical location. This provides a method for fraudsters with relatively low risk of getting caught [4].

There are different types of telecommunication frauds and these can occur at various levels. Different authors categorized frauds in different ways. For instance, [5, 6] divide Telecommunication fraud in to subscription fraud and superimposed fraud as the most common fraud types,

whereas [4] classifies them in to seven groups as Superimposed, Subscription, Technical, Internal Fraud, Social Engineering, Fraud based on loopholes in technology, and Fraud based on new technology. Similarly, [7] lists the top three types of telecommunication fraud that cause a significant loss to be International Revenue Share Fraud, Premium Rate Service Fraud and Bypass fraud. Subscriber Identity Module Box (SIM-Box) Fraud is a bypass fraud that has emerged with the use of Voice over Internet Protocol (VoIP) technologies which is identified as the most harming fraud type, because it is the main cause for international revenue lose [8, 9]. Moreover, it is a reason for networks traffic congestion that leads to quality of service degradation on the cells overloaded by those SIM-Boxs. SIM-Box fraud is a technique by which fraudsters re-route international calls, by means of SIM-Box device and local Subscriber Identity Module (SIM) cards, from telecom operators and deliver as local calls.

As [9] estimated, revenue loss of telecommunication fraud is $29.2 billion USD, and that of SIM-Box fraud is $ 4.27 billion USD in 2017. Similarly, ethio telecom revenue loss to fraudsters estimated around $52 million USD in 2017 [10].

Major approaches used to overcome SIM-Box fraud includes, Test Call Generation (TCG) testing different international routes of the network, rule based Fraud Management System (FMS) use rules predefined by domain experts [11] and Controlling distribution of SIM Cards. On the other hand fraudsters work extremely in mitigating these detection methods. Fraudsters invent new fraud techniques that protects them from detection. Whenever it becomes known that one detection method is in place, the fraudsters will change their methods and use new. For instance, fraudsters use techniques to avoid test calls to prevent detection, by analyzing the voice call traffic coming and based on defined patterns fraudsters could determine whether the calls are real subscriber calls or originated from a TCG system. Fraudsters use Smart SIM-Boxes which able to imitate the activities of normal subscribers using Human Behavior Simulation to avoids being detected by rule based methods.

1.1 STATEMENT OF THE PROBLEM

SIM-Box fraud is a reason for significant amount of revenue loss for telecom operators. Telecom operators need to have well-organized fraud detection techniques and standards in order to protect themselves and their customers from SIM-Box fraud. Telecommunication fraud detection techniques explore Call Detail Record (CDR) and detect fraud from their patterns. Telecom operators apply different approaches to overcome the problem such as TCG and rule based FMS.

However, the large amount of data generated by telecommunication systems and the dynamic behavior of fraudsters makes TCG and rule based FMS methods easily overcome by fraudsters [5, 11]. As discussed previous fraudsters use techniques to avoid test calls to prevent detection, by analyzing the voice call traffic coming and based on defined patterns they could determine whether the calls are real subscriber calls or originated from a TCG system. In addition, TCG costs much the operator to generate test calls to the entire international routes. Similarly, fraudsters devised smart SIM-Boxes which able to imitate the activities of normal subscribers by using Human Behavior Simulation to avoids being detected by rule based methods. Moreover rule based systems require upgrading to keep them up to date with current methods, upgrade and maintenance costs are high. Besides, rule based methods require very accurate definitions of thresholds and parameters [2, 5].

To overcome this gap, there is a need to explore other techniques that learn the dynamic change of behavior from the data to identify fraudulent activities. Data mining is one of the techniques with a capability to learn from the situation and enrich its performance through learning which is attained by an iterative process. This enables to detect frauds without predefined patterns [2]. Data mining techniques have been practiced based on CDR. Data mining techniques followed different learning mechanisms, mainly supervised and unsupervised. For this particular research, we chose to apply machine learning techniques that able to classify fraudulent patterns from labeled CDR data and provides an optimal solution to detect SIM-Box fraud. In contrast to the related works, our method is trained and tested on a larger data sample, we design some features in different approaches, such as time gap between calls, ratio of distinct outgoing to total outgoing calls, ratio of data usage to outgoing calls and, we tested narrow granularity level. This research attempts to answer the following questions

1. What usage data features can be sensible for SIM-Box fraud detection?
2. What data granularity level is effective in mitigating SIM-Box fraud problem?
3. What machine learning technique can effectively predict patterns of SIM-Box fraudsters behavior form usage data?

1.2 OBJECTIVE

1.2.1 General Objective

To build a model which uses machine learning techniques to detect redirected SIM-Box fraud calls of fraudulent users from usage data with a better performance and accuracy.

1.2.2 Specific Objectives

In order to meet the general objective, the following Specific objectives are listed:

- To classify SIM-Box fraud usage data behaviors and select the relevant attributes for building detection model
- To build a model that will be able to classify fraudulent calls form legitimate calls based on historical usage data
- To identify machine learning tools, techniques and algorithms which are appropriate for SIM-Box fraud detection
- To propose SIM-Box fraud detection model which can be implemented in real environment and attain a better performance and accuracy

1.3 SCOPE AND LIMITATIONS

There are many known fraud types and variety of data that exist in the telecom sector. However, this study has focused only on detecting SIM-Box fraud. There are a number of ways and tools in detecting SIM-Box. In this work we are limited to data mining techniques using CDR data. We took ethio telecom CDR data as a case study.

1.4 CONTRIBUTIONS OF THE RESEARCH

In this research we made the following contributions. Provided insight on SIM-Box fraud detection theoretical views, and SIM-Box fraud detection methods that enables telecom operators to detect SIM-Box fraudulent SIM cards. Increase awareness about this SIM-Box fraud and to show the limitations of conventional techniques for this fraud detection. SIM-Box fraudulent subscribers behavior were studied and useful features of CDR data were proposed. These proposed features

have significant factor in pattern generation from user profile that facilitates SIM-Box fraud detection.

Early detection of fraud provide operators to block fraudsters early before they make harm. Applying a narrow granularity level delivers a near-to-real time fraud detection capability. To our knowledge, different researches tested minimum of one day granularity level. A day of granularity level may provided plenty of time to fraudsters to sustain in the business. In this research we tested the applicability of 4 hour aggregated CDR data for SIM-Box fraud detection. Compared to other granularity levels such as daily and monthly it attains acceptable level of performance. This provide operators to detect fraud in near-to-real time, which helps them to detect fraudsters early.

1.5 LITERATURE REVIEW

A number of research has been conducted in telecom fraud detection and prevention using different tools and techniques. In this section we review some work related to fraud detection in telecommunication industry with more relevance to SIM-Box fraud detection devised machine learning techniques.

The paper [12] recommends to assess the changes on patterns of usage behavior of subscribers over a period of time for fraud detection. CDR data requires analysis to make them applicable for pattern generation and input for data mining techniques as they are unstructured and unorganized. Article [13] presented a rule-based approach to detect anomalous telephone calls. The method described used CDR data sampled over two observation periods (study and test period). The study period contains CDR of customers with non-anomalous behavior. Customers were grouped according to their similar usage behavior such as average number of local calls per week and so on. They come with a probabilistic model to describe their usage for customers in each group. Then the thresholds were determined by calculating change within a group.

A classifier comparison research made by AlBougha, [2] which compares the detection performance of four data mining classification algorithms in detecting SIM-Box fraudulent subscribers from a real CDR data. They trained the classification algorithms applying daily aggregated and labeled dataset. They found that among the Classifiers, Logistic, Boosted Trees, Support Vector Machine (SVM), and Artificial Neural Network (ANN), Boosted Trees and Logistic performed the best.

Hilas [14], in the research with a title "Data mining approaches to fraud detection in telecommunications" describes profiling as cumulated numerical summary of past behavior of users. Future behavior compared to this profile in order to examine the consistency in behavior, any deviation from his may imply fraudulent activity. In their work three different profile types created and tested. The profiles were created from real CDR aggregated daily and weekly as cumulated daily, cumulated weekly, and detailed daily. The selected features includes number of calls duration and fee. These features were categorized as their mean, max and standard deviation values and in time variation of working hour, afternoon, and night.

A research by Sallehuddin *et al.*, [8] with the title "Classification of sim box fraud detection using support vector machine and artificial neural network." They design and compare two classifiers SVM and ANN. A CDR data of 234,324 calls made by 6,415 subscribers from only one Cell, so it contains only one cell-ID, for a period of two months were collected. The dataset consisted of daily aggregated 2,126 fraud subscribers and 4,289 normal subscribers which are of two-thirds legitimate and one-third SIM-Box fraudulents. The researchers extracted 9 features, like Total Calls, Total Number Called, Total Minutes and Average Minutes, etc. Then they used the extracted features to train the two proposed classifiers. They trained the algorithms applying the prepared dataset and variety of parameter settings. Finally, they found that SVM has better accuracy compared to ANN, SVM gave 99.06% accuracy with lesser training time, while ANN model gave 98.69% accuracy

Another research was conducted by Murynets *et al.*, [15] with the title "Analysis and detection of SIMbox fraud in mobility networks." They applied supervised classification techniques. The classifiers are linear combination of three decision tree (alternating decision tree, functional tree, and random forest). They applied it on real CDR data form an operator in USA, a larger dataset was used with accounts distributed nationwide. The data is collected having 93000 legitimate accounts and 500 of fraudulent accounts. For training the classifier, they split the dataset to two-thirds for training and one third for testing. Using International Mobile Equipment Identity (IMEI) as a device identifier other than the subscriber identifier they computed 48 features characterizing patterns of legitimate and fraudulent IMEIs. They observed that fraudulent SIM-Boxes have common patterns as the following. High number of International Mobile Subscriber Identity (IMSI)s per IMEI, Static physical location, large number of international phone calls and large volume of outgoing calls generated compared to incoming calls. And the classifier attained an accuracy of 99.95% with lowest false positive achieved by the Random Forest (RF).

In contrast, to the use of CDR for detecting SIM-Box, Reaves *et al.* [16] used real-time call audio analysis. They designed a system that relies on the raw voice data received during a call to distinguish errors from the distinct audio distortions caused by delivering the call over a VoIP. They used fast signal processing techniques to identify whether the calls are made by a SIM-Box. Their resultant system was able to detect 87% of real SIM-Box calls in only 30 seconds of audio with no false positives.

To the level of our Knowledge, related works applied wider data aggregation period, i. e. daily, weekly, monthly, etc. This may increase generalization accuracy, though it opens ample time for fraudsters to make money by extending the time to action for operators. Which able fraudsters to stay in business. Subsequently narrow data granularity level is recommended in order to adopt near-to-real-time detection scheme [2, 17]. However there is a trade-off, lesser Data granularity level may diminish suitable patterns for fraud detection.

In this work, we tested the 4 hour granularity level with comparison to daily and monthly. In addition, in contrast to the related works, our method is trained and tested on a larger data sample. And we design some of features in different approaches, such as time gap between calls, ratio of distinct outgoing to total outgoing calls, ratio of data usage to outgoing calls.

1.6 METHODOLOGY

The main purpose of this study is to develop predicting model for detecting SIM-Box frauds using data mining techniques. In order to achieve the objectives of the study and answers the research questions, the following method was designed. To understand and formulate the problem domain, academic literature on telecom fraud specifically related to SIM-Box fraud detection were reviewed. Furthermore, successive discussions with domain experts of the field were Performed. Dataset were prepared from collected one month CDR data, the collected data Understood, analyzed and selected relevant features for the research. Then after the selected features were preprocessed to make it ready for the experiment. Finally datasets with labeled records, fraud and legitimate, were ready for training and testing the models. Predictive models were trained applying the proposed supervised machine learning algorithms and prepared dataset. The developed models were tested and evaluated applying variety of performance measure tools. Results were discussed and reported. At the end outcomes were summarized and recommendations were provided

1.7 THESIS ORGANIZATION

This thesis contains six chapters. Chapter 1 describes introduction to the research, which covers the research background, statement of the problem, outline of the research objectives, review of Literature and related works, and concludes with an outline of the structure of the Thesis. Chapter 2 determined the techniques on telecommunication fraud and SIM-Box fraud problems, detection mechanisms and limitations. Chapter 3 deals with the concepts of machine learning, machine learning methodology and machine learning methods which are used in this study. Chapter 4 is about data preparation, which deals in collecting describing preprocessing the data to make it ready for experimentation. Chapter 5 focuses on experimentation, which includes building models, testing and evaluating the models, and discussing the outcomes of the experiment. The last chapter, Chapter 6, is dedicated to convey conclusions and recommendations of the study.

2

TELECOMMUNICATION FRAUD

Telecommunication Fraud can be defined as any activity by which service is obtained with intention of not to pay [18]. The intention of perpetrators is progressed from not willing to pay to making money [3, 19]. Telecommunication Fraud is a worldwide problem that affects revenue of operators. It is also a treat to the national security beyond the economic losses [19, 20]. The practice of fraud is usually related to money, but there are some other reasons like political motivations, personal achievements and self-preservation that motivates fraudsters [4]. Telecom fraud arise to exist from the time when the commercial telecommunication service begins. Since then, the fraudsters have been causing financial damage to the companies who offered telecom services. Global Fraud Loss Survey shows mobile communications industry lose $29.2 billion in 2017 [9]. As [9, 12] indicated the fraud lost by operators is about 2% to 3% of their total revenue.

Since the beginning of commercial telecommunications, the fraudsters have been causing financial damage to the companies who offered these services [5]. Due to the technological advances, the cost of a fraud attack to the carrier has been decreasing, but on the other hand, the amount of occurrences of fraud attack have been increasing creating a constant financial damage [5]. Despite the telecommunication industry suffers major losses due to fraud, there is shortage of comprehensive published research on this area mainly due to lack of publicly available data to perform experiments on. On the other hand, any broad research published publicly about fraud detection methods utilized by fraudsters to evade detection [1, 7, 21].

More recently, telecommunication fraud increased with the dramatically expansion of telecommunication industries [1, 8] in method and techniques of frauds, this offers a high challenge to telecom operators. However it is a must task to operators to protect their customers and their revenue. That is why operators investing in new security systems in order to prevent and fight fraudsters.

Even though fraudsters change their way of attacking techniques frequently, knowing patterns of frauds deeply is vital in preventing fraud. In order to understand fraud patterns deeply studding normal usage gives way to identify anomaly behavior of the network usage. Knowing the

patterns of fraud guides to move in parallel with the fraudsters to device new detection and prevention method. There is endless fight between the operators and fraudsters, the operators trying to protect their systems from fraudsters and the fraudsters also trying to identify and use weaknesses on the systems [4].

These days, subscribers demand is endless and the technology is moving fast and the fraudsters are advancing in new techniques and tools. Serving customers what they require, while protected from attack encounters any operator to stay in business. This demands to move faster with along the change. On the other hand, operators which are using old methods of protecting their network for several years may costs them a lot. Making the cost of doing business for fraudsters higher, i. e. advancing in techniques of fraud detection, forces fraudsters to move to other operator [3]. Making your telecom ecosystem harder minimizes the frequency of fraudsters attempt. In addition, detecting fraud prematurely gives time to take corrective action early.

2.1 TELECOMMUNICATION FRAUD TYPES

Telecommunication industry, being one of the major sectors in the world infiltrated by fraud. Telecommunication fraud is a combination of variety of illegal activities like unauthorized and illegitimate access, subscription identity theft and revenue share etc.

There are different types of telecommunications fraud and these can occur at various levels. Different authors categorized frauds in different ways. For instance, [5, 6] categorized subscription fraud and superimposed fraud as the most common fraud types, whereas [4] classifies them in to seven groups as Superimposed, Subscription , Technical , Internal Fraud, Social Engineering, Fraud based on loopholes in technology, and Fraud based on new technology. Similarly, [7] lists the top three types of telecommunication fraud that cause a significant loss to be International Revenue Share Fraud, Premium Rate Service Fraud and Bypass fraud. [19] classifies as Contractual, Hacking, technical and Procedural frauds.

The development of telecommunication and technology, and the big size of telecom market that found very attractive to fraudsters [22], the traditional types of fraud has been replaced with more complex ways of frauds that was spread too fast in the world. Types of fraud can be also divided into fraud in traditional networks and fraud in new technology. The first type of fraud has many ways such as subscription fraud, which is the signing up for a service using fake or stolen identification, with no commitment of paying the bills. And other types such as SIM cloning, premium Rate Service fraud, internal fraud, dealer fraud, roaming fraud, and calling

card fraud. The other fraud type is based on VoIP network, which can be done by using VoIP techniques. As example of this type includes call transfer fraud, location route number fraud and bypass fraud.

In this paper we discussed two fraud types, International Revenue Share Fraud and Interconnect Bypass Fraud, that are categorized as the most worrying fraud type by [9].

2.1.1 *International Revenue Share Fraud*

International Revenue Share Fraud (IRSF) is the largest contributor to the overall fraud losses according [9]. It occurs when an operator makes an agreement with another party which will generate calls to premium rate number to generate revenue for increasing traffic. IRSF often involves a combination of multiple fraud schemes. One of the technique is exploiting roaming SIM cards or Dialer Mal-ware, call divert and call forwarding, social engineering techniques. Fraudsters generate high traffic calls to high cost destinations and gets revenue from the sharing agreements.

2.1.2 *Interconnect Bypass Fraud*

Interconnect Bypass fraud is unauthorized insertion of traffic onto another carrier's network. This fraud type can be named as Interconnect fraud, Global System for Mobile Communications (GSM) Gateway fraud, or SIM-Boxing. This scenario requires that the fraudsters have access to advanced technology such as VoIP, which is capable of making international calls appear to be cheaper domestic calls, effectively bypassing the normal payment system for international calls. The fraudsters will typically sell long distance calling cards. When customers call the number on the cards, operators are able to switch the call to make it look like a domestic call.

The most common implementation of interconnect bypass fraud is known as SIM-Boxing, which enabled by VoIP GSM gateways. SIM-Box Fraud transport international calls through VoIP, then routs them to local cellular network via a collection of SIM cards inside a SIM-Box device. The main concern of this thesis, SIM-Box fraud, were discussed in detail in Section 2.2.

2.2 SIM-BOX FRAUD

SIM-Box fraud is a technique by which local SIM cards used for rerouting international calls away from mobile network operators, transfer them over the Internet and deliver them back by means of VoIP gateway device called SIM-Box, as local calls to the operators cellular network. As a result,

the calls become local at the destination network, and the telecom operators of the intermediate and destination networks do not receive payments for call routing and termination.

The drive of this fraud type is the tariff difference between local call and international call, where termination costs of international calls is high [15]. This difference in tariff is high in the developing world as it subsidizes their telecommunication infrastructure expansion cost [16]. Since this international call is terminated on the receiver side as local call, the operator charged it considering as a local call. The telecom operator loses a revenue of tariff difference between local and international calls. In the contrary the fraudster benefits from this difference in tariff of the calls. The loss in revenue due to SIM-Box fraud globally, as estimated by [9] is about $ 4.27 billion USD, this is a 6.8% of the total fraud loss estimated globally $29.2 billion USD, so bypass will be financially worthwhile. As [16] fraudsters device different techniques, including present themselves as a legitimate telecommunications company, offering discounted call rates directly through the sale of international calling cards, to collect this profit.

This fraud type is a reason for interconnect revenue lose, traffic congestion and quality of service degradation [16]. The cells where it operates overloaded, and voice calls routed over a SIM-Box have poor quality, which results in customer dissatisfaction.

A SIM-Box is VoIP gateway device that maps international calls from VoIP to a local SIM card of the mobile operator. SIM-Box fraud is a fraud type that has emerged with the expansion of VoIP technologies, and its successes is depend on the obtainability of SIM card and SIM-Box devices. In countries that have less control on the distribution of SIM cards and availability of SIM-Box devices as observed in some countries of Africa and Asia, this fraud type is a main challenge [8]. SIM-Box equipment includes SIM slots, antennas, and Ethernet ports that can be used to get the SIM-Box equipment connected to the Internet[2]. Fraudsters install SIM-Box with multiple SIM cards, by means of this setup the fraudsters can forward international calls through local phone numbers in the respective country to make it appear as local call. This way, fraudsters bypass interconnect charges.

2.2.1 *SIM-Box Fraud Scenario*

Whenever a subscriber places a call to an international destination, the call passes through different entities [2]. To explain how SIM-Box fraud is committed, it is sensible first to describe the legitimate international call route, then after, discuss the fraud bypass scenario. Suppose

subscriber A and subscriber B Reside in different countries, country A and B respectively. In legitimate route of an international call:

- Subscriber A places a call to subscriber B over the mobile operator and pays the service provider for the call.
- The call generated by subscriber A forwarded to international gateway in country A.
- The home international gateway of country A routes the received call to a transient operator and pays for it.
- The transient operator then routes this call to a destination (country B) international gateway and pay a toll to the destination international operator.
- Finlay, the international gateway of country B terminates the call through his network to subscriber B

This legitimate transaction route is Demonstrated in Figure 2.1.

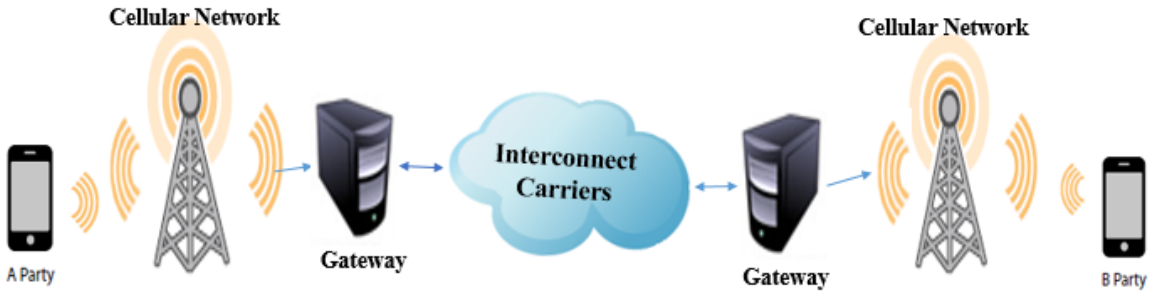


Figure 2.1: Legitimate Route of International Call, adopt from [7]

Where as in SIM-Box fraud route of an international call [15]:

- Subscriber A places a call to subscriber B in the domestic mobile operator network and pays it for the call.
- The call generated by subscriber A forwarded to home international gateway in country A.
- The home international gateway of country A routes the received call to a transient operator and pays for it.
- The transient operator then routes this call to a SIM-Box placed in country B using VoIP and pay a toll to the SIM-Boxer.

- The SIM-Box then places a separate call on the network of country B to subscriber B using its local SIM card, that is why it looks like a local call, and pay only for the local call by avoiding interconnect cost.
- Finlay, subscriber B in country B receivers a international call from abroad but with a local number, it may Amazed.

IN Figure 2.2 the SIM-Box fraud rout of international Call is illustrated diagrammatically.

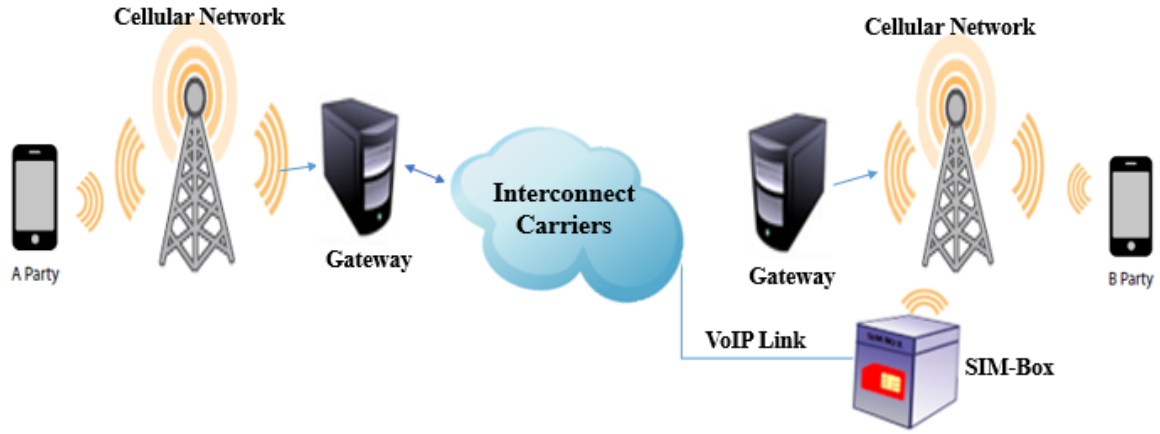


Figure 2.2: SIM-Box Fraud Rout of International Call, adopt from [7]

2.2.2 *SIM-Box Fraud Detection*

Major methods used in battling SIM-Box Fraud nowadays includes TCG, rule based FMS and Controlling distribution of SIM Cards.

TCG is used as an active method to detect bypass fraud, where operators test different international routes to their network and analyze whether calls coming through an international number or local number routes. If it came from a local number, Definitely it is associated with some SIM card used in a SIM-Box that granted the fraud department to process it. TCG is one of the effective methods to detect SIMs used in SIM-Boxes. However, this method costs the operator extremely to generate test calls to the entire international routes. Also, since this method is based on generating calls to random numbers, getting SIM cards used by SIM-Boxes is probabilistic.

FMS is based on predefined rules and threshold that are manually defined by domain field experts aimed to segregate SIM-Box fraudsters. These rules and threshold are summarized features constructed from CDR data of subscribers. These are used to identify abnormal behavior of SIM cards by comparing with analyzed subscriber profile. Through this process it able to distin-

guishes fraudulent from legitimate usage, if abnormal subscriber behavior is observed then the system generates an alarm and experts performs analysis on it.

SIM Card Distribution Control SIM cards are indispensable to SIM-Box fraudsters to survive, fraudsters need to maintain sufficient supply of SIM cards to be in business. However, SIM card distribution control will make this process difficult. Requiring government IDs and limiting the number of SIM cards per ID will prevent fraudsters from obtaining a large number of SIM cards to install in their SIM-Boxes.

On the other hand fraudsters work extremely in mitigating these detection methods. They invent new techniques that makes them Subsist in the business. For instance, fraudsters use tricks to avoid test calls to prevent detection, by analyzing the voice call traffic coming toward their SIM-Boxes and based on usage and other patterns they could determine whether the calls are real subscriber calls or originated from a TCG system. They could then either block the test calls from reaching the SIM-Box, or reroute the calls to a legitimate route. Smart SIM-Boxes are designed to imitate the activities of normal subscribers by using Human Behavior Simulation. This technique makes generated fraudulent calls appear like legitimate, which avoids being detected by typical detection means, such as rule based methods.

Human Behavior Simulation encompasses the following methods [2]:
SIM Migration: Fraudsters are deploying many gateways in different locations, and once in a while they swap the SIM cards between the gateways, so it would look like that the user is moving. The swapping operation could be done manually or automatically using software.

SIM Automatic Rotation: Fraudsters can be detected easily if they operate their SIMs excessively, so they limit their usage by rotating SIMs. This will make SIMs operate in limited hours a day, which simulates the behavior of ordinary customers.

Usage of Other Network Services: Most of the SIM-Boxes are using only voice services and that exposed them to detection. In order to Safeguard themselves from this vulnerability smart SIM-Boxes are making calls and sending Short Message Service (SMS) to each other by applying call. This helps them to look like legitimate.

Changeable IMEI: number SIMs associated to a single IMEI is important feature to identify SIM-Box fraud. To overcome this weakness,SIM-Boxers device a technique provides the ability to change the IMEI associated to any inserted SIM inside the SIM-Box.

Family Lists: Traditional SIM-Boxes just reroute the call from VoIP to the GSM network, so they make calls to large numbers of different network customers. A smart SIM-Boxes assign list of numbers to a specific SIM. This helps evading the setup due to large different numbers detection.

Human Behavior Simulation makes the work of detecting bypass fraud challenging. Whenever it becomes known that one detection method is in place, the fraudsters will change their tactics. The fraudsters dynamic behavior, the vastly increase of generated amount of data, and availably of SIM-Box equipment easily forces SIM-Box fraud detection to find advance methods.

2.2.3 *SIM-Box Fraud Features*

Even if the dynamic innovation of SIM-Boxers makes SIM-Box fraud detection harder through different techniques to diminish behavior pasterns, this may not succeed for advanced methods because changes in user behavior can be observed in the call data which define usage patterns. The following common patterns are Still Worthwhile [7] in detecting SIM-Box fraud.

- since the probability of a SIM-Box subscriber calls to the same subscriber she calls before is very small. To the contrary a legitimate subscriber may call repeatedly to similar subscriber, hence the ratio of distinct calls to its total calls is an important attribute in detecting SIM-Box fraud.
- The time between calls of fraudulent subscribers is very small whereas that of the legitimate subscribers is relatively larger. This is due to that of the SIM-Box will receive a call and redirect it as soon as the previous call ended but in the case of a legitimate subscriber it took some time to generate new call after it ends the current call. So the call duration ratio to the time of calls is an important feature.
- A SIM-Box contains so many SIM cards and one or few IMEI, High number of IMSIs per IMEI will be perceived.
- Static physical location of SIM-Boxes
- Fraudulent subscribers generate large amount of outgoing compared to incoming calls.
- Even though they send SMS and use Internet data to look like as legitimate subscriber, it is very small compared to normal subscribers
- The subscription age of fraudulent subscribers is lesser compared to legitimate subscribers.

3

MACHINE LEARNING

Machine Learning (ML) is the study of algorithms that automatically improve their performance, with experience enrich their performance through learning, which is attained by an iterative process [2]. It provides tools by which large quantities of data can be automatically analyzed. ML algorithms have been used to build classification rules from large datasets. There are several applications for ML, the most significant of which is data mining. This chapter discusses the concepts, approaches and techniques of ML algorithms applied in this thesis work.

3.1 DATA MINING

Data mining is the extraction or *mining* of knowledge from a large amount of data [23]. Data mining is an interdisciplinary field that employs the use of analysis tools from statistical models, mathematical algorithms, and machine learning methods to discover previously unknown, valid patterns and relationships in large data sets, which are useful for detecting fraudster behaviors. The strong patterns discovered by data mining techniques can be used for the non-trivial prediction of new knowledge. Data mining is a discipline that draws on sophisticated skills in computer science, machine learning, and statistics. Due to the large amounts of data in telecommunication companies and fraudsters attempt to gain access to the data, Data mining, machine learning, statistics, and other interdisciplinary capabilities are needed to address the challenges of fraud.

The Cross Industry Process for Data Mining (CRISP-DM) provides a common and well-developed framework for delivering data mining projects. CRISP-DM identifies six steps within a typical data mining project: [23]

1. Problem Understanding
2. Data Understanding
3. Data Preparation
4. Modeling

5. Evaluation

6. Deployment

Learning patterns from subscriber behaviors is critical for fraud detection and prediction. Learning these behaviors is important, as they can identify and describe structural patterns in the data which reveals the hidden intention of fraudsters. Learning to predict require complex computation that calls for machine learning algorithms. The upcoming sections discussed certain machine learning algorithms proposed for this thesis.

3.2 MACHINE LEARNING ALGORITHMS

Machine learning is defined as the complex computational process of automatic pattern recognition and intelligent decision making based on training sample data [24]. Machine learning is considered as a subfield of Artificial Intelligence, concerned as the development of techniques and methods enable the computing machine to learn. Machine learning is an automated detection of meaningful patterns with in the data [25][24]. Machine learning algorithms have proven to be of great practical value in a variety of application domains, such as data mining, hidden and dynamically adapt to changing conditions [26]. Machine learning methods are evaluated by comparing the learning results of methods applied on the same data set or quantifying the learning results of the same methods applied on sample datasets[24][27].

3.2.1 *Types of machine learning*

There are four general machine learning methods, supervised, unsupervised, semi-supervised and reinforcement. This section introduces the types more focused on the supervised machine learning techniques.

A *Supervised Learning*

Supervised models can be described as learning a function $f(x) = y$, where y is the label (also called class) of the data and x denotes the attributes of these examples (also called features). Supervised learning models are trained with data that have been pre-classified [23]. The examples of input/output functionality are referred to as the training data. Care needs to be taken in order to ensure that the training data is correctly classified. The supervised learning methods are categorized based on the structures and objective functions of learning algorithms. Popular

categorizations include ANN, SVM, and decision trees [28]. In the case of fraud detection, since legitimate calls occur more often than fraudulent calls, the training data will mostly contain legitimate calls, leading to a misclassification of the model. This needs attention in the supervised learning models. In the upcoming sections discussed some popular supervised machine Learning algorithms.

B *Unsupervised Learning*

In this learning method, no label is given in sample data, where instances are unlabeled. Most of the time, to gate pre-classified training data is difficult, in such cases, unsupervised models are used to find groups in the training data. The aim of unsupervised learning is to identify patterns in the data that extend our knowledge and understanding of the world that the data reflects [23]. Even though they are difficult to evaluate [28], unsupervised models have advantage over supervised models, that new types of fraud may be identified. The most famous unsupervised learning methods include k-means clustering, hierarchical clustering, and self-organization map.

C *Semi-supervised Learning*

In semi-supervised learning, the given data are a mixture of classified and unclassified data. This combination of labeled and unlabeled data is used to generate an appropriate model for the classification of data [25].

D *Reinforcement Learning*

It's a type of Learning where an agent learns how to behave in a environment by performing actions and reinforcement based on the results. Reinforcement learning goes through the following steps:

1. Input state is observed by the agent.
2. Decision making function is used to make the agent perform an action.
3. The agent receives reinforcement from the environment.
4. The state action pair information about the event is stored.

3.2.2 *Random Forest*

From experience in data mining, models working together are better than one model doing it all. This incites the idea of combining multiple models into a single ensemble model. These days, it is common to see a number of algorithms generating ensembles, including boosting, bagging, and RFs. RF is the most popular ensemble classifier [29], that rely on the principle of combining multiple classifiers and diverse hypotheses, and can potentially lead to much more robust model than learning a single model. RF is a supervised classification algorithm consists a collection of tree structured classifiers. This classifier grows independent identically distributed random vectors and each vector casts a unit vote for the most popular class at the input [29].

The output of random forest is decided by the votes given by all individual trees. Each decision tree is built by classifying a random samples of the input data using a tree algorithm. Each tree has a decision to label any testing data. The RF model decides the classification result of the testing data after collecting the votes of all the tree models. The RF algorithm is presented in terms of decision trees. Though, the RF algorithm is a meta algorithm, any one of model building algorithms could be the actual model builder [29].

A *Knowledge Representation*

In any ensemble approach, the key extension to the knowledge representation is in combing the decisions made by the individual models. In deploying the model, the builder algorithm builds all models and then combines them into one model, resulting in a better overall model. At each node, decision tree performs a simple test with a single feature, to improve classification accuracy. A functional tree linearly combines several features at decision nodes and leaves, which is advantageous for large data sets.

RF algorithm builds many decision trees, the number of trees to build is a parameter, which can be selected during the learning phase. Every decision tree is learned with a random subset of features from a sampled training set with replacement. The output is decided by the votes given by all individual trees. Each decision tree is built by classifying the samples of the input data using a tree algorithm. Then, every tree will be used to classify testing data. Each tree has a decision to label any testing data. This label is called a vote. Finally, the forest decides the classification result of the testing data after collecting the votes, and the most popular class is returned. This scenario is illustrated in Figure 3.1 diagrammatically.

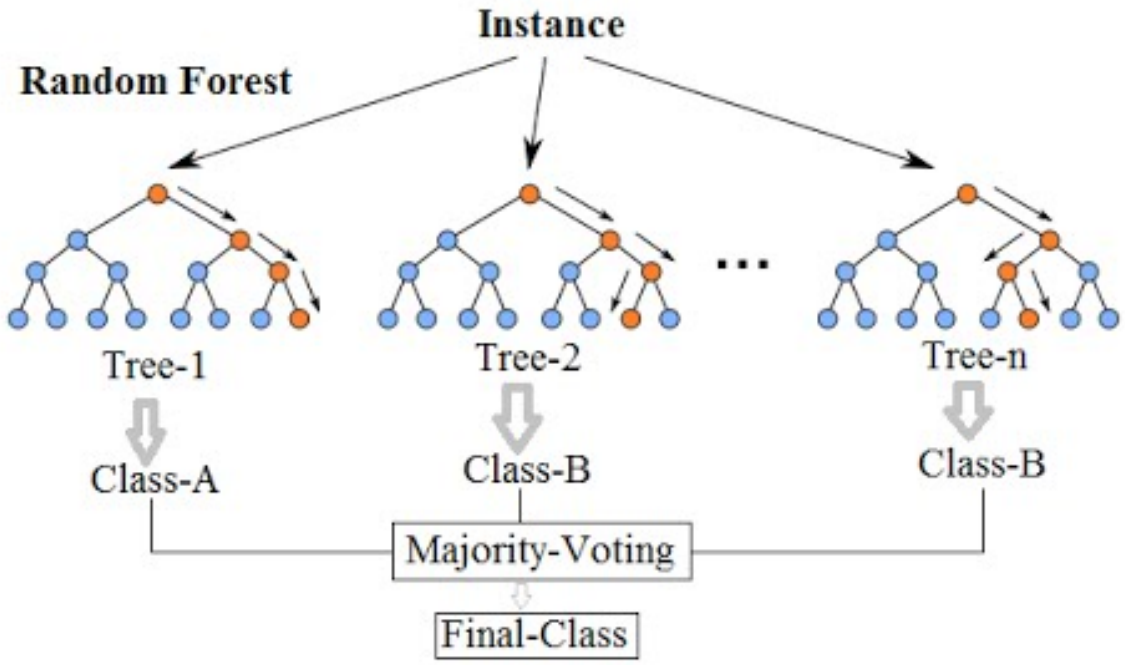


Figure 3.1: An Ensemble Classifier for RF [30]

The RF algorithm tends to be much more robust to changes in the data. Hence,it is very robust to noise, means that small changes in the training dataset will have little impact on the final decisions made by the resulting model. RF models are generally very competitive with nonlinear classifiers such as ANN and SVM. The RF algorithm tends to produce quite accurate models because the ensemble reduces the instability that observed in build single decision trees. It handle large data set with higher dimensionality.

In building a single decision tree, the model builder may select a random subset of the observations available in the training dataset. Also, at each node in the process of building the decision tree, only a small fraction of the available variables are considered. This Significantly reduces the computational requirement. It is also suitable when there are many input variables and little observations and it can handle large dataset with higher dimensionality. RF algorithms build patterns and detect outliers [28]. The strength of individual trees in the forest and the correlation between them determines the generalization error of the forest and its trees [29].

RF Algorithm performs the following sequence of steps to accomplish its classification.

Step:1 Choose T number of trees to grow

Step:2 Choose m no of variables used to split each node. $m \ll M$, M(input variables)

Step:3 Grow T trees.

When growing each tree do

- Construct a bootstrap sample of size n sampled from S_n with the replacement and grow a tree from this bootstrap sample
- At each node select m random variables and use them to find the best split

Step:4 Grow the tree to a maximal extent and there is no pruning

Step:5 To classify point X collect votes from every tree in the forest and then use majority voting to decide on the class label

A unique characteristic of RF is that only a small number of features are randomly selected for training each tree. Small values of features have shown to yield high performance values. The parameter node size controls the size of terminal nodes during node splitting while training a tree,[29] has recommended using at least 100 trees results in good performance.

Using random features has desirable characteristics such as

- Its accuracy is better than Adaboost.
- It is relatively robust to outliers and noise.
- It is faster than bagging or boosting.
- It gives useful internal estimates of error, strength, correlation and variable importance.
- It is simple and easily parallelized.
- Ability of running on large datasets.

RF model is a good choice for model building, for the reasons that, need very little preprocessing of the data, resilient to outliers, no need variable selection (the RF model builder is able to target the most useful variables), each tree is effectively an independent model (each decision tree is not influenced by the other decision tree when constructed), and the model builder tends not to over-fit to the training dataset.

3.2.3 Artificial Neural Network

ANN is a system based on the biological neural network, such as the brain. According to [31], neural networks represent a brain symbol for information processing. ANN have the capabil-

ity to learn from their environment through an iterative process of adjustments applied to its synaptic weight and bias level. They are also able to improve their performance through learning. There are many varieties of learning algorithms for the design of ANN. They differ from each other in the way in which the adjustment to a synaptic weight of a neuron (node) is formulated. Learning algorithms can be described as a prescribed set of well-defined rules for the solution of a learning problem. Error-correction, memory-based, competitive, and Boltzmann learning are among the learning algorithms for ANN. ANN learning paradigm is either supervised (associative learning) or unsupervised (self-organizing). In the case of supervised, there is a need to train or teach the input and output pattern. But for the case of unsupervised neural network, it only requires input patterns from which it develops its own representation of the input stimuli.

ANN can be classified in to *Feed forward Neural Network*, *Recurrent Neural Network* and *Self-Organizing Map* [31]. In *Feed forward Neural Network*, activation is piped through the network from input units to output units. Sometimes they are also referred as static networks. It contain no explicit feedback connections. Conventional *Feed forward Neural Network* are able to approximate any finite function as long as there are enough hidden nodes to accomplish this. It is the first and simplest type of ANN. *Recurrent Neural Network* on the other hand, are dynamical networks with cyclic path of synaptic connections which serve as the memory elements for handling time-dependent problems. *Self-Organizing Map* mainly used for cluster analysis. The big developments in ANN during the past few decades have motivated human ambitions to create intelligent machines with human-like brain. Nowadays, ANN are considered one of the most efficient pattern recognition, regression, and classification tools [27] [31].

When ANN is used as a supervised machine-learning method, efforts are made to determine a set of weights to minimize the classification error. One well-known method that is common to many learning paradigms is the least mean-square convergence. The objective of ANN is to minimize the errors between the ground truth Y and the expected output $f(X; W)$ of the network as $E(x) = (f(X; W) - Y)^2$. The behavior of ANN depends on both the weights and the transfer function, which are specified for the connections between neurons. ANN models implicitly define the relationships between input and output, and, thus, offer solutions for tedious pattern recognition problems, especially when users have no idea what the relationship between variables is.

A *Perceptron*

Perceptron is the simplest kind of ANN, which consists of a single neuron that can receive multiple inputs and produces a single output. Perceptrons are used to classify linearly separable classes. As illustrated in Figure 3.2, a perceptron takes a vector of real-valued inputs, calculates a linear combination of these inputs, then outputs a 1 if the result is greater than some threshold and -1 otherwise using the selected function. The precise learning problem is to determine a weight vector that causes the perceptron to produce the correct output for each of the given training examples. The most common way that the perceptron algorithm is used for learning from a batch of training instances is to run the algorithm repeatedly through the training set until it finds a prediction vector which is correct on all of the training sets. This prediction rule is then used for predicting the labels on the test set.

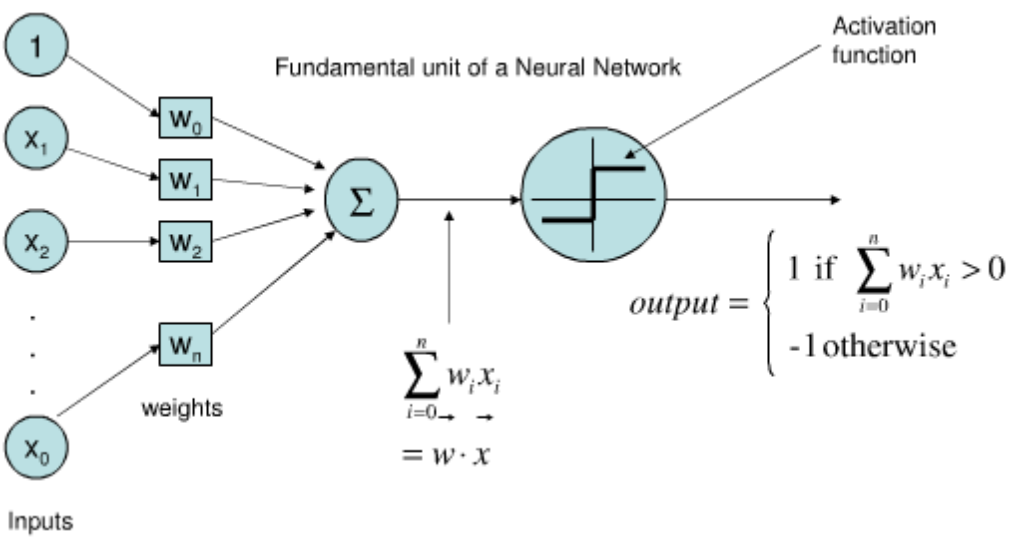


Figure 3.2: A Perceptron with Multiple Inputs and Single Output [32]

In perceptron model, the weighted sum is calculated using Equation 3.1 then evaluated and passed to an activation function, which compares it to a predetermined threshold θ. If the weighted sum is greater than the threshold θ, then the perceptron fires and outputs 1, otherwise it outputs 0 (-1).

$$\sum_{j=1}^{m} w_j.x_j = w_1.x_1 + \ldots + w_m.x_m \tag{3.1}$$

There are Varieties of activation functions that can be used with the perceptron, but the *step*, *sign*, *linear*, and *sigmoid* functions are the most popular ones.

- *step* $f(x) = \begin{cases} 1 \ if \ x > 0 \\ 0 \ if \ x < 0 \end{cases}$
- *sign* $f(x) = \begin{cases} 1 \ if \ x > 0 \\ -1 \ if \ x < 0 \end{cases}$
- *linear* $f(x) = x$
- *sigmoid* $f(x) = \frac{1}{1+e^{-x}}$

All mentioned above activation functions are triggered at a threshold $\theta = 0$. However, it is more convenient to have a threshold other than zero. For that, a bias b is added to the perceptron in to the inputs. The role of this bias b is to move the threshold function to the left or right, in order to change the activation threshold. Training the perceptron aims at determining the optimal weights and bias value at which the perceptron fires.

B *Multilayer Perceptron*

A single perceptron can solve any classification problem for linearly separable classes. If given two nonlinearly separable classes, a single layer perceptron network will fail to solve the problem. Such a nonlinearly separable problem is solved by using most popular types of ANN Multilayer Perceptron (MLP) [33]. In a MLP neural network, each perceptron receives a set of inputs from other perceptrons, and according to whether the weighted sum of the inputs is above some threshold value, it either fires or not. The ANN/MLPis ideally composed of three layers, the input layer, the hidden layer, and the output layer as show in Figure 3.3. The input layer consists of input nodes which represent the system's variable. The hidden layer consists of nodes which facilitate the flow of information from the input to the output layers. The flow is controlled by weight factors associated with each connector. The output layer consists of nodes which represent the system's classification decision. The values of the output nodes are compared with Limits to determine the output and classify each case.

The weight adjustment, training process consists of running input values over the network with predefined classification output nodes. This process runs until the weight values are minimized to an error function. Testing samples are used to verify the performance of the trained network. As [27] stated, training is defined as the process of iterating through the training set to adjust the weights. To learn a neural network, random weights and biases are generated at first. Then,

a training instance is passed to the neural network, where the output of each layer is passed to the next layer until computing the predicted output at the output layer, according to the initial weights. The error at the output layer is computed as the difference between the actual and predicted outputs. According to the error, the weights between the output layer and the hidden layers are corrected, and then the weights between the hidden layer and the input layer are adjusted in a backward (The best-known example of a neural network training algorithm back propagation) fashion. Another training instance is passed to the neural network and to the process of evaluating the error at the output layer, thereby correcting the weights between the different layers from the output layer to the input layer. Repeating this process for as many epochs will help in learning the neural network.

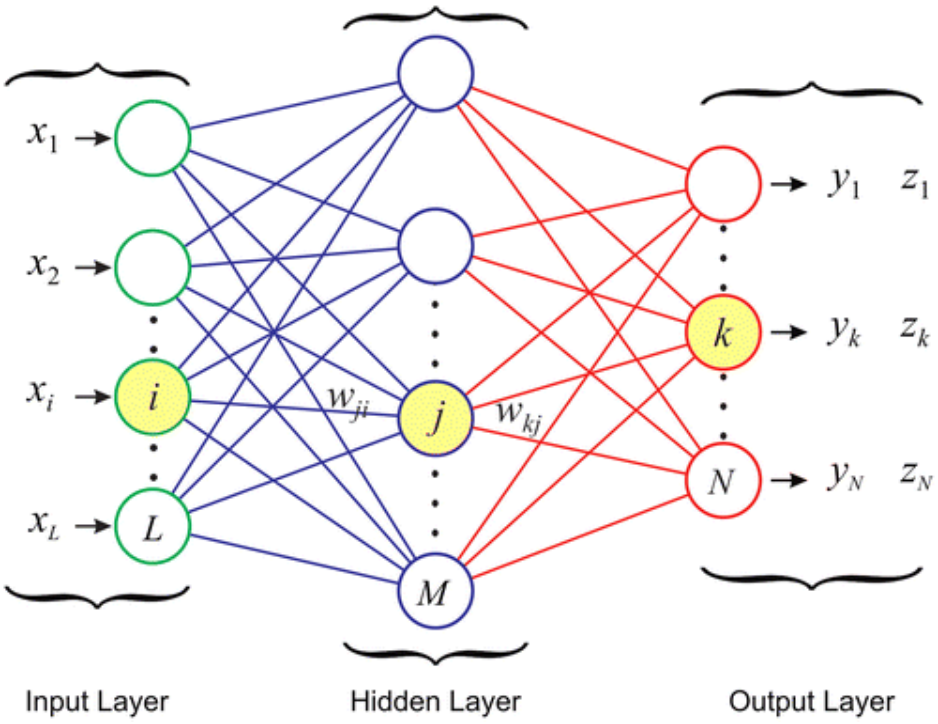


Figure 3.3: A Multilayer Perceptron Neural Network [32]

ANN are popular due to their robustness, fault tolerance, ability to learn and generalize adaptability, and parallel data processing. This enables them to solve complex non-linear and multi input-output relationship problems. And also are useful in practical applications, due to their ability to do non-linear mapping, parallel processing methodology, ability to learn from the environment and their subsequent adaptability to the environment. When compared to other methods, ANN have been shown to be superior as modeling technique for data sets with non-linear relationships. This enables them to be applied for data fitting and prediction. Pattern

recognition, forecasting, prediction, and classification are some of the business applications where neural networks have been used [31].

There can be some issues noticed in ANN, some of them are having many local minima and also finding how many neurons might be needed for a task is another issue which determines whether optimality of that ANN is reached. Another thing to note is that even if the neural network solutions used tends to converge, this may not result in a unique solution [34]. ANN design include specification of the number of hidden layers and the number of units in these layers. As good a starting point is to use hidden layer, with the number of units equal to half the sum of the number of input and output units. The number of input and output units is defined by the problem.

Generally, properly determining the size of the hidden layer is a problem, because an underestimate of the number of neurons can lead to poor generalization capabilities [35], while excessive nodes can result in over-fitting and eventually make the search for the global optimum more difficult. The increased number of parameters increases the time complexity of the learning algorithm.[12]. Select an initial configuration (typically, one hidden layer with the number of hidden units set to half the sum of the number of input and output units) then change accordingly [36].

3.2.4 Support Vector Machine

SVM is supervised machine learning algorithm that learn a model from fully annotated data and then evaluate the model using test data. It is a method for classification of both linear and nonlinear data. It uses a nonlinear mapping to transform the original training data into a higher dimension, where it finds a hyperplane that separates the data by class using essential training tuples called support vectors [27]. SVM gained popularity due to many promising features such as better empirical performance. SVM was the first proposed kernel based algorithm [27]. It uses a kernel function to transform data from input space into a high dimensional feature space in which it searches for a separating hyperplane. SVM is a technique suitable for binary classification tasks. SVM were developed to solve the classification problem, but recently they have been extended to solve regression problems by introduction of an alternative loss function [26]. In general, it is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data.

SVM provide a new approach to the problem of pattern recognition with clear connections to the underlying statistical learning theory. They differ radically from comparable approaches such as neural networks, in that, it always finds a global minimum [26], and its simple geometric interpretation provides fertile ground for further investigation. Although the training time of even the fastest SVM can be extremely slow, they are highly accurate, can model complex nonlinear decision boundaries. They are much less prone to over-fitting than other methods [27]. Afterwards, the SVM will be made non-linear and non-parametric by introducing a kernel. As explained further, it is this characteristic that makes SVM a useful tool.

A *Margin Maximization*

In SVM, the objective is to classify the data points with a hyperplane that has the maximum distance to the nearest data point on each side and extend this to non-linear boundaries using kernel trick [26]. Subsequently, such a linear classifier is also called the maximum margin classifier. There are many classifiers (hyper planes) that separate the data. However only one of these achieves maximum separation. An SVM analysis finds the line (or, in general, hyper plane) that is oriented so that the margin between the support vectors is maximized. The reason we need it is because if we use a hyperplane to classify, this plan may be a local classifier, thus we see the concept of maximum margin classifier [26].

So the best hyperplane is chosen from several possibilities according to some optimization criteria (typically training set performance). SVM algorithms find the function (hyperplane) that returns the largest minimum distance to the examples. This distance is called a margin, and the examples closest to the margins are then termed *support vectors*. In Figure 3.4, the points Laid on the margin lines are *support vectors*, and the distance between these margin lines is width of the margin. Because the solution depends only on the support vectors, the remaining examples are not important in developing the model.

As shown in Figure 3.4, any hyperplane can be written as the set of points X satisfying $w^T X + b = 0$, where the vector w is a normal vector perpendicular to the hyperplane and b is the offset of the hyperplane $w^T X + b = 0$ from the original point along the direction of w. Given labels of data points X for two classes(class 1 and class 2), we present the labels as $y_i \in \{1, -1\}$. Meanwhile, given a pair of $\left(w^T, b\right)$, we classify data X into class 1 or class 2 according the sign

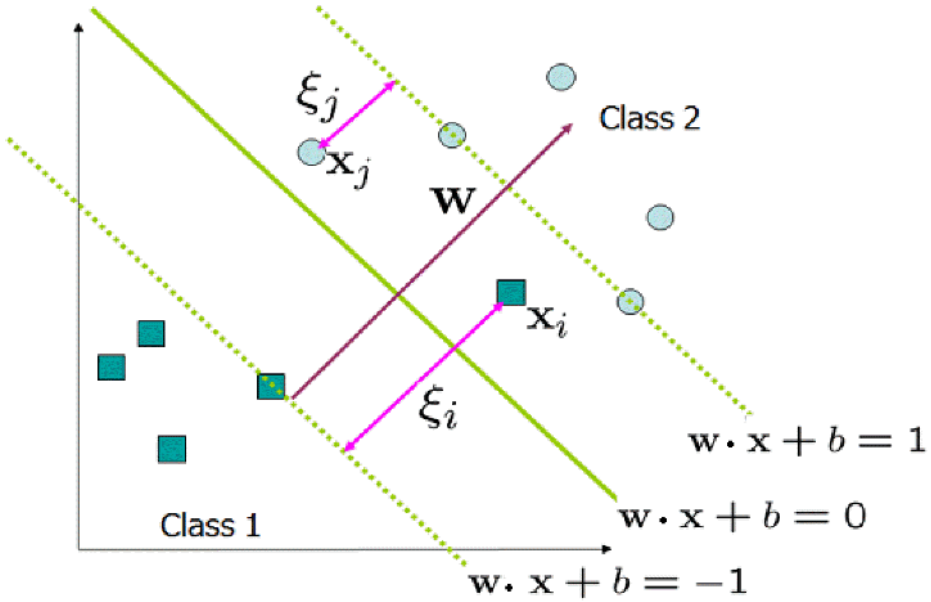


Figure 3.4: SVM Classification [23]

of the function $f(X) = \sin(w^T X + b)$. Thus, the linear separability of the data X in these two classes can be expressed as Equation 3.2 and Equation 3.3

$$\mathbf{x}_i.\mathbf{w} + b \geq +1, \quad for\ y_i = +1 \tag{3.2}$$

$$\mathbf{x}_i.\mathbf{w} + b \leq -1, \quad for\ y_i = -1 \tag{3.3}$$

Equation 3.2 and Equation 3.3 can be combined to form: Equation 3.4

$$\mathbf{y}_i(x_i.\mathbf{w} + b) \geq 1 \tag{3.4}$$

In addition, the distance from data point to the separator hyperplane $w^T x + b = 0$ can be computed as $r = (w^T x + b)/||w||$, and the data points closest to the hyperplane are called support vectors. As denoted in Figure 3.4, the distance between support vectors is called the margin of the separator, which is simply $2/||w||$. Linear SVM is solved by formulating the quadratic optimization problem as in Equation 3.5

$$\begin{aligned} &\underset{w,b}{Minimize}\ (\frac{1}{2}||w||^2) \\ &subject\ to\quad y(w^T x + b) \geq 1 \end{aligned} \tag{3.5}$$

In the case of linearly separable data, once the optimum separating hyperplane is found, data points that lie on its margin are support vector points and the solution is represented as a linear combination of only these points, other data points are ignored. Therefore, the model complexity of an SVM is unaffected by the number of features encountered in the training data (the number of support vectors selected by the SVM learning algorithm is usually small). For

this reason, SVMs are well suited to deal with learning tasks where the number of features is large with respect to the number of training instances.

Searching for the optimal hyperplane in Equation 3.5 is a quadratic optimization problem, which can be solved by constructing a Lagrangian.

$$\mathbf{L}_p = \frac{1}{2}||w||^2 + \sum_{i=1}^{n} \alpha i yi(xi.w - b) + \sum_{i=1}^{n} \alpha i \tag{3.6}$$

Most real-world problems involve non-separable data for which no hyperplane exists that successfully separates instances in the training set. One solution to the inseparability problem is to map the data to a higher dimensional space and define a separating hyperplane. This higher-dimensional space is called the transformed feature space. A linear separation in transformed feature space corresponds to a non-linear separation in the original input space. Mapping the data to some other Hilbert space H as $\Phi : \mathbb{R}^d \longrightarrow \mathcal{H}$ Then the training algorithm would only depend on the data through dot products in H, that is on functions of the form $\Phi(x_i).\Phi(x_j)$ if there were a kernel function K such that $K(x_i, x_j) = \Phi(x_i).\Phi(x_j)$, we would only need to use K in the training algorithm, and would never need to explicitly determine ϕ . Thus, kernels are a special class of function that allow inner products to be calculated directly in feature space, without performing the mapping. Once a hyperplane has been created, the kernel function is used to map new points into the feature space for classification.

Using kernel functions, nonlinear SVM is formulated into the same problem as linear SVM by mapping the original feature space to a higher-dimensional feature space where the training set is separable by using kernel functions. Nonlinear SVM is solved by using a soft margin to separate classes or by adding slack variables ξ_i, as shown in Equation 3.7 and Equation 3.8 for the positive and negative classes.

$$\mathbf{x}_i\mathbf{w} + b \geq +1 - \xi_i, \quad for\ y_i = +1 \tag{3.7}$$

$$\mathbf{x}_i\mathbf{w} + b \leq -1 + \xi_i, \quad for\ y_i = -1 \tag{3.8}$$

$$\xi_i \geq 0\ , \quad \forall_i \tag{3.9}$$

Equation 3.10 is generated combining Equation 3.7, Equation 3.8 and Equation 3.9.

$$\begin{aligned} &\underset{w,b,\xi}{minimize}\ \ (\frac{1}{2}||w||^2) \\ &subject\ to\ \ y_i(w^T\Phi(x_i) + b) \geq 1 - \xi_i \\ &\qquad\qquad \xi_i \geq 0 \end{aligned} \tag{3.10}$$

Thus, for an error to occur the corresponding ξ_i must exceed unity, so $\sum \xi i$ is an upper bound on the number of training errors. In this case the Lagrangian is:

$$\mathbf{L}_p = \frac{1}{2}\|w\|^2 + c\sum_i \xi i - \sum_i \alpha i \{yi(xi.w - b) - 1 + \xi i\} - \sum_i \mu i \xi i \tag{3.11}$$

Where the μ_i are the Lagrange multipliers introduced to enforce positivity of the ξ_i and C is a tuning parameter. Tuning this parameter can balance between the margin maximization and the classification violation. Searching for the optimal hyperplane in Equation 3.10, which can be solved by constructing a Lagrangian and transformed into the dual form and one kernel is used for the points in the feature space.

It is common practice to estimate a range of potential settings and use cross validation over the training set to find the best kernel. For this reason a limitation of SVMs is the low speed of the training. Selecting kernel settings can be regarded in a similar way to choosing the number of hidden nodes in a neural network. As long as the kernel function is legitimate, a SVM will operate correctly even if the designer does not know exactly what features of the training data are being used in the kernel-induced transformed feature space [26]. Some popular kernel functions are the following:

- linear Kernel $K(\mathbf{x}, \mathbf{y}) = \mathbf{x}.\mathbf{y}$
- Gaussian Kernel $K(\mathbf{x}, \mathbf{y}) = e^{\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}}$
- Polynomial Kernel $K(\mathbf{x}, \mathbf{y}) = (\gamma \mathbf{x}.\mathbf{y} + r)^d$
- radial basis function (RBF) $K(\mathbf{x}, \mathbf{y}) = e^{(-\gamma\|\mathbf{x}-\mathbf{y}\|^2)}$
- Sigmoid Kernel $K(\mathbf{x}, \mathbf{y}) = \tanh(\gamma \mathbf{x}.\mathbf{y} - r)$

SVM is better than ANN for achieving global optimization and controlling the over-fitting problem by selecting suitable support vectors for classification. SVM can find linear, nonlinear, and complex classification boundaries accurately, even with a small training sample size. However, selecting kernel functions and fine-tuning the corresponding parameters using SVM are still trial-and-error procedures [37].

One of the major problems in the study of SVM is kernel selection, that's based necessarily on the problem of deciding a kernel function for a particular task and dataset [26].

SVM can be a useful tool for Fraud detection, in the case of non-regularity in the data, for example when the data are not regularly distributed or have an unknown distribution. SVM deliver

a unique solution, since the optimality problem is convex. This is an advantage compared to Neural Networks, which have multiple solutions associated with local minima.

4

DATA PREPARATION

A central problem in machine learning is extracting a representative set of features from which to construct a classification model for a particular task. Recent research has shown machine learning algorithms affected by irrelevant and redundant training information [27]. If there is too much irrelevant and redundant information present or the data is noisy and unreliable, then learning during the training phase became more difficult. Data Preparation involves processing the raw data so that machine learning algorithms can produce a structural description of the information that is implicit in the data. It defines, process and makes suitable to a data mining technique. It is the first important step in data mining and plays a decisive role in the entire process.

The objective of this thesis is to detect SIM-Box fraud using historical CDR and additional subscriber profiles to identify anomalies behavior of fraudulent subscribers. The collected large amounts of CDR need to be organized to form patterns and scenarios of normal usage and fraud situations. Real-world data are typically noisy, massive in volume, and may originate from a bulk of heterogeneous sources, hence, knowing the data deeply is a vital prerequisite for data preparation [27]. Prior to modeling and evaluation,raw data should be prepared for the experiment. Data preparation involves having a closer look at fields and data values, knowing what the data implies, identifying the fields that make the data and type of values they contain and their behavior, to get a better sense of it. This helps in fixing inconsistencies incurred during data integration, identifying existence of outliers and extremes, extract similarity and deferences of data objects with respect to others.

For this particular thesis, subscriber profile and CDR data of mobile subscribers, served from ethio telecom billing system, were collected and prepared for the experiment. CDR generated from the usage of active customers for a month period has been collected, stored in a database, processed and outputted in a format that the experiment required.

Whenever a subscriber makes a call over the operator's network a toll ticket is prepared which contains complete information of the call, including subscriber id, called number, duration of the call, call start and end time, destination location and so on. This detail information about

the call is named as CDR. *CDR* is produced by telephone switches or other telecommunication equipments on call basis and contain all the information to describe the important characteristics of telephone call and other telecommunication transaction be it from or to a subscriber in the network. It contains the fields necessary for billing systems to rate a particular call and bill the subscriber. CDRs give the details of each voice, SMS and Internet data usage transaction, originating from and terminating on a subscriber's device. A *Field* represents a characteristic or feature of CDR instance. It may include, telephone numbers involved in the call, date and time of the call, duration of the call, identification of the cell transmitting the call to the subscriber's telephone and more fields. For data preprocessing to be successful, it is essential to dig deeper to understand the meaning of the records in order to have an overall picture of the data.

The following sections of this chapter discussed in depth the data processing i. e. collecting, understanding and preparing the data to the intended model building. The first part of this chapter focuses on data collection process, methods and techniques in simplifying the challenges related to transferring and storing of the raw data. Then after the next part is understanding and describing the data collected. Following this process selecting the relevant data for the experiment is performed. Finally, the data preprocessing is executed.

4.1 DATA COLLECTION

After the approval of data access request from the company, major data source, CDR data collection started. Due to the bulkiness of the record and resource limitation, storing all CDR for an extended period is challenging. Moreover, storing these records in a separate machine simplifies the data preparation process and guaranties the safety of the business operations. Taking this in to consideration, one month CDR stored in a dedicated server allocated for this thesis work.

CDR dump files in a text file format, snapshot of a file shown in Figure 4.1, are pushed every five minutes to the server. These revived files have been imported to a database prepared in advance in this server via automated data loading tool. This process continues for a month starting from April 26, 2018 until May 25, 2018. This time range is selected considering the access permission approval and readiness of the allocated server. Collecting one month data is decided upon the storage and computational resource capacity of the server, as well as sufficiency of the data for the intended experiment. Concurrently the data imported to the database is segregated based on call start date and stored in deferent tables, where each table contains one day records. Since

each day contains approximately about 150 Million records and the entire month records are about 4.5 Billion generated from about 25 million active mobile subscribers, separating records per day simplifies data manipulation process.

In addition a sample of 5,000 fraudulent subscriber service numbers were provided by the fraud management section. These fraudulent numbers were proven and blocked their access to the network due to their fraudulent activities in the specified time period. These sample fraudulent subscriber numbers have been imported in to the database as well.

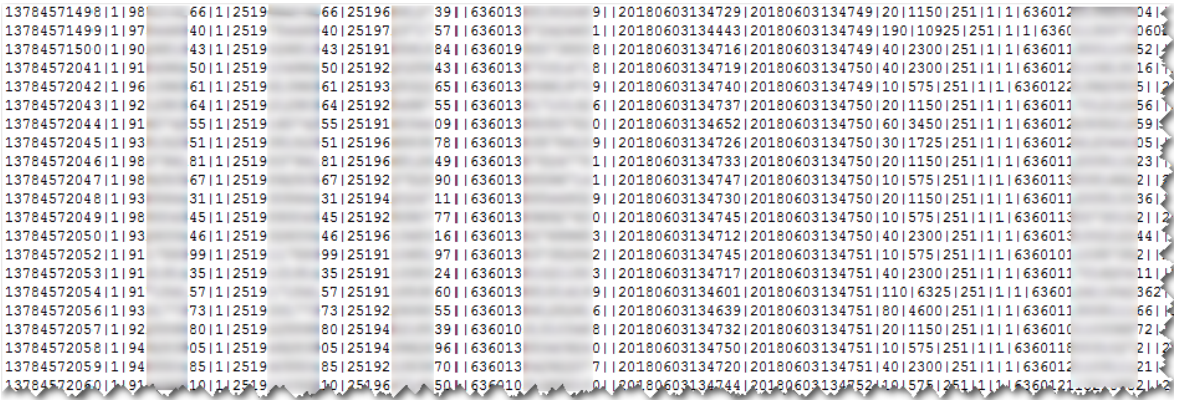


Figure 4.1: Snapshot of Raw CDR Dump File Section.

4.2 UNDERSTANDING THE DATA

Understanding the target data is primary task for mining the knowledge within it. This step includes activities such as identifying fields, examining the values they contain and evaluating their importance for this research. In this process careful analysis of data and its structure is done together with domain experts. Relationships of the data with the problem at hand and particular Data-mining tools were evaluated. The dataset acquired in this thesis is collected from ethio telecom, the telecommunication service provider in Ethiopia. It consists CDR and customer profile data.

The collected CDR as shown in Table 4.1, contains a total of 33 fields. Some fields are without values like Calling IMEI others contain duplicate values like Billing Number and Calling number. Most of them are generated for billing purpose like CHARGE, Call Fee, Account Item ID, Rate ID, Billing Date, Billing Offering ID and Billing Cycle ID. Upload traffic and Download traffic contains the Internet usage. CDR_ID uniquly identifies each CDR and RE_ID used to differentiate Voice, SMS and Internet Data usage records. CDR_TYPE included for distinguishing Mobile Originating ,Terminating or Forwarding call types. CELL_A and CELL_B as well contains the

ID of Calling and Called district or Cell. Some sensitive fields such as called, calling or Billing numbers have been hashed for privacy reasons.

Table 4.1: CDR Fields Description

NO	Field Name	Description
1	CDR_ID	CDR Sequence Number
2	RE_ID	CDR type ID for voice, SMS and Data
3	BILLING_NBR	Billing Number
4	CDR_TYPE	Call type Id
5	CALLING_NBR	Calling Number
6	CALLED_NBR	Called Number
7	CALLING_IMEI	Calling IMEI
8	CALLING_IMSI	Calling IMSI
9	THIRD_NBR	Third Party Number
10	START_TIME	Call start time
11	END_TIME	Call end time
12	DURATION	Call duration
13	CALL_FEE	Call fee
14	CALLED_COUNTRY	Called country
15	CALLING_CARRIER	Calling carrier
16	CALLED_CARRIER	Called carrier
17	CELL_A	Calling district
18	CELL_B	Called district
19	STATE_DATE	Billing date
20	CALLING_SUB_ID	Calling subscriber ID
21	BILLING_CYCLE_ID	Billing cycle ID
22	CHARGE1	Charge amount
23	CHARGE2	Charge amount
24	PRICE_ID1	Rate ID

Continued on next page

Table 4.1 – Continued from previous page

NO	Field Name	Description
25	ACCT_ITEM_ID1	Account item ID
26	TRAFFIC_UP	Upload traffic
27	TRAFFIC_DOWN	Download traffic
28	BILLING_OFFERING_ID	Billing offering ID
29	ERROR_CDR_TYPE	Error CDR Indicator
30	CALL_FORWARD_INDICATOR	Call Forward Indicator
31	HOT_LINE_INDICATOR	Hot Line Indicator (voice mail)
32	CALLING_TRUNK_ID	Calling Trunk ID
33	CALLED_TRUNK_ID	Called Trunk ID

4.3 DATA SELECTION

As ML aims to address larger, more complex tasks, focusing on the most relevant information in a possibly sufficient quantity has become higher concern. Unnecessary data should be removed from the collected data in order to provide relevant information to the ML algorithms. In order to meet this requirement of the ML process, as the clarity of input dataset has significant factor on the output, data selection needs to be accomplished with higher attention.

Data selection is a process, which requires domain knowledge to choose useful features that capture the variability and essentiality of the data for the target ML algorithm to learn patterns from the data successfully. In addition, it has a vital role in reducing complexity of learning process and increase fraud detection effectiveness. The behavior of SIM-Box fraud discussed in detail in Section 2.2.3 is an input for this selection.

4.3.1 *Field Selection*

As discussed in detail in Section 4.2 the collected CDR has fields with empty and duplicate contents, and some others are irrelevant for the intended thesis work. Among the 33 fields eight fields that are considered the most important for the study are selected. Those selected fields are described in Table 4.2. Moreover, since SIM-Box fraud is relay on mobile phone Subscribers, CDRs of only mobile phone subscribers are considered.

Table 4.2: Selected CDR Fields Description

No	Field	Description
1	Calling Number	Call originating subscriber number
2	Called Number	Call receiving subscriber number
3	Call Start Time	Date and time where call is started
4	Call End Time	Date and time where call is ended
5	Call Duration	Total time from call start to end
6	Cell_A	The caller cell ID
7	Traffic_up	Amount of Uploaded traffic
8	Traffic_down	Amount of Downloaded traffic

4.3.2 Sampling

Sampling is an important step in many practical data mining applications, and is often used in handling problems with large data. Considering all the collected records in the dataset is Challenging. Mining a database of big data is a laborious task, and requires either advanced parallelized hardware and ML algorithms, or the use of sampling to reduce size of the data to be considered. In addition, since the provided fraudulent subscriber numbers are much fewer than the target subscribers, which results unbalanced dataset proportionality of normal and fraud classes. Usually, the classification algorithms exhibit poor performance while dealing with unbalanced datasets and results bias towards majority class. In the contrary we need a fairly high prediction for the minority class.

It is agreed that proper sampling is important in ML [34]. Specially for fraud detection problems, there are many more legitimate than fraudulent samples. Hence, appropriate classification approaches are needed to classify the unbalanced data. There have been several sampling approaches for coping with unbalanced datasets. Such as selective under-sampling of majority class by keeping minority classes fixed [34]. Decisions about how large of a sample to use must be made rationally. Consulting literature [2, 8, 14, 15] and domain experts, We decide to incorporate in the dataset a proportionality of 75% normal and the remaining 25% fraudulent examples.

Since the supplied fraudulent numbers are 5,000, we included 15,000 normal numbers. totaly we used 20,000 subscriber numbers for this thesis. The included normal numbers are selected using random sampling form about 24.5 million target mobile subscriber numbers. About 14.5 million records generated by the sampled subscribers is applied in this thesis. Detail of the sampled data is tabulated in Table 4.3.

Table 4.3: Sampled Subscriber Number and CDR

	Fraud	**Normal**	**Total**
Subscriber	5,000	15,000	20,000
Records	2,941,199	11,529,383	14,470,582

4.4 DATA PREPROCESSING

Raw data are generally incomplete, noisy, and inconsistent, it requires some work to make it relevant for ML. One of the important stages in making ready the raw data to ML is data preprocessing. This might entail sourcing some additional data, cleaning up the data, dealing with missing values in the data, transforming the data, and analyzing the data to raise its efficiency through a better choice of variables.

There are important processes that can improve success when applying machine learning techniques to practical data mining problems. Preprocessing may include model building activities, because many preprocessing tools build model of the data to transform it. In fact, data preparation and modeling usually works together for same result. It is necessary to iterate results obtained from modeling that affect the choice of preprocessing techniques. Upcoming sections discuss in detail this process.

4.4.1 *Data Cleaning*

Data cleaning is to fill the vacancy value of the data, eliminate the noise data and correct inconsistencies in the data. It is time-consuming and labor-intensive procedure, but one that is absolutely necessary for successful data mining. It needs domain experts support and understanding the data in depth. It is used for making sure that the data is free from different errors. Applying tools that show the distribution of values of fields are very helpful in cleaning data. These tools simplifyes identifing missing values, outliers, and errors in the data.

When collecting data, it is not possible to ensure it is perfectly collected, there will be errors in the data, we need to address data quality issues. The collected data for this thesis, as any big data collection, have some errors, such as incomplete values, missing value, duplicate records and so on. In cleaning this data, the process started by selecting only calls generated by mobile devices, records of other than mobile subscriber numbers are removed. Records which contain any missing value in any of their fields similarly excluded from the target data. Records which contains incomplete or invalid values (such as Calling Number length different from 12 characters) are removed. Records which missed country code (251) prefix are modified by concatenating a prifix to those records. Duplicate records are removed by keeping only a single instance of the duplicates. In addition quality and validity of the target data is checked in accordance to the intended machine learning techniques.

4.4.2 Data Aggregation

Another practical question when preprocessing the collected data is, the granularity level of data aggregation. applying row CDR data in the dataset is useless for the study of SIM-Box fraud detection. The data must be aggregated to the subscriber level. But need to consider the appropriate granularity level. Selecting the right level of aggregation is usually critical for success. When considering CDR for SIM-Box fraud detection, it is important to understand with respect to pattern analysis, the time span in which the data should be aggregated over. making too narrow the granularity level of the time span, suitable patterns for fraud detection may not be able to noticed. While too broad granularity level have the potential to identify the fraud, but it might be too late to take preventative action based on the outputs. In either case the fraud detection tool would be considered ineffective. Researches agreed that accumulated characteristics of a user yield better discrimination results. However, aggregating for longer period is not advisable in order to attain some level of real-time detection ability [14].

To test appropriateness of the proposed granularities levels, In this thesis work, we derived the data into three levels of granularity i. e. 4 hour, daily, and monthly, which can aggregate user behavior to give better fraud identification capability as well as preserve some level of near to real time detection capability. In order to prepare the input data for the intended machine learning algorithms, the following steps have been taken on the sampled records discussed in detail in Section 4.3. To generate a derived aggregated attributes which are described in Table 4.4 in a Month, a Day and a 4 Hour span of time, we folowed the folowing steps.

1. The selected CDRs are aggregated in subscriber level.

2. The aggregated output of voice call, SMS, Internet data and Subscription age are integrated to form single instance per subscriber per aggregation level.

3. Class label field that would be usable in training the machine to build model has been added.

Table 4.4: Derived Attributes Description

Attribute	Description
TOT_OUT	Number of outgoing calls originated from the subscriber
DIS_OUT	Number of unique subscribers called
TOT_IN	Number of Incoming calls terminated in the subscriber
TOT_SMS	Number of text messages initiated from the subscriber
TOT_DATA	Sum of Data usage of the subscriber
CELL_OUT	Number of distinct cells Accessed by the subscriber
TIME_GAP	Average time gap between calls of the subscriber
OUT_RATE	Ratio of DIS_OUT to TOT_OUT
TOT_RATE	Ratio of TOT_IN to TOT_OUT
DATA_RATE	Ratio of TOT_DATA to TOT_OUT
CELL_RATE	Ratio of CELL_OUT to TOT_OUT
SUBS_AGE	Subscription Age of the subscriber

4.4.3 Data Integration

Before beginning work on ML Process, it is necessary to bring all the data together into an instances. Integrating data from different sources usually presents many challenges. These various sources may have dissimilar data storing formats. As discussed in Section 4.4.2 in detail the selected data is aggregated to the format of the derived attributes described in Table 4.4. However, voice outgoing call, SMS, Internet data usage, voice incoming calls and subscription age records of the selected fraudulent and non-fraudulent subscriber samples are stored in deferent tables. So, to bring all these records in to a single instance record, that is 4 hour dataset or daily dataset or monthly dataset of the subscribers, it should be integrated in to a single table based on the unique value of the records. This integration is shown in Figure 4.2. As described

all the records are integrated on the unique value (CALLING_NUM) except the voice incoming which is (CALLED_NUM).

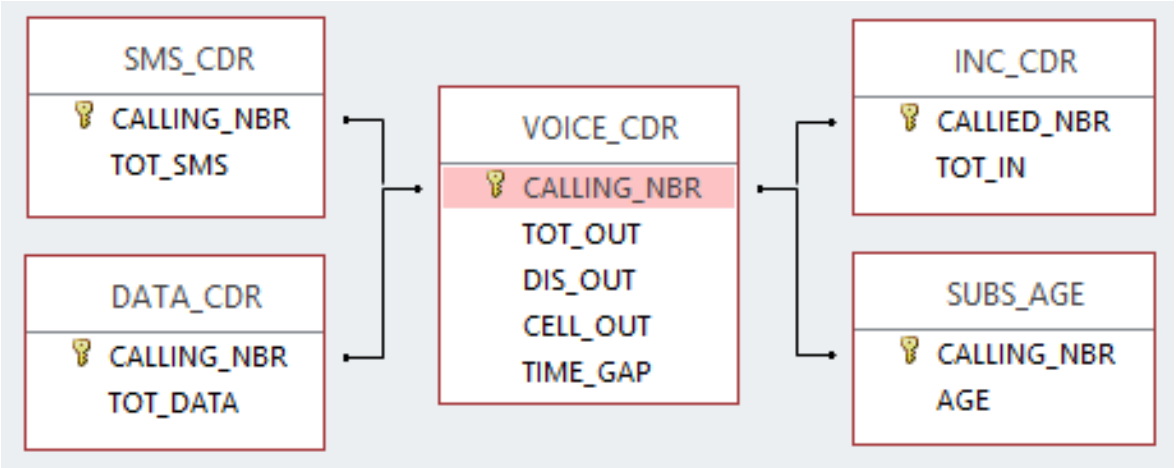


Figure 4.2: Integration of Aggregated Datasets

4.4.4 *Feature Selection*

Theoretically, dataset with more attributes in learning process gives better result. However, in practice this may not be always the case [38]. Most machine learning algorithms are designed to learn appropriate features to use for making their decisions. Adding distracting features often confuses machine learning systems. In practical situations there are many attributes for learning process, some of them feasibly significant, and some are irrelevant or redundant. The problem is identifying a representative set of features from which to construct a classification model. For that reason, the dataset must be preprocessed to select useful attributes. Even though, many learning schemes can select features appropriately and ignore irrelevant ones, but in practice their performance might be affected. Because of the negative effect of irrelevant attributes on most ML algorithms, it is common to precede learning with an attribute selection. More importantly, dimensionality reduction yields a more compact, easily interpretable representation of the target concept, focusing attention on the most relevant features[24].

Feature selection is a process of selecting subset of features from which to build a predictive model or classifier [38]. This is usually done for model simplification and increased interpretability, reducing training times and computational cost, and to help reduce the risk of over-fitting, and thus improve model generalization.

Approaches in feature selection includes

Filter method: it makes an independent assessment based on general characteristics of the

data, attributes filtered to produce the most promising subset.

Wrapper method: to evaluate feature subset using the machine learning algorithm that will be employed for learning. Making an independent assessment of an attribute subset would be easy if there were a good way of determining when an attribute was relevant. However, there is no universally accepted measure of relevance, although several have been proposed [38].

In this thesis work we applied Correlation based Feature Selection (CFS). It evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. The central idea here is that, subsets of features which are highly correlated with the class while having low inter-correlation are preferred. However, this method have limitation of selecting features that have locally predictive values when they are overshadowed by strong, globally predictive features [38]. While a single feature may account for only a very small proportion of a dataset, a number of such features may cumulatively cover a significant proportion of the dataset. Having this in to consideration, three subsets of features are proposed. In this preprocessing stage we have tested the three proposed feature subsets in small sample datasets. Out of which the subset contains all the proposed features performed better in all the sample datasets. Then the subset containing the entire features were taken for this experiment. The designed subsets of futures are listed in Table A.2.

4.4.5 Dataset Formatting

In a typical supervised machine learning task, data is represented as a table of examples or instances. Each instance is described by a fixed number of measurements, or features, along with a label that denotes its class. Formatting is a process re-engineering the input dataset into a formats that is acceptable by the particular ML algorithm. Commonly features or attributes are of nominal or numeric data type, attaining format consistency in all of the records in the entire file is critical issue, inconsistency in format of records may create problem in model building. In our case, all the records used are of numeric format except the class level which is of nominal, and consistency of format is checked carefully in the entire document, specialty on aggregating the raw data to derived attributes.

A classification task usually involves separating data into training and testing sets. The set of training examples are used to produce the learned concept descriptions, and the separate set of test examples are needed to evaluate the accuracy of the build model. Similarly in this thesis the preprocessed dataset was partitioned in to two parts, training and testing, as tabulated in Table 4.5. About 66% of the dataset instances are used for training and the remaining for testing.

The refined training and test dataset is saved in Attribute Relationship file format (ARFF), which is an ASCII file format. ARFF is a coma separated values file with a header that describes the meta-data of the attributes, it is memory efficient and faster than csv. A sample of preprocessed dataset in ARFF type is shown in Figure A.1

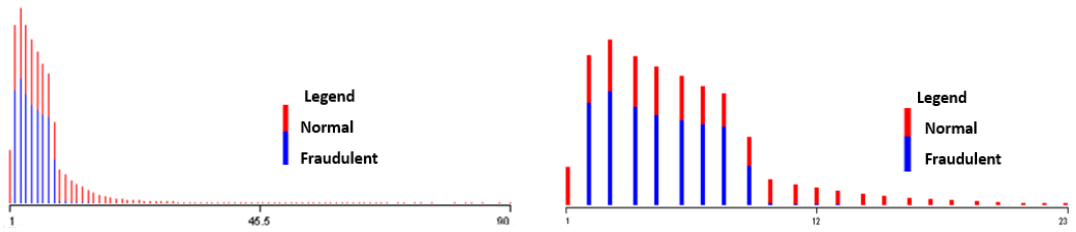
Table 4.5: Summary of Formated Dataset

	Training Dataset				Testing Dataset			
Aggregation	**Normal**	**Fraud**	**Total**	**Normal%**	**Normal**	**Fraud**	**Total**	**Normal%**
4 hour	149984	49770	199754	75.1	46998	16098	63096	74.5
Daily	149465	49842	199307	75.0	46305	16848	63153	73.3
Monthly	14964	4988	19952	75.0	4758	1894	6652	71.5

4.4.6 Removing Outliers

An outliers is an observation with values of variables that are quite different from most other observations. Typically, an outlier appears at the maximum or minimum end of a variable that distorts the distribution. Identifying whether an observation is an outlier is quite difficult, the decision of an outlier vary with application as it depends on the context and the model to be built. Perhaps under one context an observation is an outlier but under another context it might be a typical observation. When summarizing the data, performing tests on the data, and in building models, outliers can have an adverse impact on the quality of the results.

In general outlier detection algorithms include those that are based on distance, density, projections, or distributions. Distance based approaches are common in data mining, where an outlier is identified based on an observation's distance from nearby observations [23, 24]. The most common data dispersion measures are range, quartiles, interquartile range, box plots, variance and standard deviation of the data, these measures are useful for identifying outliers.



(a) Before Outliers Removed (b) After Outliers Removed

Figure 4.3: Snapshot of Incoming Call Attribute

One approach to dealing with outliers is to remove them from the dataset. However, identifying the outlier remains an issue. In this thesis we applied Interquartile Range (IQR) outliers identification and removing tool. Figure 4.3 shows a snapshot of incoming calls attribute, part (a) before outliers removed and (b) after outliers removed. similarly, All the attributes were examined for outliers and removed the identified outliers.

5

EXPERIMENTATION

In this chapter, we discussed the model building processes followed using selected algorithms for classification of SIM-Box fraud. It focused on developing models via the proposed algorithms and prepared datasets, evaluating the developed models applying various performance measures, and discussed on the outcomes of the experiment and proposed the best model in detection of SIM-Box fraud.

5.0.1 Experiment Methods

The dataset applied in this experiment, as thoroughly discussed in Chapter 4, were prepared devoting high care, and similarly the ML algorithms are carefully chosen based on their generalization capability and their detection performance. Putting into consideration the behavior of SIM-Box fraud and the available CDR data for training and testing ML algorithms, supervised classification techniques specifically, ANN, SVM and RF were used. These ML techniques are discussed exhaustively in Chapter 3.

For training the proposed ML algorithms two techniques were suggested. Explicitly, percent Split and cross validation were used in training the models.

K-Fold Cross-Validation method: In this method, the dataset is divided into mutually exclusive and equal-sized K subsets, then the classifier trained k times on the union of $K - 1$ subsets and tested on the k^{th} subset. This is repeated iteratively changing the test subset from the first to the k^{th} subset, to get a distribution of the test error of the model. The average error rate of each subset is therefore an estimate error rate of the classifier [27]. K-Fold Cross-Validation is used to achieve an unbiased estimate of the model performance. 10-fold cross-validation is the most common cross-validation technique used for medium sized dataset.

Percent Split method: In this technique the dataset splite into two parts in the model building session the first part for training and the remaining for testing. The common ratio is 66% for training and 34% for testing.

5.0.2 *Performance Assessment Measures*

In the field of machine learning, there exist certain standard evaluation tools which are commonly used to describe different aspects of the developed model. The most common tools suggested for this experiment includes accuracy, precision, recall, F-measure, Root Mean Squared Error and Receiver Operating Characteristic (ROC). The raw data produced by a classification scheme during testing are counts of the correct and incorrect classifications from each class. This information has been revealed in a confusion matrix. A Confusion Matrix is a form of contingency table showing the comparison between the true and predicted outcomes for a sets of labeled examples. All the evaluation measures depends on results of the Confusion Matrix. It shows the overall performance of a classifier. In Table 5.1 structure of the confusion matrix is described. The rows described the actual value as positive and negative classes, whereas the columns expressed the predicted values in evaluation.

Table 5.1: Confusion Matrix Values Mapping

		Predicted	
		Positive	**Negative**
Actual	**Positive**	TP	FN
	Negative	FP	TN

Description of the Four values of the Confusion Matrix are listed below:

- **True Positive (TP)**: Actual SIM-Box cases that were classified as SIM-Box cases.
- **False Positive (FP)**: Normal customers that were classified as SIM-Box cases.
- **False Negative (FN)**: Actual SIM-Box cases that were classified as normal cases.
- **True Negative (TN)**: Normal cases that were classified as normal cases.

TP and *TN* values of the confusion matrix are the correctly predicted results of the classifier, where as the values *FP* and *FN* are incorrectly predicted results. The suggested Performance Assessment Measures were discussed in detail as follows.

- **Accuracy:** is the most common form of performance metrics especially in machine learning. Although it tells about each class, it may fail in unbalanced class situations as in fraud detection. Formally, it is defined as the ability to differentiate the suspicious and legiti-

mate calls correctly. It describes the ratio of correct classification. In this thesis, classification accuracy is the primary evaluation criterion for evaluating experiments. Accuracy is calculate as the proportion of TP and TN to all evaluated instance. Mathematically, this can be stated as in Equation 5.1.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5.1}$$

- **Precision** indicates the rate with which fraction of those predicted positive are actually positive. It is generally not so descriptive of a performance alone. Because it only focuses on the positive instances, it informs nothing about the negative instances. Since there is usually a significant trade-off between precision and recall, it is more important when used in comparative to recall. Precision described mathematically as in Equation 5.2.

$$Precision = \frac{TP}{TP + FP} \tag{5.2}$$

- **Recall** also known as Sensitivity or True Positive Rate, measures the proportion of actual positives which are correctly identified as positive by the classifier. Often it is considered together with precision since there exists a visible trade off between them. Recall expressed mathematically as in Equation 5.3

$$Recall = \frac{TP}{TP + FN} \tag{5.3}$$

- **F-measure** A measure that combines precision and recall. It is the harmonic mean of precision and recall. It can be expressed mathematical as in Equation 5.4

$$F - measure = 2 \cdot \frac{\mathrm{Pr}\,esision \cdot Recall}{\mathrm{Pr}\,esision + Recall} \tag{5.4}$$

- **Root Mean Squared Error (RMSE)** is a quadratic scoring rule which measures the average magnitude of the error. It is the average of the squares of the difference between forecast and corresponding observed values,and the square root of the average is taken. Since the errors are squared before they are averaged, it gives a relatively high weight to large errors. This means it is most useful when large errors are particularly undesirable. RMSE expressed mathematical as in Equation 5.5

$$RMSE = \sqrt{\frac{1}{n}\sum_i \varepsilon_i^2} \tag{5.5}$$

- ROC curve is a graph of FP Rate versus TP Rate. The area measures the ability of the classifier to correctly classify the test data. It shows performance of models across all

possible thresholds. A model that cover larger area in the plot have better classification ability.

5.1 MODEL BUILDING

In this stage, different models were trained applying varieties of training modes and datasets. Experiments are conducted on the selected three algorithms, prepared three datasets, training methods applied, to come up with the best predictive model for SIM-Box fraud detection. All possible combination of settings were experimented and as a result, a total of 18 models were built. The statistics of models built per each algorithm, dataset and training modes are recorded in Table 5.2. Detailed performance measures of all build models were tabulated in Table A.1. The combination of settings experimented includes, three selected algorithms (RF, ANN and SVM were discussed in Chapter 3), for each algorithm three datasets (4, hour, daily and monthly), discussed in Chapter 4.

Table 5.2: Summary of Built Models Statistics

Algorithm	Training Mode	No of Modle			
		4 Hour	Day	Month	Total
RF	Cross-Validation	1	1	1	3
	Percentage Split	1	1	1	3
ANN	Cross-Validation	1	1	1	3
	Percentage Split	1	1	1	3
SVM	Cross-Validation	1	1	1	3
	Percentage Split	1	1	1	3
Total		6	6	6	18

The upcoming sections addressed model building process of the proposed algorithms. In Section 5.1.1 building of models applying RF Algorithm were discussed in detail, Section 5.1.2 focused on models built using the ANN algorithm and Section 5.1.3 discussed models trained based on the SVM algorithm. In each section some best classification models were selected and documented their detail performance measures for evaluation.

5.1.1 *Building RF Models*

As it is thoroughly discussed in Section 3.2, RF is best suited for classification problems. This experiment conducted building models on RF algorithm. We devised to experiment the classifier in two training options, percent Split and cross validation. To train it with the ready three datasets(4 hour, daily and monthly aggregated). Consuming all these impute combination 6 RF models were built. Among them 2 are built using 4 hour Aggregated dataset, the other 2 built using daily aggregated dataset and the remaining 2 using monthly aggregates dataset,(see Table 5.2). This process were explained in Section A, Section B and Section C respectively. The models were tabulated in each section. Where as, performance results of models built in the experiment of RF algorithm within various datasets and training options were summarized in Table A.1.

A *Build using 4 hour Aggregated dataset*

As described previously in the chapter, this section focuses on training of RF algorithm using 4 hour aggregated dataset. In training the model two training modes were applied. In general two models were developed and recorded. Detail performance measures of these selected models is shown in Table 5.3.

Table 5.3: Selected RF Models Using 4 Hour Aggregated Dataset

Training mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	100	903	0.960	0.995	95.987
Percentage Split	92	832	0.954	0.993	95.391

The highest classification accuracy that attained in this experiment is 95.99%, from which developed applying 10-Fold Cross Validation method. This model also attains better ROC area and F-Measure outcomes. The time took to build a little higher than the model built by Percentage Split. Generally the model built by 10-Fold Cross Validation method were considered the better on overall assessment

B *Build using Daily Aggregated dataset*

This section gives emphasis in building model of RF algorithm applying daily aggregated dataset. In training the model similarly, the settings and training modes applied in Section A were repeated.. Detail performance measures of the models is shown in Table 5.4.

Table 5.4: Selected RF models using Daily Aggregated dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	288.53	2596.77	0.972	0.977	97.262
Percentage Split	293.7	2643.3	0.974	0.971	97.4559

The highest classification accuracy attained in this experiment is 97.46%, which developed applying Percentage Split method. This model also attains better ROC area, F-Measure than the other model.

C *Build using Monthly Aggregated dataset*

This section gives emphasis in building model of RF algorithm applying monthly aggregated dataset. In training the model similarly, the settings and training modes applied in Section A and Section B are repeated. In general two models were developed. Detail performance measures of the models is shown in Table 5.5.

Table 5.5: Selected RF Models Using Monthly Aggregated Dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	7.06	63.54	0.989	0.995	98.932
Percentage Split	3.76	33.84	0.989	0.995	98.865

The highest classification accuracy attained in this experiment is 98.93%, which developed applying 10-Fold Cross Validation method. This model also attains better ROC area and F-Measure outcomes. The time took to build a little higher than the model built by Percentage Split. Generally the model built by 10-Fold Cross Validation were considered the better on overall assessment

5.1.2 *Building ANN Models*

ANN/MLP has promising performance to fraud detection problem, this is deeply discussed in Section 3.2. In this section building models using ANN algorithm in similar way to steps followed in Section 5.1.1 were applied. Following arranged settings of training methods and training dataset 6 ANN models were developed. Among which 2 are built using 4 hour Aggregated dataset, the other 2 built using daily aggregated dataset and the remaining 2 using monthly aggregated dataset (see Table 5.2). This were presented in Section A, Section B and Section C respectively. Developed models were recorded in each section. Where as, the entire experiment results of ANN algorithm with various datasets and training options were summarized in Table A.1.

A *Build using 4 hour Aggregated dataset*

In this section models of ANN algorithm using the 4 hour aggregated dataset were built. In training the model, in the similar approach to Section A, two training modes are applied. Two models were developed and recorded. Detail performance measures of the models is shown in Table 5.6.

Table 5.6: Selected ANN Models Using 4 Hour Aggregated Dataset

Training mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	77	694	0.955	0.993	95.426
Percentage Split	336	3028	0.954	0.994	95.430

The highest classification accuracy that attained in this experiment is 95.43%, which developed with Percentage Split method. This model also attains better ROC area but lesser F-Measure. The time took to build and evaluate is also higher.

B *Build using Daily Aggregated dataset*

In this section models of ANN algorithm using daily aggregated dataset were built. In training the model, in the similar approach to Section B, two training modes were applied. In general two models were developed. Detail performance measures of the models is shown in Table 5.7.

Table 5.7: Selected ANN Models Using Daily Aggregated Dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	248.52	2236.68	0.98	0.973	97.9925
Percentage Split	236.21	2125.89	0.98	0.973	98.0034

The highest classification accuracy that attained in this experiment is 98.00%, which developed with Percentage Split method. This model attains similar ROC area and F-Measure to the model built using 10-Fold Cross Validation. The time took to build and evaluate is better than the model built using 10-Fold Cross Validation. Generally this model is better than the other model on overall assessment.

c *Build using Monthly Aggregated dataset*

In this section models of ANN algorithm using Monthly aggregated dataset were built. In training the model, in the similar approach to Section C, two training modes were applied. In general two models were developed. Detail performance measures of the models is shown in Table 5.8.

Table 5.8: Selected ANN Models Using Monthly Aggregated Dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	25.42	228.78	0.991	0.995	99.0577
Percentage Split	43.91	395.19	0.989	0.995	98.9239

The highest classification accuracy attained in this experiment is 99.05%, which developed with 10-Fold Cross Validation method. This model also attains better ROC area, F-Measure, time taken to build and evaluate than the other model . Generally this model is better than the other on overall assessment.

5.1.3 *Building SVM Models*

SVM has promising performance to fraud detection problem, this is deeply discussed in Section 3.2. In this section building models using SVM algorithm in similar way to steps followed

in Section 5.1.2 were applied. Following arranged settings of training methods and training dataset 6 SVM models were developed. Out of these 2 are built using 4 hour Aggregated dataset, the other 2 built using daily aggregated dataset and the remaining 2 using monthly aggregated dataset(see Table 5.2). This were presented in Section A, Section B and Section C respectively. Developed models were recorded in each section. Where as, the entire experiment results of SVM algorithm with various datasets and training options were summarized in Table A.1.

A *Build using 4 hour Aggregated dataset*

In this section models of SVM algorithm using the 4 hour aggregated dataset were built. In training the model, in the similar approach to Section A, twotraining modes are applied. Two models were developed and recorded. Detail performance measures of the models is shown in Table 5.9.

Table 5.9: Selected SVM Models using 4 Hour Aggregated Dataset

Training mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	41062	369559	0.953	0.946	95.303
Percentage Split	24229	218059	0.953	0.946	95.302

The highest classification accuracy that attained in this experiment is 95.303%, which developed with 10-Fold Cross Validation method. This model also attains better ROC area but equal F-Measure. The time took to build and evaluate is larger than the other. Generally this model is better on overall assessment.

B *Build using Daily Aggregated dataset*

In this section models of SVM algorithm using the daily aggregated dataset were built. In training the model, in similar approach to Section B, two training modes are appliedand two models developed. Detail performance measures of the models is shown in Table 5.10.

The highest classification accuracy that attained in this experiment is 97.53%, which developed with Percentage Split method. However,its ROC area and F-Measure is the same to the model

Table 5.10: Selected SVM Models Using Daily Aggregated Dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	25533.25	229799.25	0.975	0.951	97.5104
Percentage Split	22859.62	139.05	0.975	0.951	97.5311

build using 10-Fold Cross Validation, and the time took to build and evaluate is better. Generally this model is better on overall assessment.

c *Build using Monthly Aggregated dataset*

In this section models of SVM algorithm using the monthly aggregated dataset were built. In training the model, in similar approach to Section C, two training modes are appliedand two models developed. Detail performance measures of the models is shown in Table 5.11.

Table 5.11: Selected SVM Models Using Monthly Aggregated Dataset

Training Mode	Time (s)		Result		
	Build	Evaluate	F-Measure	ROC	Accuracy
10-Fold Cross Validation	20.11	180.99	0.991	0.993	99.0477
Percentage Split	17.56	0.14	0.99	0.993	98.9534

The highest classification accuracy that attained in this experiment is 99.048%, which developed with 10-Fold Cross Validation method. However,its ROC area and F-Measure is similar the other one, and the time took to build and evaluate is moderate. Generally this model is better on overall assessment.

To summarize, the model building experiment were accomplished based on the combined Constituents, i. e. three datasets having different granularity levels; three distinct ML algorithms ; and two training methods. Total 18 models were developed (the detail is shown in Table A.1). Models of each algorithm were ecorded and discussed in respective sections. The best of these models selected and discussed their performance measures.

5.2 MODEL EVALUATION AND DISCUSSION

The aim of this study is to build a model that performs best in detecting SIM-Box fraudulent subscribers. To meet this goal, features from CDR data were collected and preprocessed . ML algorithms, which are recommended by many researchers on the field of telecom fraud, were selected. Training and testing methods applyied. Finally experiment was performed applying all these fixings. This section presents evaluation of results collected and recorded in the experiments performed in previous sections of this chapter in depth.

In the preceding sections of the chapter, for each algorithm the best models out of the models trained using 4 hour aggregated dataset based on performance measures were selected. Detail performance measure of those selected models is shown in Table 5.12 and their confusion matrix output is also tabulated in Table 5.13.

Table 5.12: Selected Models Trained Using 4 Hour Aggregated Dataset

Model	Time (s)		Result				
	Build	Evaluate	RMSE	Precision	F-Measure	ROC	Accuracy
RF	100	903	0.152	0.960	0.960	0.995	95.987
ANN	336	3028	0.161	0.954	0.954	0.994	95.430
SVM	41062	9382	0.217	0.953	0.953	0.946	95.303

The result in Table 5.12 depicts that all models have comparable performance measures. However the RF model attains a little bit better than the other two in accuracy, Precision, F-Measure, ROC, RMSE, build and evaluate time. Similarly, outcomes of the confusion matrix in Table 5.13 verifies that, the RF model have a little bit lesser FP (4150) compared to the others(5930 and 5752). In fraud detection lesser FP is preferable because, it minimizes the risky of blocking legitimate subscribers.

Moreover Figure 5.1 shows the area under ROC curve of the selected models. It highlighted that the RF model performs better because, this model covers larger area, in the FP Rate verses TP Rate plot, than the other two.

Similarly, for each algorithm the best models trained using daily and monthly aggregated datasets were selected based on their performance measures. Detail performance measure of these selected models is shown in Table 5.14 and Table 5.15 respectively. The results in Table 5.14 and

Table 5.13: Confusion Matrix of Selected Models

		SVM		MLP		RF	
		Cassified as					
		F	N	F	N	F	N
Actual	F	46318	**3452**	46385	**3385**	45904	**3866**
	N	**5930**	144054	**5752**	144232	**4150**	145834

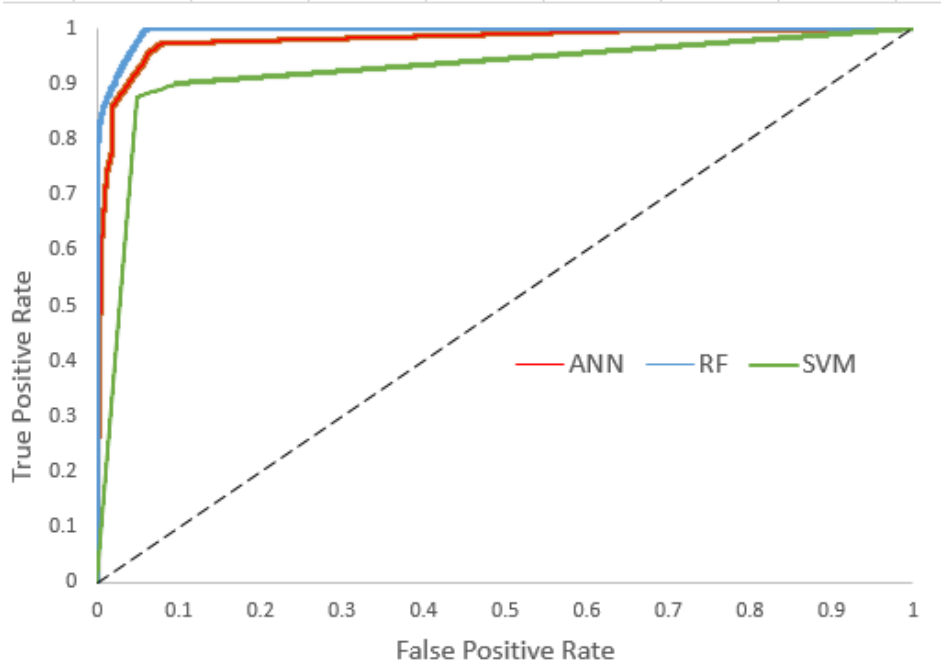


Figure 5.1: ROC Curve Comparison of Models

Table 5.15 describes that all models have comparable performance measure. However the ANN model attains a little bit better than the other two in accuracy, Precision, F-Measure and ROC.

In general, the difference in performance measure between the models is very small. For instance, the difference in accuracy between the highest and lowest performers is less than 0.7%. From the summarized results it had been observed that all the models attained an acceptable level of performance.

Similarly, we have taken the best model with accuracy 95.987, which is developed applying 4 hour aggregated dataset (see Table 5.12). The best model with accuracy 98.003, which is developed applying daily aggregated dataset (see Table 5.14), and the best model with accuracy 99.058, which is developed applying monthly aggregated dataset (see Table 5.15). The performance of models developed using different granularity level datasets were compare. This com-

Table 5.14: Selected Models Trained Using Daily Aggregated Dataset

Model	Time taken		Result		
	Build	evaluate	F-Measure	ROC	Accuracy
RF	294	2643	0.972	0.977	97.456
ANN	406	3656	0.980	0.974	98.003
SVM	22860	139	0.975	0.951	97.531

Table 5.15: Selected Models Trained Using Monthly Aggregated Dataset

Model	Time taken		Result		
	Build	evaluate	F-Measure	ROC	Accuracy
RF	7.06	63.54	0.989	0.995	98.932
ANN	25.42	228.78	0.991	0.995	99.058
SVM	17.56	0.14	0.991	0.993	99.048

parison of accuracy of the models is shown in Figure 5.2. The graph describes the comparison of the accuracy of the models in a data aggregation level verses accuracy plot. From Figure 5.2 we can observe that accuracy increases with granularity level increases form 4 hour to monthly. That means, accuracy is directly proportional to the data aggregation level of the historical data.

After that, we were tasted those three models discussed above using different granularity level test datasets, i. e. the 4 hour, Daily, and monthly test dataset to evaluate their performance in these data granularity levels. This result were summarized in Table 5.16. The first column of the table lists the selected models from different granularity level, that is 4 hour, daily, and monthly. The second column of the table displays the accuracy of the models, the third fourth and fifth columns of the table shows the accuracy of the models test result when they were tested using 4 hour, Daily, and monthly test dataset respectively. The accuracy in the third column is the result of the models when all the selected models tested using 4 hour aggregated test dataset. And similarly the fourth and fifth columns are of the daily and monthly aggregated test datasets respectively.

From the results we can observe that the 4 hour model attains acceptable accuracy level on all different test datasets, whereas the monthly model attains well only on the monthly test dataset, it scores less than 50% on the 4 hour and daily test datasets.

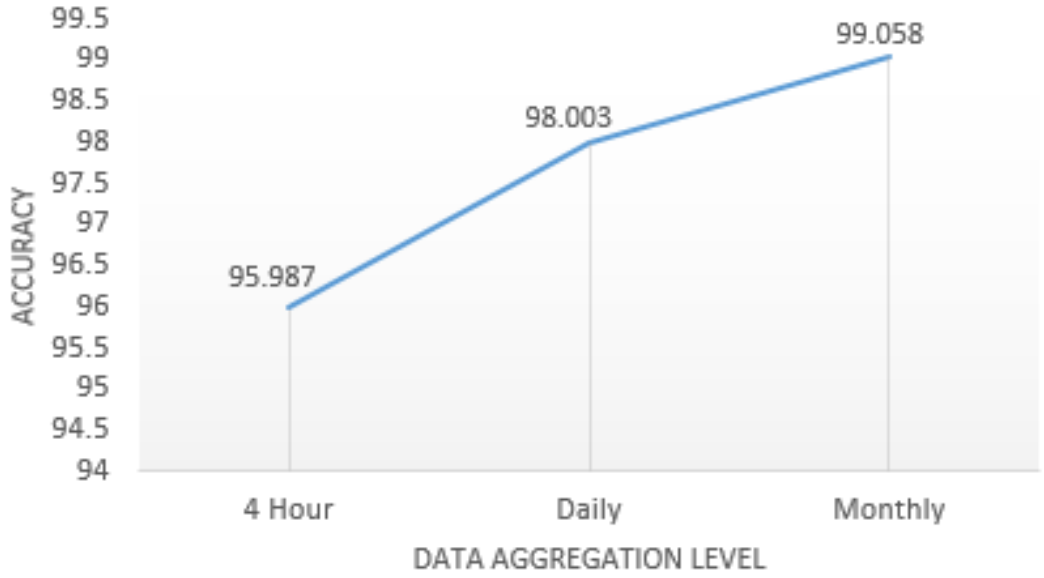


Figure 5.2: RF Models Comparison by Granularity Level

Table 5.16: Comparison of Models Built Using Deferent Datasets

Model	Test Accuracy	Test Accuracy		
		4 hour	Daily	Monthly
4 Hour	95.987	96.535	92.669	90.120
Daily	98.003	81.270	98.003	94.527
Monthly	99.058	32.819	46.708	96.96

Comparative results of these tests is displayed in Figure 5.3 diagrammatically. The horizontal axis displays the selected best models, and the vertical axis shows accuracy of the test results. On the 4 hour best model all the test results are greater than 90%, whereas in the monthly dataset model the result of the 4 hour and daily test datasets are less than 50%.

The focus of this study was to come up with a set of features that can be used to effectively identify SIM cards originating from SIM-Box devices, and machine learning techniques that can classify subscribers with high performance. To attain this objective, primarily, to identify useful features of the data in determining SIM-Box fraud. The CDR data were collected, analyzed, selected and processed to enhance significance of the data. From the entire 33 fields of the collected CDR 8 use-full field were selected and from which a total of twelve features were derived, aggregated and integrated. Different data reduction techniques were applied on the data and finally we generated three different profiles, which are 4 hour, daily and monthly aggregated

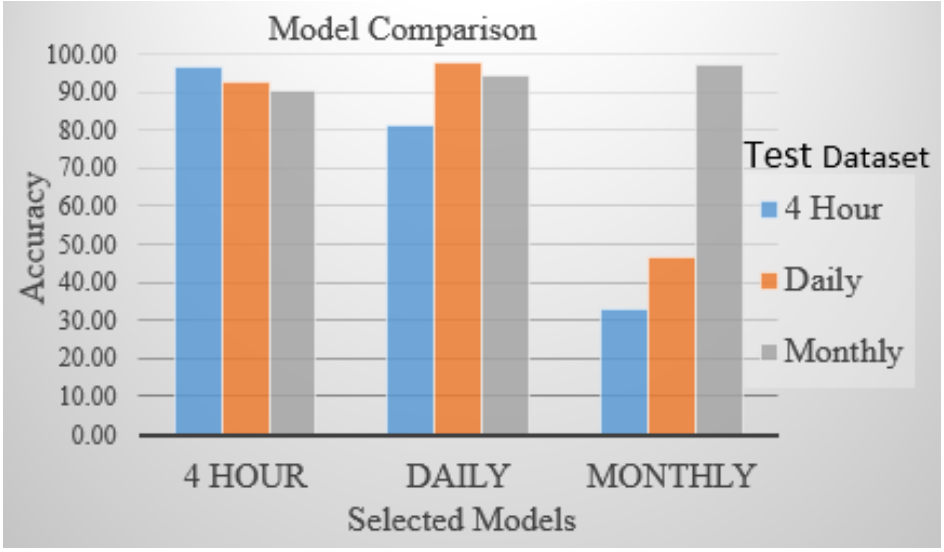


Figure 5.3: RF Models Comparison by Test Dataset

profiles. The derived features are Total Outgoing Calls, Distinct Outgoing Calls, Total Incoming Calls, Distinct Cells Accessed, Total Number of SMSs, Total Amount of Internet Data Used, Average Time Gap between Calls, Subscription Age, Total to Distinct Outgoing Calls Ratio, Distinct Cells Accessed to Total Outgoing Calls Ratio, Total Amount of Internet Data Used to Total Outgoing Calls Ratio, and Total Incoming to Total Outgoing Calls Ratio.

Secondly, we have proposed machine learning algorithms, which have better learning potentials for detection of SIM-Box fraudulent subscribers. The suggested ML algorithms are RF, ANN and SVM. Besides three model training methods i. e. Training/test dataset, 10-Fold Cross Validation and percentage split of 66%; and common performance measures were suggested and discussed in detail.

Then after, by applying these ML algorithms in the prepared dataset 18 different models were built. Among these models comparison have been made, based on suggested performance measures, to select the best model that resulted with highest performance level for each data profile. Even though the difference in performance observed between the models is very small, however the RF algorithm attains a little bit higher performance than the other algorithms in the 4 hour dataset and the ANN model in the other two dataset profile types. For the 4 hour aggregated dataset profile RF model which is developed using the method 10-Fold Cross Validation that attains accuracy of 95.987% was selected. As well for the daily aggregated dataset profile ANN model which is developed using the method Percentage Split that attains accuracy of 98.003% was selected.For the monthly aggregated dataset profile ANN model which is developed using the method 10-Fold Cross Validation that attains accuracy of 99.058% was selected. From these

outcomes we had observed that, the classification accuracy increased with the granularity level increase. This is similar to related works such as [2, 17].

Finally, these selected as the best from each data profile were tested applying the same test dataset to all of them. A 4 hour aggregated test dataset was applied to all of them and recorded an outcome of 32.82%, 81.27% and 96.54% for the Monthly, Daily and 4 hour models respectively. Similarly when a daily aggregated test dataset was applied to all of these models the perform 46.71%, 98.003% and 92.67% and when a monthly aggregated test dataset was applied to all of these models 96.96%, 94.53% and 90.12% were observed. From this observation we can finger out that the 4 hour model performs well on all profile type test datasets. This result depicts that a model trained with a dataset that contain easily identifiable patterns (like the monthly dataset) will face a challenge in predicting a test dataset that contains patterns challenging to identify (like the 4 hour test dataset). In the reverse a model trained with a dataset that contains patterns challenging to identify (like the 4 hour test dataset) can predict well in a test dataset that contain easily identifiable patterns (like the monthly dataset).

Initially, we started our thesis by stating three research questions to be answered i. e.

1. What usage data features can be sensible for SIM-Box fraud detection?
    - For SIM-Box fraud detection useful data features were identified, derived and applied in the experiment.
2. What data granularity level is effective in mitigating SIM-Box fraud problem?
    - For SIM-Box fraud detection a data granularity level of 4 hour were tested in comparison to daily and monthly granularity level. It attains an acceptable level of performance with less than 4% difference in accuracy to the monthly profile. Since, granularity level of 4 hour provides a near-to-real time fraud detection capability it is preferable for SIM-Box fraud detection.
3. What machine learning technique can effectively predict patterns of SIM-Box fraudsters behavior form usage data?
    - For SIM-Box fraud detection RF algorithm was selected as the best classification algorithm compared to ANN and SVM for the 4 hour data granularity level dataset.

Therefore, when we compare the outcomes with our objectives, we meet that, useful features for SIM-Box fraud detection were identified, the data granularity level of 4 hour, which provides

a near-to-real time fraud detection capability, were tested and compared to other granularity levels. The 4 hour data granularity level attains acceptable level of performance. RF algorithm was selected as the best classification algorithm compared to ANN and SVM for the 4 hour data granularity level dataset. Additionally, discussion with domain experts were conducted. Experts from the fraud management domain evaluate the resulted models and suggested that, though it needs time and resources to investigate thoroughly, it have promising outcomes.

6

CONCLUSION AND RECOMMENDATION

The objective of this research was to develop model for detecting and predicting SIM-Box fraud using data mining techniques. To achieve this objective CDR data collected and processed. Models were built, evaluated, and those with better performance were proposed. This chapter discusses on the outcomes and findings of the research. Based on these results derived a conclusion and provided recommendations.

6.1 CONCLUSION

Telecommunication operators in developing countries subsidize their cost of expansion by tariffs collected from international calls. However SIM-Box fraudsters abuse this scenario by delivering less expensive price to callers and divert the revenue from operators. The VoIP technology with SIM-Box and local SIM cards supports them in diverting the international call and deliver them back as a local call. A mechanism which support to detect SIM-Box frauds early, and hinder the fraudsters making business helps the operators to minimize their loss of revenue.

In this research, we have analyzed CDR data and identified twelve relevant features to distinguish SIM-Box fraudulent from legitimate subscribers. The CDR features, that reveal the useful user profile, need higher attention. Understanding behavior of the fraud type deeply is basic for the success of the study. Data is being generated at a faster rate than ever before from mobile devices, this data have valuable information about frauds within it. However, accumulating this big data for long period of time for analysis is challenging. So there is a need to device mitigating this challenge. One of which is to devise near-to-real time analysis approach. In this study we propose to test near-to-real time approach. We proposed three dataset profiles (4 hour, daily, and monthly aggregated) by deriving the identified features. Grounded on the features of the fraud, we have also selected three classifiers RF, ANN and SVM. Combining these constituents a number of models were developed. Among the obtained models those with better detection performance from each profile type were selected.

The RF models scores slightly higher performance than the other two on the 4 hour dataset profiling. The accuracy of the model built with the 4 hour datasets are 95.987% and on the daily and monthly dataset is 98.003% and 99.058 respectively. We can observe from this, the performance of models increases with the increase of data granularity level. This is due to the generation of more identifiable user patterns from historic user data accumulated for longer period. However the model with lower data granularity level tested here attains acceptable accuracy level with lesser than 4% difference from the higher granularity level.

The RF model built with the 4 hour dataset attains 96.535%, 92.669%, 90.12% when tested with the 4 hour, daily, and monthly dataset respectively. Where as the accuracy of the RF model built with monthly dataset is 32.82%, 46.71%, and 96.96% when tested with the 4 hour, daily, and monthly dataset respectively. This shows that the model built with monthly dataset can only used for datasets of monthly aggregated. Whereas the model built with the 4 hour dataset can be used in all aggregation levels. Profiling patterns with narrow granularity level have flexibility, which can be equally applicable in higher granularity level profiles, and reliable in dealing with larger amount of input data.

This result concludes that, model built with the 4 hour aggregated dataset is more appropriate for use in classification model for SIM-Box fraud detection. Since it can be applied to higher granularity level profiling types and provides the near-to-real time fraud detection capability. This allows telecom providers to detect fraudulent subscribers early and deactivate the associated SIM cards rapidly. This capability fundamentally eliminates the economic incentive to conduct such fraud. As a result the loss in revenue of operators decreases.

6.2 RECOMMENDATION

For the future work we recommend the following directions. More refinement of data features may improves performance and accuracy of the technique, for instance including the ratio of IMSI to IMEI, if the data is available. I our case this in not included because IMEI data were not available. This thesis focuses on SIM-Box fraud detection, whereas there are a lot of frauds in the industry, doing similar study on other fraud types and methods is recommendable. We analyzed CDR data, studding SIM-Box fraud detection from the effect of traffic congestion, quality of the audio of the calls and real time signal analysis (if data available) can be research topics.

BIBLIOGRAPHY

[1] M. Sahin, A. Francillon, P. Gupta, and M. Ahamad, "Sok: Fraud in telephony networks," in *Security and Privacy (EuroS&P), 2017 IEEE European Symposium on*, IEEE, 2017, pp. 235–250.

[2] M. R. AlBougha, "Comparing data mining classification algorithms in detection of simbox fraud," *St. Cloud State University theRepository at St. Cloud State*, 2016.

[3] J. Shawe-Taylor, K. Howker, and P. Burge, "Detection of fraud in mobile telecommunications," *Information Security Technical Report*, vol. 4, no. 1, pp. 16–28, 1999.

[4] R. Becker, C. Volinsky, and A. Wilks, "Fraud detection in telecommunications: History and lessons learned," *Technometrics*, vol. 52, no. 1, pp. 20–33, 2010.

[5] Y. Kou, C.-T. Lu, S. Sirwongwattana, and Y.-P. Huang, "Survey of fraud detection techniques," in *Networking, sensing and control, 2004 IEEE international conference*, IEEE, vol. 2, 2004, pp. 749–754.

[6] M. I. Akhter and M. G. Ahamad, "Detecting telecommunication fraud using neural networks through data mining," *International Journal of Scientific and Engineering Research*, vol. 3, no. 3, pp. 601–6, 2012.

[7] I. Ighneiwa and H. Mohamed, "Bypass fraud detection: Artificial intelligence approach," *arXiv preprint arXiv:1711.04627*, 2017.

[8] R. Sallehuddin, S. Ibrahim, A. Hussein Elmi, *et al.*, "Classification of sim box fraud detection using support vector machine and artificial neural network," *International Journal of Innovative Computing*, vol. 4, no. 2, 2014.

[9] C. F. C. Association *et al.*, "Global fraud loss survey," *Press Release, New Jersey, NJ (CFCA)*, vol. 10, p. 2013, 2017.

[10] Apanews. (2017). Ethiopia loses over $52m to telecom fraud-official. 2017-03-06, [Online]. Available: https://mobile.apanews.net/en/news/ethiopia-loses-over-52m-to-telecom-fraud-official (visited on 01/15/2018).

[11] J.-Y. Lee, J.-H. Lee, J.-S. Yeo, and J.-J. Kim, "A snp harvester analysis to better detect snps of ccdc158 gene that are associated with carcass quality traits in hanwoo," *Asian-Australasian journal of animal sciences*, vol. 26, no. 6, p. 766, 2013.

[12] O. A. Abidogun, "Data mining, fraud detection and mobile telecommunications: Call pattern analysis with unsupervised neural networks," PhD thesis, University of the Western Cape, 2005.

[13] R. K. Gopal and S. K. Meher, "A rule-based approach for anomaly detection in subscriber usage pattern," in *Proceedings of World Academy of Science, Engineering and Technology*, 2007, pp. 396–399.

[14] C. Hilas, "Data mining approaches to fraud detection in telecommunications," in *2nd Pan-Hellenic Conference on Electronics and Telecommunications PACET*, vol. 12, 2012.

[15] I. Murynets, M. Zabarankin, R. P. Jover, and A. Panagia, "Analysis and detection of simbox fraud in mobility networks," in *INFOCOM, 2014 Proceedings IEEE*, IEEE, 2014, pp. 1519–1526.

[16] B. Reaves, E. Shernan, A. Bates, H. Carter, and P. Traynor, "Boxed out: Blocking cellular interconnect bypass fraud at the network edge.," in *USENIX Security Symposium*, 2015, pp. 833–848.

[17] C. S. Hilas and P. A. Mastorocostas, "An application of supervised and unsupervised learning approaches to telecommunications fraud detection," *Knowledge-Based Systems*, vol. 21, no. 7, pp. 721–726, 2008.

[18] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.

[19] P. Gosset and M. Hyland, "Classification, detection and prosecution of fraud in mobile networks," *Proceedings of ACTS mobile summit, Sorrento, Italy*, 1999.

[20] O. Ogundile, "Fraud analysis in nigeria's mobile telecommunication industry," *International Journal of Scientific and Research Publications*, 2013.

[21] A. H. Elmi, S. Ibrahim, and R. Sallehuddin, "Detecting sim box fraud using neural network," in *IT Convergence and Security 2012*, Springer, 2013, pp. 575–582.

[22] H. Marah, O. M. Elrajubi, and A. Abouda, "Fraud detection in international calls using fuzzy logic," in *Computer Vision and Image Analysis Applications (ICCVIA), 2015 International Conference*, IEEE, 2015, pp. 1–6.

[23] G. Williams, *Data mining with Rattle and R: The art of excavating data for knowledge discovery*. Springer Science & Business Media, 2011.

[24] I. Witten, E. Frank, M. Hall, and C. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[25] M. Mohammed, M. B. Khan, and E. B. M. Bashier, *Machine learning: algorithms and applications*. CRC Press, 2016.

[26] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[27] M. Kamber, J. Han, and J. Pei, *Data mining: Concepts and techniques*, 2012.

[28] S. Dua and X. Du, *Data mining and machine learning in cybersecurity*. CRC press, 2016.

[29] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[30] A. K. Mishra, S. V. Ramteke, P. Sen, and A. K. Verma, "Random forest tree based approach for blast design in surface mine," *Geotechnical and Geological Engineering*, vol. 36, no. 3, pp. 1647–1664, 2017.

[31] P. Gaur, "Neural networks in data mining," *International Journal of Electronics and Computer Science Engineering*, vol. 1, no. 3, 2013.

[32] K. Verma and P. K. Singh, "An insight to soft computing based defect prediction techniques in software," *International Journal of Modern Education and Computer Science*, vol. 7, no. 9, p. 52, 2015.

[33] S. Zhang, C. Zhang, and Q. Yang, "Data preparation for data mining," *Applied artificial intelligence*, vol. 17, no. 5-6, pp. 375–381, 2003.

[34] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial intelligence*, vol. 97, no. 1-2, pp. 245–271, 1997.

[35] S. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3–24, 2007.

[36] T. O. Ayodele, "Types of machine learning algorithms," in *New advances in machine learning*, InTech, 2010.

[37] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[38] M. A. Hall, "Correlation-based feature selection for machine learning," 1999.

A

APPENDIX

A.1 SUMMARY OF MODELS BUILD

Table A.1: Summary List of Models Build

Test mode	Time taken		Result		
	Build	evaluate	F-Measure	ROC	Accuracy
RF - 4 Hour					
Cross Validation	100	903	0.960	0.995	95.987
Percentage Split	92	832	0.954	0.993	95.391
RF - Daily					
Cross Validation	289	2597	0.972	0.977	97.262
Percentage Split	294	2643	0.974	0.971	97.456
RF - Monthly					
Cross Validation	7	64	0.989	0.995	98.932
Percentage Split	4	34	0.989	0.995	98.865
ANN - 4 Hour					
Cross Validation	77	694	0.955	0.993	95.426
Percentage Split	336	3028	0.954	0.994	95.430
ANN - Daily					
Cross Validation	397	3569	0.980	0.974	97.991
Percentage Split	406	3656	0.980	0.974	98.003
ANN - Monthly					
Cross Validation	25	229	0.991	1	99.0577

Continued on next page

Table A.1 – Continued from previous page

Test mode	Time taken		Result		
	Build	evaluate	F-Measure	ROC	Accuracy
Percentage Split	44	395	0.989	1	98.9239
SVM - 4 Hour					
Cross Validation	41062	9382	0.953	0.946	95.303
Percentage Split	24229	3191	0.953	0.946	95.302
SVM - Daily					
Cross Validation	25533	229799	0.975	0.95	97.5104
Percentage Split	22860	139	0.975	0.95	97.5311
SVM - Monthly					
Cross Validation	20	181	0.991	0.993	99.048
Percentage Split	18	0	0.990	0.993	98.953

A.2 SAMPLE TRAINING RUN INFORMATION

Listing A.1: A Snapshot of RF Model Training Run Information

```
=== Run information ===

Scheme:       weka.classifiers.trees.RandomForest -P 100 -I 100 -num-
   slots 1 -K 0 -M 1.0 -V 0.001 -S 1
Relation:     h4_00
Instances:    199754
Attributes:   13
TOT_OUT
DIS_OUT
OUT_RATE
TOT_IN
TOT_RATE
CELL_OUT
CELL_RATE
TIME_GAP
TOT_SMS
TOT_DATA
DATA_RATE
SUBS_AGE
CLS
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-
   check-capabilities

Time taken to build model: 100.35 seconds
```

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      191738               95.9871 %
Incorrectly Classified Instances      8016                4.0129 %
Kappa statistic                          0.893
Mean absolute error                      0.0471
Root mean squared error                  0.1524
Relative absolute error                 12.5966 %
Root relative squared error             35.2373 %
Total Number of Instances           199754

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall   F-Measure   MCC      ROC Area
   PRC Area  Class
0.922    0.028    0.917      0.922    0.920      0.893    0.995
   0.986     F
0.972    0.078    0.974      0.972    0.973      0.893    0.995
   0.998     N
Weighted Avg.    0.960    0.065    0.960      0.960    0.960
   0.893    0.995     0.995

=== Confusion Matrix ===

a      b   <-- classified as
45904   3866 |      a = F
4150 145834 |      b = N
```

A.3 SUBSETS OF FUTURES SELECTED

Table A.2: Subsets of Selected Features

Subset 1	**Subset 2**	**Subset 3**
TOT_OUT	**X**	**X**
DIS_OUT	DIS_OUT	DIS_OUT
OUT_RATE	OUT_RATE	OUT_RATE
TOT_IN	TOT_IN	TOT_IN
TOT_RATE	TOT_RATE	**X**
CELL_OUT	**X**	**X**
CELL_RATE	CELL_RATE	CELL_OUT
TIME_GAP	TIME_GAP	TIME_GAP
TOT_SMS	TOT_SMS	TOT_SMS
TOT_DATA	TOT_DATA	**X**
DATA_RATE	DATA_RATE	DATA_RATE
SUBS_AGE	SUBS_AGE	SUBS_AGE

A.4 SAMPLE DATASET

```
@relation day08

@attribute CALLING_NBR numeric
@attribute TOT_OUT numeric
@attribute DIS_OUT numeric
@attribute OUT_RATE numeric
@attribute TOT_IN numeric
@attribute TOT_RATE numeric
@attribute CELL_OUT numeric
@attribute CELL_RATE numeric
@attribute TIME_GAP numeric
@attribute TOT_SMS numeric
@attribute TOT_DATA numeric
@attribute DATA_RATE numeric
@attribute SUBS_AGE numeric
@attribute CLS {F,N}

@data
25193031333238325000,70,70,1,0,0,4,0.06,6.59,0,0,0,15,F
25193230353733325000,29,8,0.28,17,0.59,12,0.41,21.66,1,0.07,0,2988,N
25193032333336306000,76,76,1,30,0.39,3,0.04,1.59,5,139,1.83,680,F
25193130363237315000,36,13,0.36,15,0.42,18,0.5,23.6,1,34.8,0.97,3331,N
25193132383336346000,23,11,0.48,16,0.7,11,0.48,37.67,4,0,0,2990,N
25193337383839370000,51,51,1,3,0.06,2,0.04,8.56,24,157,3.08,1614,F
25193131313737300000,30,15,0.5,25,0.83,17,0.57,27.9,2,32.5,1.08,4785,N
25193133323430390000,20,15,0.75,50,2.5,7,0.35,37.21,1,16.51,0.83,3912,N
            . . .                       . . .
199761  25193431343836330000,21,9,0.43,7,0.33,5,0.24,7.51,1,0.01,0,1277,N
199762  25193131323630375000,11,5,0.45,5,0.45,11,1,6.48,5,15.44,1.4,1294,N
199763  25193131353439355000,14,10,0.71,4,0.29,2,0.14,28.79,1,6.42,0.46,1230,N
199764  25193138323739345000,21,16,0.76,5,0.24,7,0.33,11.59,1,0.01,0,2893,N
199765  25193232323230314000,11,7,0.64,4,0.36,4,0.36,17.58,1,4.75,0.43,2430,N
199766  25193133363030335000,5,3,0.6,3,0.6,4,0.8,36.44,8,4.15,0.83,3581,N
199767  25193132343334300000,3,1,0.33,6,2,2,0.67,17.49,1,0.7,0.23,3238,N
199768  25193734303338373000,21,21,1,0,0,3,0.14,0.77,0,0,0,84,F
199769  25193131383838376000,12,8,0.67,3,0.25,4,0.33,17.5,5,9.11,0.76,2908,N
199770  25193131343234343000,10,9,0.9,8,0.8,6,0.6,19.64,2,1.96,0.2,1465,N
199771  25193032363639312000,11,11,1,5,0.45,4,0.36,7.7,0,0,0,11,F
199772  25193330353431323000,8,5,0.63,7,0.88,3,0.38,13.22,1,0.24,0.03,66,N
199773
```

Figure A.1: A Snapshot of ARFF Format Sample Dataset, 4 Hourly aggregated sample profile