Scam Call Detection and Categorization: A Machine Learning Approach to Thwart Telecommunications Fraud

by Karol A. Pierre

B.S. in Computer Engineering, May 2019, Villanova University
M.S. in Security Informatics, Dec 2020, Johns Hopkins University

A Praxis submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial fulfillment of the requirements
for the degree of Doctor of Engineering

August 31, 2024

Praxis directed by

Hamza F. Alsarhan
Professorial Lecturer of Engineering Management and Systems Engineering

The School of Engineering and Applied Science of The George Washington University certifies that Karol Pierre has passed the Final Examination for the degree of Doctor of Engineering as of July 22, 2024. This is the final and approved form of the Praxis.

Scam Call Detection and Categorization: A Machine Learning Approach to Thwart Telecommunications Fraud

Karol A. Pierre

Praxis Research Committee:

Hamza F. Alsarhan, Professional Lecturer in Engineering and Applied Science, Praxis Director

Amir Etemadi, Associate Professor of Online Engineering Programs, Committee Chair

Andrea Molina, Professional Lecturer in Engineering and Applied Science, Committee Member

© Copyright 2024 by Karol A. Pierre
All rights reserved

Dedication

The author wishes to dedicate her research to her family, the source of her strength and inspiration to conduct this work. She also wishes to dedicate her work, in particular, to her grandfather, Jean, who first ventured alone to the United States to build and create a life of opportunity for his children and grandchildren. *Merci Papi Voltaire!*

Acknowledgements

The author wishes to acknowledge and thank Dr. Hamza Alsarhan for his endless support, time, and insight as both a professor and a research advisor. Dr. Alsarhan provided guidance throughout the research proposal and active research phases and aided in shaping this work.

The author also wishes to acknowledge and thank her amazing community and council of friends, mentors, and advisors that have been relentless in their support of her endeavors. Through words of encouragement, prepared meals, prayer, study snacks, and more, her community has poured into her so that she may be able to do the same unto others.

Abstract of Praxis

Scam Call Detection and Categorization: A Machine Learning Approach to Thwart Telecommunications Fraud

Phone scam calls have become a feature of a user's daily life. A user can almost expect to receive at least one scam call a day. Threat actors use these calls as a medium or an attack vector to execute further cybercrimes such as identity theft and financial fraud. This issue has impacted and continues to impact society 20 years later by continuously evolving and evading various state-of-the-art methods and techniques. In an effort to address this problem, this praxis introduces a machine learning model for both binary prediction and multi-class classification of scam calls. The model presented is developed by using data aggregated from the United States Federal Trade Commission (FTC) organization, phone number metadata, and the Universal Licensing System. This data provides insight into known scam calls between the months of October 2023 and December 2023.

The solution provided in this praxis not only detects incoming phone calls as malicious or authentic but can further categorize those calls into six unique classes: Authentic, Medical, Utilities, Impersonator, Technical, and Financial. These categories or labels aim to provide a user with more insight into the intention of a caller, especially in circumstances where the area code of an incoming call is familiar, to prevent a suspected malicious call from being answered. The objective of the solution presented is to reduce the likelihood that a user becomes a victim to cybercrime conducted over the phone.

Table of Contents

Dedication iv

Acknowledgements v

Abstract of Praxis vi

List of Figures x

List of Tables xi

Chapter 1—Introduction 1

1.1 Background 1

1.2 Research Motivation 2

1.3 Problem Statement 2

1.4 Thesis Statement 3

1.5 Research Objectives 4

1.6 Research Questions and Hypotheses 4

1.6.1 Research Questions 4

1.6.2 Research Hypotheses 5

1.7 Scope of Research 5

1.8 Research Limitations 6

1.9 Organization of Praxis 6

Chapter 2—Literature Review 8

2.1 Introduction 8

2.2 Telecommunications Background & Ecosystem Overview 9

2.3 Scam Call Overview and Types 10

Chapter 3—Methodology 30

3.1 Introduction 30
3.2 Data Collection 31
3.3 Dataset Development 33
3.4 Exploratory Data Analysis 35
3.5 Data Preprocessing 38
3.5.2 Categorical Data Encoding 39
3.5.3 Feature Selection 40
3.5.4 Data Balancing 41
3.5.5 Data Splitting 41
3.6 Model Selection 42
3.7 Hyperparameter Tuning 44
3.8 Model Training & Testing 45
Chapter 4—Results 46
4.1 Introduction 46
4.2 Data Collection 46
4.3 Dataset Development 49
4.4 Exploratory Data Analysis 50
4.5 Data Preprocessing 54
4.5.1 Data Standardization & Consolidation 55
4.5.2 Categorical Data Encoding 56
4.5.3 Feature Selection 58
4.5.4 Data Balancing 64
4.5.5 Data Splitting 66

4.6 Hyperparameter Tuning 66

4.7 Model Training & Testing 70

Chapter 5—Discussion and Conclusions 78

5.1 Discussion 78

5.1.1 Research Hypothesis 1 78

5.1.2 Research Hypothesis 2 78

5.1.3 Research Hypothesis 3 79

5.2 Conclusions 79

5.3 Contributions to Body of Knowledge 80

5.4 Recommendations for Future Research 81

References 83

Appendix A 91

List of Figures

Figure 2-1. Hierarchy of Phone Calls 11

Figure 3-2. Iteration Between EDA and Data Processing 37

Figure 4-1. Original Category Distribution 65

Figure 4-2. Category Distribution with SMOTE 65

Figure 4-3. Hyperparameter Tuning for Binary Classification using Decision Tree 68

Figure 4-4. Hyperparameter Tuning for Multiclass Classification using Decision Tree.. 68

Figure 4-5. Hyperparameter Tuning for Binary Classification using XGBoost 69

Figure 4-6. Hyperparameter Tuning for Multiclass Classification using XGBoost 69

Figure 4-7. Decision Tree Binary Confusion Matrix 71

Figure 4-8. XGBoost Binary Confusion Matrix 72

Figure 4-9. Decision Tree Multiclass Confusion Matrix 74

Figure 4-10. XGBoost Multiclass Classification Confusion Matrix 76

Figure A-1. Hyperparameter Tuning for Binary Classification using Decision Tree..... 93

Figure A-2. Hyperparameter Tuning for Binary Classification using XGBoost 93

Figure A-3. Hyperparameter Tuning for Multiclass Classification using Decision Tree. 94

Figure A-4. Hyperparameter Tuning for Multiclass Classification using XGBoost 94

List of Tables

Table 3-1. Categorical Encoding 40

Table 4-1. Data Source Details 47

Table 4-2. Supplementary Data Source Details 48

Table 4-3. Dataset Construction 50

Table 4-4. Dataset Metrics 52

Table 4-5. Feature Mapping 53

Table 4-6. Feature Data Types 54

Table 4-7. State Abbreviation Standardization 55

Table 4-8. Subject Consolidation for Multiclass Categorization 56

Table 4-9. One-hot Encoding for *Carrier_Name* Attribute 57

Table 4-10. One-hot Encoding for *Consumer_City* Attribute 57

Table 4-11. One-hot Encoding for *Consumer_State* Attribute 57

Table 4-12. Data Prior to Factorize Encoding 58

Table 4-13. Data After Factorize Encoding 58

Table 4-14. Feature Importance for Decision Tree Binary Prediction 59

(using one-hot encoding) 59

Table 4-15. Feature Importance for XGBoost Binary Prediction 59

(using one-hot encoding) 59

Table 4-16. Feature Importance for Decision Tree Multiclass Prediction 60

(using one-hot encoding) 60

Table 4-17. Feature Importance for XGBoost Multiclass Prediction 60

(using one-hot encoding) 60

Table 4-18. Feature Importance for Decision Tree Binary Prediction 61
(using factorize encoding) 61
Table 4-19. Feature Importance for XGBoost Binary Prediction 62
(using factorize encoding) 62
Table 4-20. Feature Importance for Decision Tree Multiclass Prediction 63
(using factorize encoding) 63
Table 4-21. Feature Importance for XGBoost Multiclass Prediction 63
(using factorize encoding) 63
Table 4-22. Dataset Split 66
Table 4-23. Decision Tree Binary Classification Report 70
(without SMOTE) 70
Table 4-24. Decision Tree Binary Classification Report 70
(with SMOTE) 70
Table 4-25. XGBoost Binary Classification Report 71
(without SMOTE) 71
Table 4-26. XGBoost Binary Classification Report 72
(with SMOTE) 72
Table 4-27. Decision Tree Multiclass Classification Report 73
(without SMOTE) 73
Table 4-28. Decision Tree Multiclass Classification Report *(with SMOTE)* 74
Table 4-29. XGBoost Multiclass Classification Report *(without SMOTE)* 75
Table 4-30. XGBoost Multiclass Classification Report *(with SMOTE)* 75
Table A-1. Dataset Feature List 91

List of Acronyms

CDR	Call Detail Record
CNN	Convolutional Neural Network
DNS	Domain Name System
EDA	Exploratory Data Analysis
FCC	Federal Communications Commission
FTC	Federal Trade Commission
IRS	Internal Revenue Service
LSTM	Long Short-term Memory
MMS	Multimedia Messaging Service(
MNO	Mobile network operators
MvNO	Mobile virtual network operators
NLP	Natural Language Processing
PII	Personally Identifiable Information
PMG	Progressive Multi Gray-Leveling
RNN	Recurrent Neural Network
SDAE	Stacked Denoising Autoencoder
SHAKEN	Signature-based Handling of Asserted Information Tokens
SMS	Short Message Service
SPIT	Spam over Internet Telephony
STIR	Secure Telephone Identity Revisited
SVM	Support Vector Machine
ULS	Universal Licensing System

VoIP	Voice over IP

Chapter 1—Introduction

1.1 Background

Phone scams have become an ordinary expectation of everyday life and despite advancements in telecommunication security, scam calls continue to have lasting consequences on users. According to the Federal Trade Commission (FTC) and other government organizations, phone scams yielded the highest reported financial loss per person in the United States for the year 2022 (Federal Trade Commission, 2023). Its impact results not only in financial loss but also in identity theft and financial fraud.

To better understand how phone scams are positioned within the telecommunication threat landscape, an understanding of phone scams must be established. Social engineering is the process of targeting an individual to reveal personal information, such as personally identifiable information (PII) or credentials, for nefarious purposes. It can occur face-to-face, virtually, or over the phone. Vishing or voice-phishing is a type of social engineering attack where a threat actor attempts to gather information from a target by employing different tactics and techniques, including impersonation of a government representative. Phone scams are a form of vishing attack. Throughout this work, this type of cybercrime will be addressed as such.

A socially engineered attack that occurs over the phone, also known as vishing, is an invisible act. At the moment, a threat actor is able to execute an attack on an unsuspecting victim, not witnessing the impact or long-term result of said attack. The work conducted in this praxis aims to address the issue of scam calling by exploring existing solutions in order to develop a novel detective and preventative application.

1.2 Research Motivation

The telecommunication problem of fraudulent phone calls continues to impact users today, with incessant scam-based calls bombarding the average user. Data reported by the Federal Trade Commission regarding *Do Not Call* complaints indicate that over two million complaints were made for the year 2023 alone (Federal Trade Commission, 2023). These numbers are only representative of reported instances and do not account for the scam calls received by millions of other users, which are never reported.

Tools integrated within Mobile Network Operators (MNOs) such as T-Mobile, AT&T, and Verizon offer some relief by distinguishing between scam calls and authentic calls, often with a label of "Scam Likely". These tools have been available on mobile devices for the last few years, but scam calls still persist, with the amount lost to these crimes increasing year over year (Federal Trade Commission, 2023). The motivation of this praxis is to present a novel approach that uses machine learning to accurately detect and categorize scam-based calls. By offering classification and categorization of scam calls from authentic calls, users will be able to better determine the legitimacy of a call and the caller.

1.3 Problem Statement

Phone scams yielded the highest reported financial loss per person in the United States for the year 2022, contributing to almost 8.8 billion dollars lost to fraud overall. (Federal Trade Commission, 2023)

Fraudulent phone calls or phone scams are invisible in nature. The act of committing telecommunications fraud and targeting a user is a faceless act because the threat actor does not directly see the person they are harming. The impact, which can

present itself as identity theft and financial fraud, is a personal experience that can devolve into the plummeting of one's credit score and the inability to apply for home mortgages or other credit-based purchases. These outcomes affect not only the victim but also their family and loved ones, potentially curbing their shared goals and endeavors. The mean value lost to these crimes from unsuspecting victims is $1400 (Federal Trade Commission, 2023). In order to prevent the continued proliferation of financial loss due to phone scams, the integration of a mobile detective and preventive tool is necessary.

1.4 Thesis Statement

A machine learning-based Anti-Scam call Tool is required to reduce the likelihood of social engineering attacks in the form of scam calls.

The usage of a predictive-based anti-robocall tool can aid in reducing the occurrence of vishing-based attacks such as scam calls, which are a form of social engineering. This solution exists as both a detective and preventive measure that can be integrated with existing anti-spam security tooling offered by Mobile Network Operators. The model's primary objective is to detect whether an incoming call is scam or authentic and to offer a categorical determination of the type of call if detected as scam. The step the solution takes includes gathering call data record information such as location (city, state, zip code) and supplementing that information with a reverse look-up to gather additional details such as carrier data. The determination of a call as scam and its simultaneous categorization is made within seconds, enabling a vishing-based attack to be thwarted before it is answered and removing the nuisance of receiving incessant calls by potential threat actors.

1.5 Research Objectives

The research objective of this work is to develop a predictive machine learning model that can accurately detect spam-based calls and categorize the type of scam to aid in improving the existing state-of-the-art solutions. The key goals required to do so include:

- Distinguish between authentic and scam-based calls.
- Improve the performance (prediction accuracy) of a predictive model with the use of phone carrier metadata.
- Categorize calls identified as scam calls with appropriate labels (Telemarketer, Medical, Debt, etc.).

1.6 Research Questions and Hypotheses

In this work, the following questions and hypotheses are posed and presented in an effort to improve the state of the art of existing solutions and serve as a guide through the development of a potential solution:

1.6.1 Research Questions

RQ1: Does the use of phone call metadata improve the performance of predicting whether a call is scam-based?

RQ2: Does the use of a scam call prevention tool accurately block only scam/spam calls without interruption to user experience and impact to user privacy?

RQ3: Does the use of a scam call prevention tool accurately categorize the type of scam call for user awareness?

1.6.2 Research Hypotheses

H1: Phone call metadata improves the performance of a predictive model for scam call detection.

H2: A scam call prevention tool accurately blocks scam calls without interruption to user experience and impact to user privacy.

H3: The use of a scam call prevention tool accurately categorizes the type of scam call for user awareness.

1.7 Scope of Research

This research explores mechanisms to detect fraudulent phone calls and offers a practical countermeasure to reduce the likelihood of occurrence and prevent users from falling victim to these telecom crimes. The objective of the study is to develop and validate a solution capable of distinguishing scam calls from legitimate calls and categorizing the types of scam calls into classes. The methodologies that will be employed to conduct this work include machine learning and experimentation. The data that will be used for this study includes phone call metadata for over 10,000 scam calls from FTC and authentic calls from Alesco Data combined. The data collection and dataset development processes will be discussed in further detail in Section 3. All data points are supplemented with phone carrier metadata as part of the feature engineering process. This work will produce deliverables that include an informational report outlining empirically-proven countermeasure recommendations and an anti-robocall tool to support the reduction and prevention of users from becoming targets of telecommunications fraud.

1.8 Research Limitations

This research focuses on the detection and categorization of fraudulent phone calls by examining supervised machine learning models such as Decision Tree and eXtreme Gradient Boosting.

Studies in this area have advanced in the last fifteen years, as will be focused on in Section 2; however, there are limitations that prevent the widespread adoption of certain solutions while others are incapable of being utilized as a preventative mechanism. Call detail records which is the metadata of any given phone call, such as time of call and call duration, are often used as the attributes to develop the training and testing datasets during model development. These common features which are leveraged across the various works that will be reviewed in Section 2 indicate a lack of research around supplementing call metadata to include information that extends beyond a phone call actually being answered. Additional information that can be leveraged to prevent scam calls from occurring include carrier data and FCC Universal Licensing System information.

This praxis is centered around and limited to accurately detecting scam calls and categorizing them into 6 unique classes which include: Authentic, Medical, Utilities, Impersonator, Technical, and Financial.

1.9 Organization of Praxis

This praxis is organized into five key chapters:

Chapter 1, as we have discussed, lays the foundation for this research, offering context to the problem of mass scam calls and highlighting the key objectives of this work to mitigate the outcome of becoming a victim. Chapter 2 reviews and critiques existing

literature related to the focus of this work. It establishes the baseline for the state of the art from which the proposed solution is able to build on. Chapter 3 discusses how the study was conducted by exploring the methods leveraged for experimentation, which include data gathering, data cleaning, feature engineering, and more. Chapter 4 highlights the results of the study and experimentation. The performance of the proposed solution is evaluated and demonstrated using graphs and tables, which include the confusion matrix; this measures the performance of a given ML model. Chapter 5 closes out the study by dissecting the result and its impact on reducing the likelihood of a user falling victim to cybercrime via phone scams.

Chapter 2—Literature Review

2.1 Introduction

This chapter focuses on the state of the art of solutions aimed at solving the phone scam call problem. It is organized into seven sections, starting with 2.1, Introduction, which offers context for the structure of the literature review. Section 2.2, Telecommunications Background & Ecosystem Overview, establishes the foundation of telecommunications and provides an introduction to its landscape. Section 2.3, Scam Call Overview and Types, follows next to define what scam calls are and describes examples of the varying types of malicious calls. Section 2.4, Blacklisting-based Scam Call Mitigations, presents existing solutions that incorporate the use of blacklisting mechanisms, the process of generating a list or repository of known bad callers, for the purpose of preventing scam calls from occurring. Section 2.5, Audio-based Scam Call Mitigations, highlights approaches to the scam call problem that employ the usage of audio and audio data to supplement their methodology. Section 2.6, Machine Learning-based Scam Call Mitigations, presents solutions that implement the use of machine learning models for their methodology. Section 2.7, Government Laws & Mobile Network Operator Mitigations, dissects the existing technologies offered by mobile carriers to combat the issue. This chapter concludes in Section 2.8, Summary and Conclusion, with a summary of the critique for the types of solutions presented, underscores the gap that exists in today's solutions and offers a novel mitigation strategy to solve this phone scam call problem.

2.2 Telecommunications Background & Ecosystem Overview

The telecommunication infrastructure has transformed the capabilities of modern society; it has enabled its users to communicate from long distances over optical fiber, radio, and more through mediums such as phone calls, text messages, and the internet. The focus of this literature review and work is to specifically highlight the ecosystem of telecom as it relates to phone calls and review the current state of the art in voice phishing detection or phone s**c**am call detection and prevention.

The traditional phone system, or the Public Switched Telephone Network (PSTN), is simple in its design as it is comprised of two wires, a speaker, and a microphone (Doherty, 2006). This simple design has evolved over the years, from the need for operators and switchboards to the AT&T Bell Lab creation of the 3-3-4 numbering system in the United States. The components of the numbering system (area code - exchange prefix - line number) are what have allowed for the scalability of phone numbers in a given region (Doherty, 2006). This translates to the capability of coast-to-coast communication from one home to another. From this, the need to be in contact while mobile or to provide alternatives to remain in contact introduces the inception of the wireless phone system with mobile phones and the internet with Voice over Internet Protocol (VoIP).

The mobile phone system has improved connectivity while on-the-go enabling people to be reachable almost instantaneously. Mobile phone providers or mobile network operators such as AT&T, T-Mobile, and Verizon exist as some of the largest organizations that offer these wireless telecommunication services in the United States. In a similar fashion, Voice over Internet Protocol (VoIP), also known as IP telephony,

allows users the ability to also make phone calls, but rather, over an internet connection. Examples of this can include communication application solutions such as Skype, WhatsApp, Zoom Phone, and Viber. As the solutions for digital connection and interconnectivity continue to grow, the security around these technologies should follow suit to ensure users do not fall vulnerable to telecommunications fraud.

2.3 Scam Call Overview and Types

Within the mobile telecommunication ecosystem, the threat landscape is vast, ranging from scam calls and voice phishing to eavesdropping and Short Message Service (SMS)/Multimedia Messaging Service(MMS) phishing. Scam calls or unwanted calls, in particular, “including illegal and spoofed robocalls - are the Federal Communications Commission's top consumer complaint” (FCC, Stop Unwanted Robocalls and Texts, 2023). These calls can create an interruption to a user’s daily life with greater impact, leading to identity theft/fraud, credit fraud, and money loss.

A scam phone call can be defined as an unwanted or unsolicited call from a potential threat actor that masks their identity by pretending to be from a legitimate source. Robocalls, which at times are considered unwanted calls, are automated calls with a pre-recorded message. According to the FCC and FTC, while some robocalls are legal, if there is an attempt for something to be sold, it is likely illegal, with most of them being scams (Heras, 2023). In both instances, threat actors often employ social engineering tactics and techniques to create a sense of urgency or concern in their target. This is often done by indicating an emergency or threatening an immediate impact on finances.

Social engineering is the application of social psychology and as described by Hadnagy (2010) is the act of social engineers hacking humans. According to researcher

Peltier (2006), there are three key aspects that threat actors employ when executing a socially engineered attack. These aspects are to have alternative routes to persuasion, attitudes and beliefs that affect human interactions, and a technique for persuasion and influence. These concepts applied to an unsuspecting user create the environment in which various scam calls can occur.

There are numerous types of phone call scams that can occur. The following graphic highlights a high-level list of the various types of phone scams:

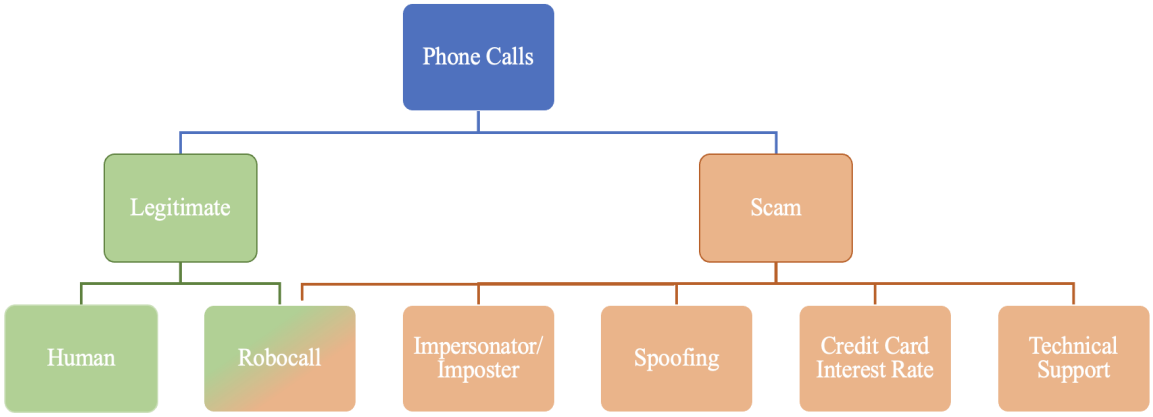

Figure 2-1. Hierarchy of Phone Calls

2.3.1 Robocall

As previously mentioned, robocalls are phone calls that contain pre-recorded messages for the recipient. While some robocalls can be legitimate, those that attempt to sell the recipient anything are illegal and are likely scams. These calls can be sent en mass to a list of phone numbers using a mechanism known as an autodialer (Tu et al., 2016). An autodialer provides the programmatic or automated capability to send a single message to an array of targets. As noted by authors Tu et al., it is often made possible via

VoIP technology, which provides scheduled broadcast and customizable caller ID to further obfuscate one's identity.

2.3.2 Impersonator/Imposter

An impersonator or imposter-based scam call is typically when a threat actor poses as an individual from a legitimate source such as the Internal Revenue Service (IRS) or even poses as a family member or loved one to gain money (FCC, 2024). Examples of legitimate sources that threat actors have historically used to obfuscate their identity, as denoted by the FCC and FTC, include the IRS, the Social Security Administration, bill-collecting agencies, utility companies, and health insurance agencies. The effectiveness of a scam call where a threat actor impersonates an authoritative figure can be connected to one of Cialdini's (2007) six principles of influence, which is authority. People are less likely to question the validity of a given situation under the premise of being contacted by a government agency. This can become exacerbated when the factor of time is also considered. According to Stajano & Wilson's (2011) seven principles of scams, people are likely to sacrifice proper assessment, reasoning, or rationality when making a decision due to the pressure or limitation of time. Comparably, authors Clare & Levine (2019) also state that because people are in a truth-default state, where there is an assumption of honesty, people are not actively evaluating the validity of what is being communicated unless there is a trigger.

In order to further understand how threat actors manage to be successful with social engineering attacks, researchers Jones et al. (2021) examined the effectiveness of persuasion techniques, specifically as it applied to voice phishing. Their approach was to review and encode 86 real-world vishing attack examples to determine which of the five

social engineering persuasion principles: authority, commitment, reciprocation and consistency, distraction, similarity and deception, and social proof were present. Their work revealed that in instances where scam calls or vishing attacks were conducted 95% of those cases featured the persuasion technique of authority.

2.3.3 Spoofing

Spoofing is the practice of intentionally and maliciously choosing to modify the caller ID to increase the likelihood the phone call is answered (FCC, 2024). Typically, this can be done by leveraging technologies such as VoIP mobile applications to allow threat actors to select regions or local areas for their phone number generation. By committing caller-id spoofing which can include selecting a phone number with a similar area code to that of selected targets, or modifying the displayed phone number, as described by Song et al. (2014), a malicious actor is able to create a sense of familiarity for the potential target, increasing the probability the phone call is answered.

2.3.4 Credit Card Interest Rate

Credit card interest rate scams via phone calls are centered around eliciting personally identifiable information (PII). Threat actors will pose as representatives from banks or credit card companies offering the ability to reduce rates (FCC, 2024). By collecting a few key data points such as name, birth date, and social security number in order to secure and “approve a new, lower rate”, they lure unsuspecting victims into handing over data that can be used for nefarious purposes.

Credit-based phone scam calls can also be considered a type of impersonator scam due to the similar tactics employed to trick unassuming victims.

2.3.5 Technical Support

Technical support-based scam calls can occur as an extension of adware. The role of adware in a scheme, which is often characterized by unwanted pop-ups on a digital device, is to create a sense of urgency for the user to act. Examples of this can be a pop-up ad because the user is the 1,000th visitor on a website and has just won an iPad or a pop-up because a virus has been found on the user's system and requires technical assistance to have it removed. The latter of these examples urges a user to contact technical support immediately to resolve the issue. As highlighted by the Federal Communications Commission, scammers will employ tactics such as spoofing in a technical support scam to further advance the scheme; this is done to gain credit card information and even gain remote access to a target's physical system.

2.4 Blacklisting & Trust-based Scam Call Mitigations

The state-of-the-art mitigations for voice phishing or the phone scam problem are varied and have evolved over time. One such area of solutions and among the earliest of its kind highlighted by researchers are those that leverage the concept of whitelisting vs. blacklisting and profiling vs. reputation-score as a method. Dietrich and Hesse (2007) formulated an anti-spam mechanism that addressed the rise of Spam over Internet Telephony (SPIT). The scenarios they considered included the likely chance that traditional phone numbers can be reached by internet telephones or those that leverage voice over IP networks. They proposed a solution that utilized both blacklisting and whitelisting to combat this potential issue. They specifically indicated that a countermeasure against SPIT "could be blacklisting of VoIP servers that initiate unwanted calls" (Dietrich and Hesse, 2007). While this is a way forward, there is a

potential disadvantage where a legitimate user could get blacklisted. The process of blacklisting uses the IP address of those VoIP servers. Those servers carry signaling information for users to initiate their calls. As discussed in their work, because VoIP servers or proxies are often used for a magnitude of users, if this solution is solely leveraged, when the VoIP server is blocked, all users associated with the same server are also blocked. The authors address this problem by presenting and dissecting the use of whitelisting. By leveraging both options simultaneously, when a legitimate user is blacklisted, immediate action can be taken to add them to the whitelist. This solution that researchers Dieterich and Hesse proposed was presented in 2007, and while it may have been an initial stopgap to prevent spam calls, the VoIP landscape has advanced. This solution is a theoretical solution with no empirical data to support its effectiveness. Additionally, it would be unable to properly scale to adequately serve as a response to the capabilities of modern systems. A similar argument is presented by researchers Quittek et al. (2008), where they aggregated requirements for a preventative system to be effective against SPIT within the VOIP environment. Among those, they include a system that "should be flexible in order to cope with SPIT getting more sophisticated."

Hoffstadt et al. (2007) further advanced the options available to remediate the issue of fraud and misuse of VoIP by offering a comprehensive framework titled the SUNSHINE framework. It leverages an intrusion detection system, a firewall, a distributed sensing system, Domain Name System-based (DNS) real-time blacklisting, call data record analysis based on statistical analysis, and more. The solution presented by the authors attempts to address the multi-faceted nature of dealing with threats of spam calling over VoIP and suggests that there is no single solution. While this is true,

there seems to be a lack of empirical evidence to suggest its effectiveness and practical implementation.

Blacklisting as a mitigation has been long considered an initial solution to various types of phishing attacks, including voice phishing or the act of scam calling. Researchers Khonji et al. (2013) surveyed various solutions as it applied to email phishing and voice phishing, and they concluded that the active detection of these attacks by using blacklisting does not provide protection against zero-hour phishing attacks. These types of attacks are threats that are executed via a novel or previously unknown exploit, specifically, as it relates to malicious sites. Similarly, Researchers Dissanayake et al. (2023), argue that "traditional approaches which detect frauds using blacklisted telephone numbers are no longer useful due to advancements gained by VoIP". A relative argument can be concluded as it relates to blacklisting as a singular approach to address the act of scam call detection. It is insufficient because it is unable to protect against new phone numbers or phone numbers not previously blocked. With the ability to easily generate a new phone number and obfuscate one's identity, the act of blacklisting would be several steps behind and unable to keep up with the speed in which new numbers are introduced into the telecommunication environment.

Shin et al. (2006) proposed the calculation of a gray level for callers based on past calls for scam prediction. Their mechanism is known as the Progressive Multi Gray-Leveling (PMG), which calculates the gray level of a caller. This gray level is considered the likelihood of a caller being a source of spam or not. While this mechanism does lack empirical data to determine effectiveness, the solution would not be able to apply to other telecommunication environments that extend beyond VoIP.

Pushkar et al. (2008) also explored a similar mechanism against the rise of scam calls via VoIP. They propose a Centralized-Session Initiation Protocol (SIP) approach, which leverages proxy servers to identify spammers' calls during the call establishment phase. Additionally, they use a decentralized referral social network, where users are assigned a reputation score by a given neighbor. This reputation, over time, would be used as an indicator to determine whether a call should be answered. This solution fails to address threat actor obfuscation through the use of faux numbers at scale using VoIP-based technology. This would enable a threat actor to be considered a new user with each call.

Chaisamran et al. (2011) presented a hybrid-based approach that applies a trust rating while also leveraging audio-based call features. The feature that is used is *call duration;* this offers a simple mechanism to automatically assign a trust value to a caller. (Chaisamran et al., 2011). The trust value decreases in instances of shorter calls and increases in instances of longer calls. Their approach achieves an accuracy of 98% and 95% in the detection of spam calls and in the detection legitimate calls, respectively.

2.5 Audio-based Scam Call Mitigations

The literature around scam call mitigation has evolved to explore the integration of audio. Lieto et al. (2019) are among the first to have investigated the distinction between human speech and bot speech, which could be useful when encountering scam-based robocalls. Their approach takes human audio and bot-generated audio using text-to-speech technology to create an audio spectrogram. An audio spectrogram, a visual representation of an audio signal, is fed into a Convolutional Neural Network for classification between the two (Lieto et al., 2019). The factors studied here are the types

of spectrograms generated (classic and mel) and their validation accuracy, as well as the types of bots used. The test bots used in their experiments include the Polly bot, an Azure bot, and a Watson bot, each with their highest classification accuracy being 98.12%, 74.29%, and 77.56%, respectively. When the solution is presented with unseen bot audio, which is converted into spectrograms, the neural network starts to generalize with an accuracy greater than 70%. While this work does offer one of the first audio-based approaches to combat bot-based calls, the critique of this method centers around how the bot-based audio is generated. Rather than using text-to-speech technology to generate the audio, utilizing in-the-wild bot audio from voicemails and other phone calls enables the creation of a dataset with a more accurate and unbiased representation of bot-generated audio from bad actors.

Researchers Gamage et al. (2023) also explored a similar approach that focused on voice feature analysis and the ability to distinguish between human and robotic voices. In their work, they used Natural Language Processing models (NLP), among other methods, to evaluate their detection performance. Their solution differentiated between humans and bots with an accuracy of 89.17%. Their work aids in expanding how natural language processing can be used to mitigate scam calls, specifically those that are made using bots; however, their approach requires active listening of potentially sensitive calls to decide whether a caller is by a human or a bot.

Elizalde et al. (2021) expanded the work in this area by using acoustic features of incoming voicemails to detect and distinguish between a robocall and a human call, as well as a human spam call vs a non-spam call. Spectro-temporal features of the audio from the voicemails were extracted to generate their dataset. This data was split into the

traditional 80-20% split for testing and training and fed into a Support Vector Machine (SVM) (Elizalde et al., 2021). The results of their work indicate an accuracy of 93.12% for the ability to distinguish between human and robocalls and a range of 75.86%-82.60% for the ability to distinguish between human spam calls vs. human non-spam calls. This work focuses more on the threat of robocalls/spam calls and how to mitigate them once the calls occur. The objective is to prevent the scam calls from occurring. This solution only allowed for the implementation of corrective controls that could be implemented in the future rather than offering a proactive solution that can immediately prevent scam calls. These calls would still persist as a nuisance to the average user, with the greater impact of financial loss and identity theft looming if a user answers the call.

Pandit et al. (2020) presented a solution within the space of audio-based detection for scam calling prevention. They introduced a virtual assistant called RobocallGuard, which was tasked with screening incoming calls to detect malicious calls and block them. The authors' primary goals included maintaining a user's normal mobile phone experience without interruption due to the use of the application. By employing the keyword spotting algorithm, SnowBoy, and the usage of audio transcription, they develop a mechanism that detects the right keywords, often the callee's name, within their response recognizer (Pandit et al., 2020). Their solution achieves 100% accuracy in detecting and blocking robocalls, with the virtual assistant being able to accurately label the robocalls 97.80% of the time. This solution is helpful with battling mass robocalls but is limited to specific hardware as this application, RobocallGuard, is currently operational only on Android devices. Furthermore, considering there is a screening mechanism where

a caller must share certain keywords to gain access to the callee, the authors also recognize that the solution cannot guarantee privacy.

Akinyemi et al. (2022) explored the use of audio data with supervised learning models, which include C4 Decision Trees and SVM, to prevent robocalls and related spam calls. They evaluate an approach that enables filtering between machine-based calls and human-based calls. The solution achieves an accuracy of over 99.15% in distinguishing between authentic phone calls and robocalls. However, a gap in the work is that there may be some overfitting, as the modification of the data can impact the structure of the model (Akinyemi et al., 2022).

Prasad et al. (2020) aimed to characterize robocalls through audio and metadata analysis. By developing a telephony honeypot, the researchers collect call audio from those spam calls. Key data they collected are call detail record (CDR) and SIP header logs. Their work ultimately revealed that there are, on average, over 4000 unsolicited spam calls per day; this value is relative to the metrics that will be discussed in Section 3 surrounding the data sources used for this praxis and the number of reported scam calls throughout the United States. Additionally, they highlight that the development of an anti-spam solution access to the call audio is crucial to understanding the intent of the call (Prasad et al., 2020). The use of call audio as a mechanism to classify a call as a scam call raises potential privacy and ethical concerns. In addition to this, spam audio-based solutions require the scam call to occur, which presents a situation where the solution is reactive rather than preventative and does not block the call from occurring.

Zhen and Gao (2023) further proved the point that operating with audio-based features requires potential scam calls to successfully go through, whether directly

answered or sent to voicemail, to function. In their suggested solution, they utilize call detail records for their features. More specifically, they use the voice record (VOC) characteristic, among other data points, to build their fraud telephone detector solution. This solution uses a Text2IMG method that leverages a Convolutional Neural Network with image augmentation. Their methodology achieves an F1-score of 89.80% and an AUC of 92.83% as their best score across varying training set sizes. Despite this performance, the solution was trained with a dataset specific to a competition, with little insight into its source and how it was generated and developed. This introduces a likelihood that the proposed solution may not be applicable to real-world scenarios.

Researchers Adam et al. (2023) also explored the use of phone call audio as input for scam call detection through the creation of their solution known as the Vishing Defender. Their system design is comprised of modules that include a data pre-processing step where phone calls undergo a transformation from speech to text, and then text analysis and pattern-matching stages occur. Their key objective was aimed at the detection of a vishing attempt and the reduction of processing time to make that determination. Approaches that are tested include cosine similarity, Knuth–Morris–Pratt (KMP), Natural Language Processing, and more. The Trie method provided the best total processing time to accuracy ratio with a pre-processing time of 0.000133 seconds and a processing time of .151 seconds with an accuracy of 70.90%. While their work offers quick response times in determining whether an active phone call is legitimate or not, the researchers admit to privacy concerns and accuracy concerns due to the limitations of the access to phone transcripts.

2.6 Machine Learning-based Scam Call Mitigations

Exploring the literature around machine learning-based approaches to resolve this scam phone call prediction problem reveals that the introduction of automated solutions with less reliance on having to wait for a scam call to be successful. Li et al. (2018) proposed a solution that does not require the use of the underlying phone network infrastructure. Their solution features TouchPal, an Android application that serves as their data aggregator where users label calls as malicious or not over a period of three months. This data was then used to pull 29 features with further ablation analysis revealing that the usage of 10 of those features, such as the date and time of the call, produces the same performance as leveraging all 29 (Li et al., 2018). They apply four machine learning models: Neural Networks, Random Forest, Support Vector Machine (SVM), and Logistic Regression. Their random forest model achieves an area-under-curve (AUC) score of at least 99.00%, indicating the classifier's high ability to distinguish between two classes. Furthermore, they effectively reduced 90.00% of malicious calls formerly considered unpreventable. This solution offers an effective mechanism to reduce malicious calls from occurring; however, the data with which the models are trained and tested are generated from users of the TouchPal application. There was a reliance on users to tag the calls in an effort to establish ground truth for the data, which leaves room for human error in the interpretation of what is considered a malicious call.

Hong et al. (2023) also explored the use of machine-learning models by exploring both natural language processing techniques, such as preprocessing text and audio-to-text conversion, in tandem with deep learning models that include Long Short-term Memory

(LSTM) and Convolutional Neural Networks (CNN). They utilize six features of call detail records to build their dataset. This is essentially the metadata of any given phone call. Their solution produces an accuracy of 85.61% with the LSTM neural network. While this solution does offer LSTM as a solution against malicious calls, as noted by the researchers, it is limited in its ability to detect emerging trends as the solution is trained with an initial dataset composed of generated scam vs non-scam audio.

Xing et al. (2020) presented a solution that makes use of both supervised learning algorithms and deep learning. Their approach, which uses call detail record features for their dataset, integrates automatic feature learning where the feature engineering process and its selection occur automatically without human intervention. Their solution achieves a detection accuracy of 99.70%, 99.00%, and 99.65% across CNN, LSTM, and Stacked Denoising Autoencoders (SDAE) models, respectively, for identifying malicious phone calls. This solution is among the first to have examined neural networks as a methodology to combat scam calls; however, as highlighted by the researchers, the dataset on which this model was trained and tested was imbalanced with a ratio of "8.2 million normal phone call samples and 8284 fraudulent phone call samples" (Xing et al., 2020). This results in a low precision and low recall score. Furthermore, among the features trained for this solution is a *call_duration* attribute. This attribute indicates that it requires a scam call to successfully go through, which has been previously described as being directly answered or sent to voicemail. The objective of the proposed solution in this praxis is to prevent a malicious call from ever having to be answered.

Gowri et al. (2021) proposed a solution that is able to predict malicious calls without accessing the underlying infrastructure by using time series data. They leverage

the Recurrent Neural Network (RNN) deep learning model and treat time steps as an input for the model. Their proposed methodology achieves an accuracy of 87.00%. This solution, which takes a novel approach with their data input, introduces a persisting issue. In an effort to generate the necessary data points required for the solution, a string of malicious calls need to occur to generate the time steps that are used as the data inputs for the model. This places users in a vulnerable position where they can become targets of these scam calls, potentially resulting in financial fraud and identity theft.

Malhotra et al. (2023) introduced the application of Artificial Intelligence to detect fraudulent phone calls. The researchers used a dataset of phone call recordings and specifically leveraged call detail record features, which include call duration, location, phone numbers, etc. The methodologies applied are Support Vector Machine (SVM) and Recurrent Neural Network (RNN) for training and testing. Their solution achieved an accuracy of 97.00% and a precision of about 97.60%. This solution proved the ability to achieve high accuracy in the detection of fraudulent calls. Similarly, Sandhya et al. (2020) generated a CDR-based dataset by pulling information from audio files and developing an SVM model for their methodology, achieving a precision of 97.56%. However, the approach highlighted in Malhotra et al. (2023) and Sandhya et al. (2020) requires a suspected scam call to go through, whether answered directly or sent to voicemail, for the necessary features to be captured. Allowing these calls to go through raises the risk that a user falls victim to fraud.

2.7 Government Laws & Mobile Network Operator Mitigations

Due to the ongoing nature of scam calls and mass unwanted calls, the United States government has attempted to intercede through the enactment of certain laws.

Among these laws is the Telephone Robocall Abuse Criminal Enforcement Deterrence (TRACED) Act. This act empowers the Federal Communications Commission (FCC) to enforce protections for phone users by requiring voice service providers and mobile network operators to adhere to certain best practices for caller authentication (United States Congress, 2019). The FCC has required voice service providers to implement the combined Secure Telephone Identity Revisited (STIR) and Signature-based Handling of Asserted Information Tokens (SHAKEN) protocols. The STIR protocol is used to identify the phone numbers for both the calling and receiving party, and the SHAKEN protocol assures end-to-end cryptography and protects the integrity of the information belonging to the originator of a phone call (Koilada, 2019). Collectively these protocols protect end-users by countering the spoofing of a phone number.

The implementation of the TRACED Act includes caller ID authentication tools, consumer protection tools, enforcement mechanisms, and annual reporting requirements to Congress (Consumer and Governmental Affairs, 2023). The STIR/SHAKEN protocols, which are a subset of the caller ID authentication tools, can include resources such as number or caller verification. Mobile network operators (MNOs) or voice service providers operating in the United States have started to meet the FCC requirements, but authentication of a phone number alone is not sufficient to prevent scam-calling. Furthermore, according to authors Wang et al. (2023), since the STIR/SHAKEN mandate, the protocol has not been as effective as anticipated, and the number of unwanted calls has increased in October 2022.

Mobile network operators (MNO) or mobile network carriers provide wireless communication services through their owned and operated telecommunication

infrastructure. Some of the most common MNOs that operate out of the United States are T-Mobile, AT&T, and Verizon. Each of these companies offers solutions that align with the STIR/SHAKEN protocols that are aimed at empowering their users to determine whether a call is a scam call or a legitimate call or to simply block unwanted callers. These features are known as Scam ID/Scam Block, Digital Phone Call Protect, and Call Filter/Call Filter+, respectively, and are detective and preventive controls. These solutions exist and offer an immediate mechanism to identify a scam call today; however, users are still likely to answer scam phone calls. Researchers Fischer et al. (2023) aimed to understand why people fall victim to fraudulent scam communications. They highlighted a psychological process that indicates "the communication medium and detail may increase the perception of trust and authority" (Fischer et al., 2023). This is evidenced by receiving phone calls with similar area codes. Despite the phone call being denoted as *scam likely,* there is a perception of familiarity and, thereby, trust that the present phone call is legitimate.

Ruan et al. (2019) propose a solution that is based on a cooperative fraud detection model. This model requires collaboration across multiple telecommunication operators in order to thwart threat actors from utilizing numerous mobile network operators to execute their objectives. This solution would entail the sharing of call detail records between operators with features such as duration, profiling the behavior based on the collected data, and using their match module mechanism based on Maximum Mean Discrepancy (MMD) to determine similarity in accounts between operator A and operator B (Ruan et al. 2019). The researchers run a simulation of their proposed solution using collected CDR features and are able to achieve an area under the curve of the receiver

operator characteristic (AUC-ROC) of 98.70%, which is indicative of the performance of the classification of malicious calls at different thresholds. While this solution does propose a novel approach to coordinate collaboration across operators to profile users and identify characteristics that are emblematic of similar accounts, there is a risk of infringement of privacy due to the nature of information stored by mobile network operators. The researchers addressed this by introducing noise into their match module method, but it likely has an impact on the results and performance of the simulation.

2.8 Summary and Conclusion

This literature review chapter presents a taxonomy of mitigation strategies against phone scam calls. Research in this area has been conducted over the last 15 years, with each solution evolving from the last to keep pace with which technology grows and the techniques of threat actors advance. Researchers Chu et al. (2023) also revealed, in analyzing the behavior of threat actors to telecommunication ecosystems, the three challenges that continue to make telecommunication fraud detection difficult. These challenges include the scale of telecommunication networks with the numbers of users and interactions occurring, the dynamic graph structure in which the network is constantly changing due to users joining and leaving a given network, and the obfuscation of threat actors where they mimic normal user behavior patterns and thwart detection methods. The literature reviewed highlights that some solutions are just frameworks of potential mitigations that lack empirical evidence to indicate their effectiveness. Other solutions present an ethical issue on user privacy and would likely not have widespread adoption due to potential infringement on regulations such as those related to the medical industry, such as the Health Insurance Portability and

Accountability Act, and educational fields, such as the Family Educational Rights and Privacy Act.

The current state of the art reveals three key things which indicate that the detection and prevention of telecommunication fraud as it relates to phone calls have evolved from blacklisting and assigning trust ratings. Evolving solutions to the scam call detection problem include methods that leverage audio-based detection; this method of identifying fraudulent phone calls has made remarkable strides but infringes on user privacy. More recent methods that have leveraged call detail records as input data for the development of machine learning models have achieved some of the highest accuracies, with some researchers reporting detection accuracies of over 99.00%; however, one of the most common attributes leveraged in these scenarios is *call_duration.* This feature creates a dependency where it is necessary for scam calls to be answered, whether directly answered or sent to voicemail, for the relevant characteristics to be collected. Furthermore, considering the requirements being imposed by the TRACED Act that enable the FCC to require voice service providers to utilize STIR/SHAKEN for call verification, researchers Edwards et al. (2020) have discovered that "answering a call is based on trust and expectations. Even if someone does not trust a call, they sometimes feel they have to answer it because it may be related to work, a job interview, or an expected callback" (Edwards et al., 2020). There exists a chance that end users will still answer a given phone call despite it not being marked as verified. In a focus group study conducted by Munyaka et al. (2020) reviewing user responses and expectations to Caller ID and other warnings for spam calling, some participants explicitly shared that despite a phone call possibly being a spam call, because they used to live in that state and

recognized the area code, they would still answer. Another participant shared that their family lived in a specific state, and they recognized the area code, so they would be comfortable with answering the call.

This review of the state-of-the-art highlights the need for a solution that overcomes the existing limitations of solving this telecommunication problem. Therefore, *A machine learning-based scam call categorization tool is required to reduce the likelihood of targeted social engineering attacks in the form of scam calls.*

Chapter 3—Methodology

3.1 Introduction

This chapter presents and details the methodology employed for predicting and categorizing scam calls. Using machine learning (ML), a mechanism is developed that accurately distinguishes between malicious calls and authentic calls and further categorizes those calls into six unique classes: Authentic, Medical, Utilities, Impersonator, Technical, and Financial.

The multi-pronged solution presented and evaluated in this praxis, as depicted in Graph 3-1, leverages supervised machine learning models, including Decision Tree and eXtreme Gradient Boosting. The XGBoost ML model, for example, is known for its efficiency, and near real-time applications in other related security areas, such as network intrusion detection, offer a quick approach to determine if an incoming call is legitimate (Dhaliwal et al., 2018).

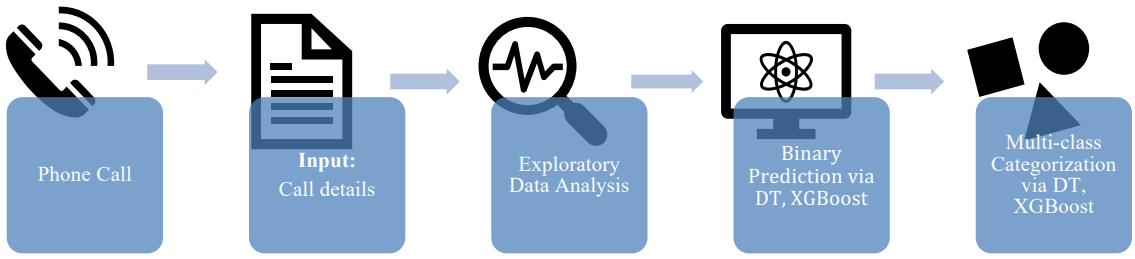

Figure 3-1. Scam Call Detection & Categorization Workflow

With the use of a supervised learning model, the dataset is formed with a ground truth that is used for evaluation. This ground truth, or the labels that indicate whether an entry is legitimate or malicious, will serve as the Y-axis or the labels for this model, while the features of each entry serve as the X-axis. The XGBoost model was developed and validated within a Jupyter notebook, which is an interactive Python notebook

(.ipynb) typically used for model development. Python, version 3, the programming language, is used as a native language for the Jupyter notebook, offering ease-of-use of libraries, including Pandas and Sklearn, and enabling flexibility during development.

The following section, Section 3.2, introduces the data sources used for this praxis and how that data was collected. Section 3.3 discusses how the dataset was developed, specifically highlighting the reason for and the process of data supplementation, as well as the features that the dataset is comprised of. Section 3.4 walks through the exploratory data analysis and its outcomes. Section 3.5 examines the multi-step process of data preprocessing, which includes data standardization, data encoding, feature selection, data balancing, etc. Section 3.6 analyzes selected models and their alignment with the research hypotheses presented. Section 3.7 concludes with an insight into the training and testing of the selected models.

3.2 Data Collection

The data used as one of the primary sources for this work derives from the Federal Trade Commission (FTC) office of the United States. The FTC regulates and enforces federal consumer protection laws that are aimed at the prevention of fraud, deception, and unfair business practices across various areas and industries, including communications. The data collected by the FTC is the *Do Not Call (DNC) Reported Calls Data,* which centers around malicious phone calls that users receive throughout the nation. Files are generated daily from Monday through Friday, specifying call details of those spam calls received for that day with an average of 4,000 entries daily. Key features offered or included in this data are *Consumer_City, Consumer_State, Recorded_Message_Or_Robocall,* and *Subject.* The *Consumer_City* and *Consumer_State*

features refer to the target destination of where the call was received. The *Recorded_Message_or_Robocall* is a feature used as an indicator of whether a recorded voicemail message was left for the receiver of said call. The *Subject* feature of this data indicates the primary objective of the threat actor, which can include calls aimed at reducing debt, home security and improvement, government impersonation, and other financial scams. This data is denoted throughout the rest of this chapter as the source for malicious calls.

The data used and denoted as the source for authentic calls are sourced from Alesco Data. This organization is a data company that specializes in providing targeted consumer data, data solutions, and technologies to marketing agencies, companies, and brands. As part of the efforts to develop a preventative solution for scam calls, privacy is at the forefront of this process. Only general data points were gathered when collecting authentic data from Alesco. Personally identifiable information, such as full name and street address associated with a given phone number, were omitted from the data retrieval. In order to ensure consistency between the features originally collected from the malicious data source, only similar features were pulled from the data source for authentic calls. These features also include city and state with *Recorded_Message_or_Robocall* and *Subject* fields being set to *"No"* and *"Authentic"*, respectively.

Although these features provide insight into targeted areas and highlight patterns of social engineering attacks to execute scam calls, they are inadequate. As highlighted in the literature review, the work conducted by Hong et al. (2023) only leveraged six call detail features; this resulted in the researchers only achieving an accuracy of 85.00% for

the performance of their machine learning model. With the aim of supplementing the initial call data retrieved, which is similar to the data used by Hong et al., and improving the accuracy of the proposed model, additional features were gathered. This was executed by running a reverse phone number lookup for each call entry across both the malicious and authentic calls. These additional features include carrier name, line type, and mobile network code. Discussion on the details of how the data was supplemented and how the final dataset was generated is highlighted in Section 3.3.

3.3 Dataset Development

The sources for the final dataset data are attained from the FCC and Alesco Data company, each representing malicious calls and authentic calls, respectively. There were over 15,000 call entries combined. This initial dataset only included seven unique features, including City, State, a flag for whether a recorded message was left, the ground truth label, and various other features. As highlighted in related work and earlier sections, the use of only a few call features to train a model has proven to be insufficient, as only an accuracy of 85.00% was able to be achieved (Hong et al., 2023). With the aim of ensuring considerable improvement in the performance of a given model, more features were required.

By leveraging a reverse phone number lookup Application Program Interface (API) from Twilio, each call entry was supplemented with four additional features. An API allows for programmatic access and communication between two applications. Twilio is a cloud-based communications company that provides programmable tools, with the reverse lookup API being among them for communicating via calls, texts, and other modes of contact. The reverse lookup API offered by Twilio enables the input of a

given phone number as a query and returns information about that inputted phone number. A Python script was developed leveraging the Twilio API client, the Pandas library, and the JSON library to query the phone number from each call entry. The Twilio API client enables the developed code to interact with their reverse lookup tool by iterating through a list of phone numbers. The Pandas library allows for the ability to read in the raw data and generate an output file with the data returned from the reverse lookup tool. The JSON library is used for the formatting of the data being aggregated from that particular data source. An additional parameter was used in the API call, titled *line_type_intelligence*, which is the key to providing supplementary information about a given number, including the line type such as mobile, landline, and fixed voice-over-IP. Once each phone number was queried, that information was outputted to a file and integrated into the dataset. This new dataset of 15,000+ call entries had grown from 8 features to 12 features, including the ground truth label.

The solution proposed to combat scam call detection requires the distinction between malicious and authentic calls, as well as the further classification of those malicious calls. Despite the results achieved for binary prediction after initial model development, as is discussed in later sections of the praxis, the features collected prove to be insufficient for multi-class classification. As a result, further data supplementation was necessary. The objective of further data supplementation is to gather features that are indicative of and unique to specific classes. The first two rounds of the data collection process only provided features that could shed insight into whether a given call was malicious or not; it lacked detail regarding any given class, such as any features that could distinguish a utility scam call from a financial scam call.

The Universal Licensing System (ULS), the source for additional supplementary data in the formation of this final dataset, is managed by the Federal Communications Commission (FCC); it is a portal that provides information about the licenses and permissions granted to businesses or individuals for the use of wireless technologies as it relates to communication across various mediums. The FCC, which is an independent U.S. government agency, regulates communications across radio, television, wire, and satellite. It also maintains jurisdiction over broadband access, radio frequency use, and public safety, among other areas. With this licensing system, telecommunication companies and mobile providers such as AT&T, T-Mobile, and Verizon indicate their intended use of a communication method and request permission for use.

The ULS data source offers information regarding the licensee and type of organization, that is, the radio service of use and the organization's regulatory status. These features provide specificity around the common use of a particular telecommunication agency or carrier. As a result, an additional five features were integrated across each call entry for the combined malicious and authentic calls. This final dataset of over 15,000 call entries, prior to the data analysis and data cleaning, now includes 17 unique features that offer insight, such as the target destination of these calls, the carrier used to make such calls, the validity of any licenses used, and more. In the following section, exploratory data analysis is conducted to clean the data and identify features that are relevant to the intended objective of binary prediction and multi-class classification.

3.4 Exploratory Data Analysis

Exploratory Data Analysis (EDA) allows for the discovery of patterns and a more in-depth understanding of a given dataset. It is a critical aspect in model development in

order to determine the data's quality and applicability for a given problem. Key steps involved and applied against the call dataset used in this praxis include initial data review, data cleaning, assessing feature importance, and conducting feature selection.

The EDA process begins with loading the dataset into the Python notebook for use. Initial statistics and metrics about the dataset are gathered by running the *shape()*and *head()* functions, which indicate how many rows and columns are included and display the first few lines of the dataset, respectively. Furthermore, the *describe()* method is used to gather related metrics such as the standard deviation and mean values. Following this, rows with empty values or NaN for features are dropped or removed using the *dropna()* method. This action is conducted to ensure that each call entry has the same available features with valid values in each field. Additionally, a column that is immediately dropped is the *Recorded_Message_or_Robocall* feature. The purpose of the proposed solution is to create a preventative mechanism rather than a reactive mechanism to ultimately stop a scam call from successfully going through or being answered. As highlighted in Appendix A, Table A-1, this feature is indicative of a scam call being answered; it would not be helpful to the objective of this solution as it requires the malicious phone call to fully occur to be deemed useful. Furthermore, a subsequent feature dropped from the dataset is the *Company_Phone_Number.* This column was removed to prevent any data bias in the development of the model; maintaining the phone number could create a condition of a partial look-up table where only pre-existing phone numbers can be identified rather than using the features of the calls to make a prediction for classification.

Categorical data encoding, a process conducted as part of the data processing and discussed further in a later section, is required for the development of the ML model. After conducting initial one-hot encoding, due to the existence of values with object data types rather than values with numeric data types, the feature list grew exponentially to over 2,000 features. This is primarily due to the variability of cities across the fifty states present in the data, which created dimensionality or a circumstance where there is an inverse relationship between the number of features present in the dataset and the performance of an ML model. Dimensionality reduction was required to ensure that the value of the original data was retained. This resulted in the removal of the *Consumer_City* feature from the dataset.

The Exploratory Data Analysis technique, as highlighted in Figure 3-2, Iteration Between EDA and Data Processing, becomes an iterative process that collaborates with mechanisms that occur during the data processing phase such as encoding, feature importance, and feature selection.

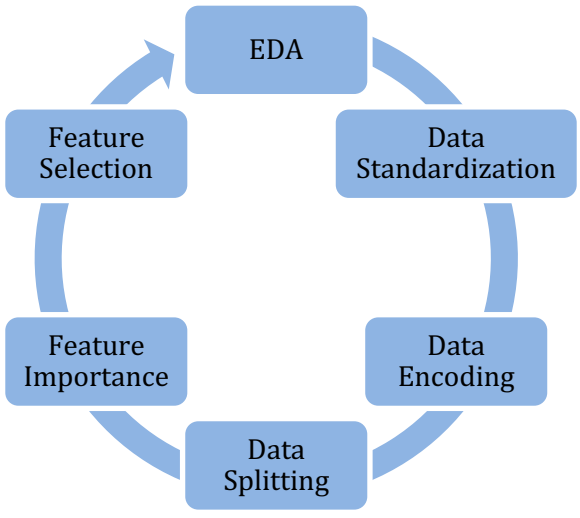

Figure 3-2. Iteration Between EDA and Data Processing

3.5 Data Preprocessing

Data preprocessing is a necessary step involved in model development as it prepares a given dataset for ingestion by a model. This phase of development allows for further data cleaning and transformation of the data for interpretability by selected machine learning models. The steps for data preprocessing conducted in this praxis include data standardization, categorical data encoding, and data splitting.

3.5.1 Data Standardization & Consolidation

Data standardization may be required during model development if values for the same characteristics or features are inconsistent with one another. Different values that are used to represent that same characteristic can create misinterpretation for the machine learning model where those values are unintentionally observed as a unique label. An example of this in relation to the call dataset is the reference values for the *Consumer_State* feature; there are entries that refer to the state of New York as *New York*, while other entries refer to the same state as *NY*.

Standardization is applied to the dataset by first filtering the data according to the state column. Once filtered, each state is listed by changing the state abbreviation, where necessary, to the full state name (e.g., CA to California).

Additional changes applied to the dataset include consolidation of certain *Subject* values. Consolidation is helpful to the cleaning of the data as it eliminates entries with any overlap. An example of this application includes consolidating the original subject values of *Reducing debt, Warranties,* and *Sweepstakes* into *Financial.* The call dataset originally had over ten subject classes that have been consolidated into six unique classes or categories.

3.5.2 Categorical Data Encoding

Data encoding is the process of transforming raw data into a readable format that can be interpreted by an algorithm. This applies in model development because raw data can include text, which is typically categorical in nature and has an object data type. Categorical data are features that aid in the classifying or grouping of data. An example of categorical data from the generated dataset used in this praxis is the feature *line_type. line_type as* described in Table A-1, the Dataset Feature List, indicates the type of phone line used to execute the phone call. Some of the labels available for this category are mobile, landline, and VoIP. These attributes group related data together.

The call dataset generated for this praxis is a mixed-data dataset comprised of both categorical and numerical input data. Due to the object data type being present in the dataset, which is commonly associated with categorical information, those data points are changed to a numeric data type via one-hot encoding.

One-hot encoding is an approach used to represent categorical data as numeric data. The benefit of using this method is that it can aid in preventing weight or hierarchical bias, which can occur when a feature is being encoded by attributing numbers to the values with minimal label variance, such as “1, 2, 3”; this creates an unintended or a false ordering of the labels for that given feature. The features in the call dataset with categorical data include *Consumer_City*, *Consumer_State*, and *Supplementary.carrier_name*, among others. As depicted in Table 3-1, these features such as *Consumer-State* are transformed programmatically using the one-hot encoding technique within the Jupyter notebook by using the *get_dummies()* method of the Pandas Python library.

Table 3-1. Categorical Encoding

Consumer_State
Arizona
Virginia
California
Massachusetts

State_Arizona	State_Virginia	State_Massachusets
1	0	0
0	1	0
0	0	0
0	0	1

3.5.3 Feature Selection

As highlighted and briefly discussed in an earlier section, the outcome of applying one-hot encoding across all the features with categorical data in the call dataset created a condition referred to as the curse of dimensionality. The curse of dimensionality is a case where the increased number of features impacts the performance of an ML model. In this scenario, there was an exponential growth to over 2,000 features due to the variability of cities present for each state, similarly as is seen in Table 3-1 where a single feature, state, becomes fifty unique features, necessitating the need for dimensionality reduction.

Dimensionality reduction is the process of minimizing the number of needed features for a training/testing dataset during model development. Reducing features may be necessary when forming the training and testing datasets because it allows the model to operate more efficiently by using features relevant to improve model performance. Due to the call dataset's high dimensionality, a modified dimensionality reduction technique, known as forward feature selection, was applied. Forward feature selection is the method of training a model by integrating a single feature at a time in an iterative manner to identify characteristics that produce improvement in model performance. The modified approach of forward feature selection applied in this study leverages feature importance to identify the top relevant features. In running feature selection, it is revealed

that the *Consumer_City* characteristic has an imperceivable percentage regarding relevancy to model performance. As a result, the *Consumer_City* feature is dropped and removed from the dataset. Additional features dropped from the dataset include *Created_Date and Violation_Date.*

3.5.4 Data Balancing

Data balancing is a technique applied to a dataset when the classes available are imbalanced. This is especially relevant during model development for multi-class categorization as a disproportionate dataset can introduce bias during the building of the model and impact its performance. To counteract this occurrence, an oversampling approach can be applied. An example of this is the Synthetic Minority Over-sampling Technique (SMOTE), which expands on simple oversampling by generating synthetic samples from the minority class (Dash et al., 2020). SMOTE, as a data balancing technique, is used to generate synthetic data points for the minority classes or the classes with the least representation in the dataset.

The call dataset's minority classes are related to the Utilities, Medical, and Technical -based scam calls. SMOTE is only applied to the training dataset, ensuring that the testing dataset remains unchanged as described by researcher Alsarhan (2023), by leveraging the python-based *imbalanced-learn* library. By integrating this technique, the minority classes are better represented, specifically, during the development of the training dataset which aids in improving the final model performance.

3.5.5 Data Splitting

Data splitting is the process of creating a distinct training dataset and a testing dataset for a given model. This allows the machine learning model to be trained on a

percentage of the overall data while the other portion is used for testing and validation. It is important to have separate datasets for this process to ensure the model's performance and results are not influenced by testing the same data points on which the model was trained.

Data splitting is applied to the call dataset by using a 70/30 split. 70% of the data is used to create the training dataset for the binary prediction and the multi-class prediction, and the other 30% of the data is used for the generation of the testing dataset for the same type of prediction.

3.6 Model Selection

Model selection is an imperative step in the model development process as it determines which algorithm best aligns with the problem being solved. It requires having a hypothesis that needs to be proven and an understanding of the data that is available to help solve that problem. Key research hypotheses that are proven in this work relate to the use of phone call metadata to improve prediction of scam calls and to accurately categorize the type of scam call. The models selected to substantiate that theory are Decision Tree and eXtreme Gradient Boosting.

The Decision Tree ML model is an efficient and effective approach for classification, allowing for ease of use when conducting iterative testing during model development (Azam et al., 2023). A decision tree utilizes a tree-like structure to model decisions and their outcomes. It is constructed from a root node that grows based on class decisions. A branch represents the value of a tested attribute, and a leaf node represents the final classification for the decision tree (Azam et al., 2023). This is useful considering

that the call dataset consists of categorical data, now encoded, that require a decision-making process to distinguish between each characteristic.

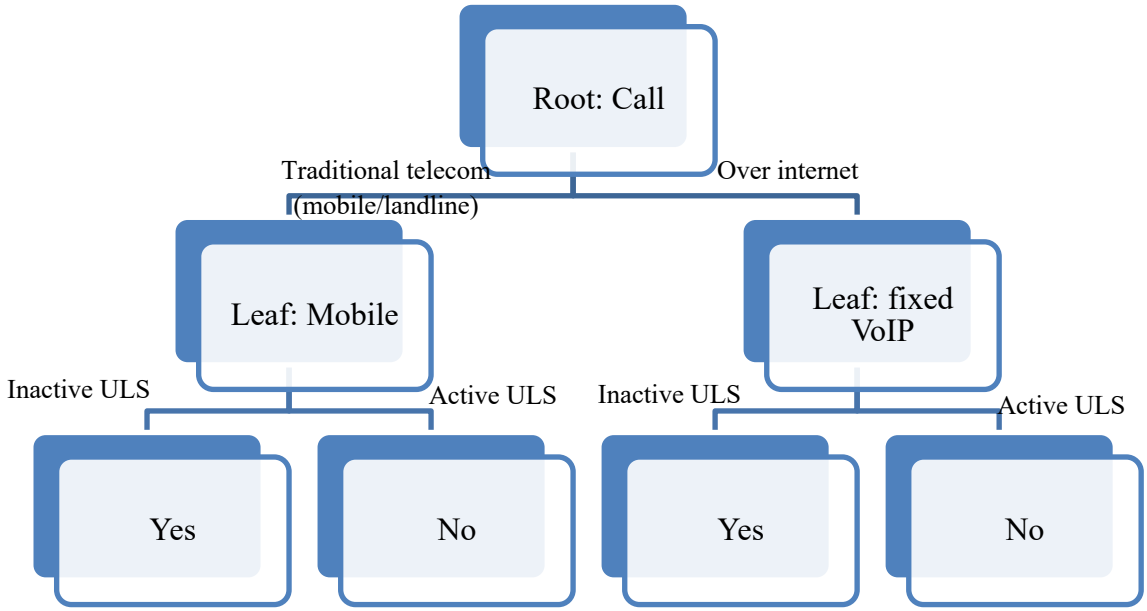

Figure 3-3. Example Decision Tree

The XGBoost ML model is a scalable and distributed gradient-boosted decision tree that is one of the most efficient models, performing ten times faster than existing methods (Dhaliwal et al., 2018). It has security applications relating to the classification of security threats for Internet of Things devices and intrusion detection of network systems. XGBoost models have the advantage over Decision Trees in that they leverage weak learners to improve their predictions. Furthermore, due to its capabilities, which include parallel processing, the ability to handle large amounts of data, platform portability, and programming language availability, this model is well suited to handle real-world classification problems similar to that of scam calling (Dhaliwal et al., 2018). XGBoost is applicable to the objective of the research hypotheses and the data available as it can handle fields or features with missing data, which is likely to occur when applying this solution in real-world scenarios or simulations. While XGBoost operates

well with real-world classification problems, some of its limitations center around performance impact when processing imbalanced datasets, as described by Zhang et al. (2022), which is further discussed in Section 4. This can apply to the work conducted in the praxis as the dataset constructed, prior to data balancing, is imbalanced.

3.7 Hyperparameter Tuning

A hyperparameter, as it relates to machine learning and model development, is a value that is used to control the learning process (Sipper, 2022). Hyperparameter tuning or optimization allows for the discovery of the most favorable parameter configuration to produce optimal results for a given machine learning model. Researchers Rimal et al. (2023) share a similar sentiment in the conclusion of their work that in order to achieve the most accurate model, it is imperative to invest time in hyperparameter fine-tuning. This refinement process can occur through iterative testing of various combinations of parameters until the highest performance for a model is achieved. Due to the seemingly endless possible number of combinations that can occur when manually tuning, studies that use ML models for their methodology, such as those of Sanchez et al. (2021) and Anyanwu et al. (2023), have favored the use of grid search for extensive parameter optimization in an automated manner.

Grid search is applied to this work to identify the ideal values to generate the best model performance. The hyperparameters used are the max_depth and min_samples_split for Decision Trees in addition to the n_estimators and max_depth parameters for XGBoost. They are evaluated using cross-validation. The max_depth hyperparameter defines the number of layers or the depth of the trees and determines the model's complexity. The min_samples_split defines the minimum number of samples that are

needed to split an internal node and is helpful against overfitting by limiting the creation of branches with few samples. The n_estimators parameter defines the number of trees used for the model; proper refinement of all these parameters is necessary to prevent underfitting or overfitting for the models.

3.8 Model Training & Testing

Model training and testing is the source of the efforts described throughout this chapter. This final step in the model development phase is typically centered around establishing a baseline for performance regarding accuracy and precision. This evolves to include iterative testing with techniques such as forward feature selection of relevant features as described in Section 3.5.3 to improve the model's performance. Using the 70/30 split described in Section 3.5.5, both the Decision Tree and XGBoost model are developed and trained by first creating the appropriate classifier using the *sklearn* and *xgboost* python libraries. The sklearn libraries are machine learning libraries used in Python for the creation of ML models that address certain objectives such as classification, dimensionality reduction, and more. Once a classifier is created for the selected model such as Decision Tree and XGBoost, the training of that model begins. These models are trained on the generated dataset using the *fit()* method. Evaluation of the predictions and the model's performance is conducted by generating a classification report that includes the overall accuracy, the precision between classes, recall, and f1-score for both binary prediction and multi-class prediction, accordingly. Further evaluation of the model's performance includes the usage of a confusion matrix to understand model training and cross-validation to prevent the condition of overfitting.

Chapter 4—Results

4.1 Introduction

This chapter analyzes the outcome of the model development and experimentation detailed in Section 3. In order to identify the optimal solution, the model performance of two models, Decision Tree and eXtreme Gradient Boosting (XGBoost), across binary prediction and multi-class classification is evaluated by gathering and comparing model metrics such as accuracy and precision.

This chapter is organized into the following order: 4.2 Data Collection, 4.3 Dataset Development, 4.4 Exploratory Data Analysis, 4.5 Data Preprocessing, and 4.6 Model Training & Testing. Section 4.2 details information related to the data collected for the development of the dataset, including source, quantity, and data treatment required. Section 4.3 showcases the iterative approach taken to supplement the existing information and attributes used in the foundational dataset. Section 4.4 dives into the exploratory data analysis of the aggregated data used for model development by offering supplemental tables and graphs. 4.5 breaks down the steps involved in data preprocessing, which include data standardization, categorical data encoding, feature selection, and more. Section 4.6 demonstrates how the models were developed, evaluated, and validated, in addition to the analysis of their performance.

4.2 Data Collection

The data sources leveraged as part of the data collection process are from the Federal Trade Commission (FTC) office of the United States and Alesco Data. They are used to collect malicious and authentic calls, respectively. The following table, Table 4-1, further describes those sources. Moreover, Table 4-2 describes the data sources used for

the data supplementation of the initially developed dataset. Those data sources listed are in alignment with and are used to prove the Research Hypothesis H1 defined for this praxis, *Phone call metadata improves the performance of a predictive model for scam call detection*. The original dataset, comprising six features, is supplemented using a reverse phone number lookup via Twilio and is further supplemented using information collected from the FCC-managed Universal Licensing System (ULS).

Table 4-1. Data Source Details

Dataset Name	**Do Not Call (DNC) Reported Calls Data**	**Alesco Phone ID Database**
Source Link	Federal Trade Commission (https://www.ftc.gov/policy-notices/open-government/data-sets/do-not-call-data)	Alesco Phone ID Database (https://datarade.ai/data-products/alesco-phone-id-database-the-industry-s-largest-and-most-ac-alesco-data)
Brief Description	This dataset includes Do Not Call and Robocall complaints reported directly to the FTC on a weekly basis.	This dataset provides valid phone numbers with representation across all 50 states.
Format	.csv	.csv, .xls, .txt
Time Span	Oct 2023 – Dec 2023	N/A
Number of Records	4000+ daily	9000
Purpose	This dataset will be used to aggregate malicious phone calls. It provides insight into the tactics and techniques employed to conduct social engineering attacks such as scam calls.	This dataset will provide valid U.S phone numbers that are representative of authentic phone calls.
Data Treatment	**Data imputation** for missing derivable values (i.e., area code provided but consumer city missing) **Data filtering** to remove data points missing key information like location or phone number. **Data aggregation** to centralize target data points across a few months.	**Data filtering** to remove unnecessary features to create consistency with FTC data

Table 4-1 details the sources used to develop the foundation of the dataset that is comprised of malicious and authentic calls. As discussed in Section 2 and Section 3, some of the prior work conducted aimed at scam call detection has only been able to achieve an accuracy of about 85% when leveraging six features of the Call Detail Record (CDR) information (Hong et al., 2023). In an effort to improve the accuracy of scam call detection and also categorize those scam calls, the H1 and H3 research hypotheses of this praxis were formulated, and the following data sources outlined in Table 4-2 were used for supplementation of the foundational dataset.

Table 4-2. Supplementary Data Source Details

Name	Twilio Reverse Phone Number Lookup	FCC Universal Licensing System
Source Link	https://www.twilio.com/code-exchange/lookup	https://wireless2.fcc.gov/UlsApp/UlsSearch/searchLicense.jsp
Brief Description	Reverse Phone Number Lookup with the use of line type intelligence provides carrier information and line type data.	The ULS portal provides information about the licenses and permissions granted to businesses or individuals for the use of wireless communication technologies
Format	API	N/A
Number of Records	Based on number of entries for foundational dataset	Based on number of entries for foundational dataset
Purpose	This data source will provide additional phone call metadata to supplement the existing dataset.	This data source will provide supplementary information about the carriers of each phone call entry.
Data Treatment	**Data aggregation** to integrate the additional features to the existing dataset	**Data aggregation** to integrate the additional features to the existing dataset

As briefly described, the data sources noted in Table 4-2 are used as supplementary attributes in conjunction with the H1 and H3 hypotheses. H1 states that the use of phone call metadata will improve accuracy performance, while H3 states that

the use of the developed model will accurately categorize or classify an incoming scam call. The features offered by the reverse phone lookup provide insight into the phone carrier and line type. This information is indicative of the mode or specific medium, in addition to the mobile network operators being used by a threat actor to execute their scams. The features offered by the ULS portal provide an understanding of the posture of the telecommunication companies and mobile providers as it relates to permissions to operate over broadband, wire, radio, and satellite, among other communication mediums. This is characteristic of their operation processes and their validity as a telecommunication provider.

4.3 Dataset Development

The attributes associated with each data source listed in Table 4-1 and Table 4-2 are specified and described in Section 3.2.

In the development of the phone call dataset, the objective has been aimed at creating consistency across all entries for authentic and malicious phone calls. This is established by ensuring the same properties or features are used for both data sources. The foundational CDR features collected across both classes are *phone number, city, state, area code, subject,* and *recorded message.* Following the first round of data supplementation via Twilio's reverse phone look-up API, the carrier'*s name, mobile country code, mobile network code*, and *line type* attributes are integrated. Subsequently, the final round of data supplementation using information derived from FCC's ULS system includes *active license, foreign ownership, licensee type, radio service*, and *regulatory status.*

Table 4-3. Dataset Construction

Company Phone Number	Consumer_ City	Consumer_ State	Consumer_ Area_Code	Subject
4806901334	phoenix	Arizona	602	Financial
9288323036	phoenix	Arizona	602	Financial
9799774131	Tucson	Arizona	602	Financial
6174585687	Roxbury	Massachusetts	617	Utilities
6177746518	Roxbury	Massachusetts	617	Medical
4084029160	San Jose	California	408	Impersonator
6177081989	Roxbury	Massachusetts	617	Medical

Supplementary. carrier_name	Supplementary. mobile_country _code	Supplementary. mobile_network _code	Supplementary. type
Twilio - SMS/MMS-SVR	311	950	0
Twilio - SMS/MMS-SVR	311	950	0
IP Horizon LLC	310	0	2
Onvoy, LLC	313	987	0
MCI Metro, ATS, Inc.	310	0	2
Level 3 Communications, LLC	310	0	1
Comcast	313	985	1

ULS_Active_ License	Foreign_ Ownership	Licensee Type	Radio Service	Regulatory Status
1	0	0	0	0
1	0	0	0	0
0	0	2	2	2
1	1	1	1	1
0	0	1	3	1
1	0	1	0	0
1	0	0	0	0

Table 4-3 provides a visual breakdown of the foundational data set alongside the supplementary attributes collected during the reverse phone lookup process and the ULS portal review, resulting in 15+ features for the final dataset.

4.4 Exploratory Data Analysis

The Exploratory Data Analysis process provides insight into the dataset during model development. Among the various steps that are conducted during this phase, the first step taken was to run the *describe()* method against the dataset to collect related metrics. Table 4-4 reveals metrics such as count, mean, standard deviation, and more to

help define the contents of the dataset. Initial observation of these metrics, particularly the min and max of the *consumer_area_code, mobile_country_code*, and *mobile_network_code* attributes, reveal that the area codes range from 9 to 989, the mobile country codes range from 302 to 338, and the mobile network codes range from 0 to 994, respectively.

Two features that are immediately dropped are the *Company_Phone_Number* and the *Recorded_Message_or_Robocall* attributes. The phone number column is removed to prevent any bias and the unintended creation of a look-up table. The recorded message attribute is removed to eliminate any data that may be indicative of a scam call being answered by a user. As previously discussed, the intention of this solution is to propose a mechanism that detects and categorizes scam calls *without* the call needing to be answered. The recorded message feature is predicated on a scam call being answered and is, therefore, not needed.

As the EDA phase continues, the next step is understanding how the distinct classes or categories are represented. Figure 4-1 depicts the distribution of the data across the six total classes, which include the Authentic, Medical, Utilities, Financial, Technical, and Impersonator classes. As a consequence of the *Subject* column consisting of data points with object data types, the process of data encoding is applied. Additional information regarding the steps involved in encoding the data is highlighted in Section 3 and detailed in future sections.

Table 4-4. Dataset Metrics

Features	Count	Mean	Standard Dev
Company_Phone_Number	11031	5809382572.98	2446724304.17
Consumer_Area_Code	11031	586.49	240.95
Subject	11031	1.01	1.65
Recorded_Message_Or_Robocall	11031	0.29	0.46
Supplementary.mobile_country_code	11031	310.40	1.40
Supplementary.mobile_network_code	11031	130.67	327.10
Supplementary.type	11031	1.15	0.92
ULS_Active_License	11031	0.28	0.45
Foreign_Ownership	11031	0.43	0.73
Licensee Type	11031	0.74	0.87
Radio Service	11031	2.74	4.10
Regulatory Status	11031	1.11	1.01
Label	11031	0.33	0.47

Features	Min	25%	50%	75%	Max
Company_Phone_Number	2012007066	3346826774	6064751427	8035742596	9898970463
Consumer_Area_Code	9	352	609	803	989
Subject	0	0	0	2	5
Recorded_Message_Or_Robocall	0	0	0	1	1
Supplementary.mobile_country_code	302	310	310	310	338
Supplementary.mobile_network_code	0	0	0	0	994
Supplementary.type	0	0	1	2	3
ULS_Active_License	0	0	0	1	2
Foreign_Ownership	0	0	0	1	2
Licensee Type	0	0	0	2	6
Radio Service	0	0	2	3	45
Regulatory Status	0	0	1	2	7
Label	0	0	0	1	1

To better understand Figure 4-1, Table 4-5 can be used as a legend; it maps a given category to its encoded value. Using this table, it is clear that Figure 4-1 depicts that the authentic calls are well represented with a count exceeding 7000 data points. Minority classes, such as Utilities and Technical, are the least represented. This

distinction in the representation of data is consistent with real-world scenarios where more valid calls are processed in comparison to malicious calls.

Table 4-5. Feature Mapping

Category Name	Category Value
Authentic	0
Medical	1
Utilities	2
Financial	3
Technical	4
Impersonator	5

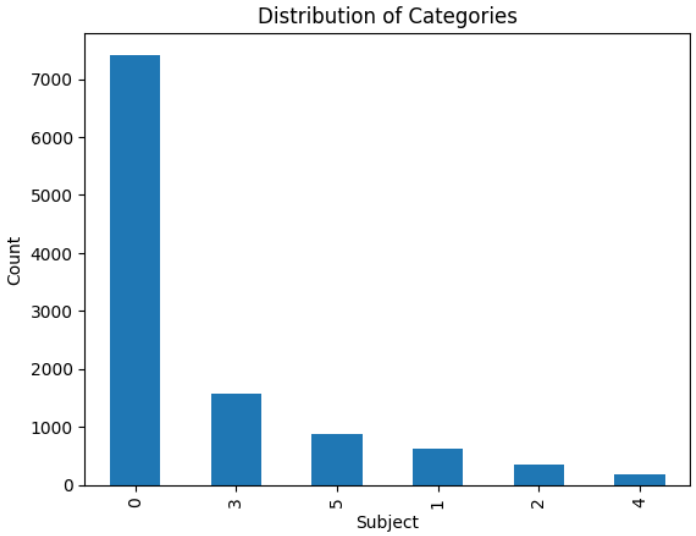

Figure 4-1. Category Distribution

As previously mentioned, the dataset is comprised of a mix of data types. In order to prepare the dataset for model training and testing across various models, the data types must be consistent. Listed below in Table 4-6, each feature is followed by its original data type. Any attribute denoted with an object data type has undergone the process of data encoding.

Table 4-6. Feature Data Types

Features	Data Type
Company_Phone_Number	int64
Consumer_City	object
Consumer_State	object
Consumer_Area_Code	int64
Subject	object
Recorded_Message_Or_Robocall	int64
Supplementary.carrier_name	object
Supplementary.mobile_country_code	int64
Supplementary.mobile_network_code	int64
Supplementary.type	int64
ULS_Active_License	int64
Foreign_Ownership	int64
Licensee Type	int64
Radio Service	int64
Regulatory Status	int64
Label	int64

4.5 Data Preprocessing

As depicted in Chapter 3 by Figure 3-2, the data preprocessing phase is an iterative process alongside EDA. Data is prepared for better interpretability by ML models. In the work conducted in this praxis, this phase includes data standardization and consolidation, categorical data encoding, data balancing, and data splitting.

Data standardization allows for establishing consistency among characteristics and their values. Introducing different values that signify a similar meaning for the same characteristic can create a condition where the values for that feature are processed and observed as unique labels. During the dataset construction, this is seen particularly when dealing with state names and state abbreviations. One data point may list the value for the *Consumer_State* feature as AZ while another may refer to the same state with the state's

full name, Arizona. These two features are providing the same information, so maintaining only one of them should be sufficient. Consolidation is necessary to remove any data entries that may have similarities or overlap. Categorical data encoding enables data with numerical data types to be converted and ingested during the model development phase. Data balancing is the process of creating proportionality for an imbalanced dataset. It helps in developing a more balanced dataset for processing during model development. Data splitting is one of the final steps conducted prior to model selection, training, and testing. This is the division or separation of the developed call dataset into a training set and a testing set.

The following sections provide a comprehensive review of what the aforementioned steps of data preprocessing entail.

4.5.1 Data Standardization & Consolidation

Data standardization for the call dataset primarily occurs for the *Consume_State* feature, where some entries refer to a state by its abbreviation while other entries refer to the same state by its full name. Data points that refer to a state by its abbreviation are transformed to the full state name. Table 4-7 provides an example of what that change looks like within the call dataset.

Table 4-7. State Abbreviation Standardization

State Abbreviation	Full State Name
AZ	Arizona
MA	Massachusetts
CA	California
NY	New York

Data consolidation within the focus of this praxis is aimed at reducing the number of categories where overlap may exist. As reviewed in Section 3.5.1, an example of this process occurs for some of the original *Subject* values that were present in the call dataset; the values of this attribute represent the possible classes for the multiclass categorization. Table 4-8 displays all the original Subject values and the classes into which they have been consolidated.

Table 4-8. Subject Consolidation for Multiclass Categorization

Official Category	Original Category Name(s)
Medical	Medical & Prescription
Utilities	Energy, Home security, Home improvement
Financial	Reduce debt, Work from home, Warranties, Sweepstakes, Charities
Technical	Computer/tech support
Impersonator	Calls pretending to be government, business, or family and friends

4.5.2 Categorical Data Encoding

Machine learning models require a readable format for interpretability when processing data. The method of encoding is a mechanism that enables this data transformation. In this praxis, two approaches are considered and explored to conduct encoding: one-hot encoding and the factorize() method.

As outlined in previous sections, when leveraging one-hot encoding across all applicable features, this introduced dimensionality to the dataset at an exponential level. This was particularly evident for the *Consumer_City* attribute where the list of features grew to over 2,000. The following figure, Tables 4-9 through 4-11 showcase some of the transformations conducted for the applicable categorical data points.

Table 4-9. One-hot Encoding for *Carrier_Name* Attribute

Supplementary. carrer_name	carrier_name _123.Net, Inc	carrier_name_Twilio - SMS/MMS-SVR	carrier_name_ IP Horizion LLC	carrier_name_ Onvoy, LLC
123.Net, Inc	0	0	0	1
Twilio - SmS/MMS-SVR	0	1	0	0
IP Horizion LLC	0	0	0	0
Onvoy, LLC	0	0	0	0

Table 4-10. One-hot Encoding for *Consumer_City* Attribute

Consumer_City	City_Phoenix	City_Tucson	City_Roxbury	City_Louisville
Phoenix	0	0	1	0
Tucson	1	0	0	0
Roxbury	0	0	0	0
Louisville	0	0	0	0

Table 4-11. One-hot Encoding for *Consumer_State* Attribute

Consumer_State	State_Arizona	State_Virginia	State_Massachusets	State_Kentucky
Arizona	0	0	0	0
Virginia	0	0	0	0
Massachuseets	0	0	0	0
Kentucky	0	0	0	0

Due to the high variability introduced when using the one-hot encoding method against the developed call dataset, the encoding method transitioned to a mechanism that still enabled the transformation of data without introducing new features. The factorize technique was applied across all features as the alternative encoding method. This was executed using the *factorize()* method supported by the pandas library. The following

figure demonstrates how the data transitioned from a dataset with heterogeneous data types to a dataset with a homogeneous data type of integer-based values.

Table 4-12. Data Prior to Factorize Encoding

Consumer_State	Consumer_Area_Code	Supplementary.carrier_name	Supplementary.mobile_country_code
Arizona	602	Twilio – SMS/MMS-SVR	311
Arizona	602	Twilio – SMS/MMS-SVR	311
Massachusetts	617	Onvoy, LLC	313
Massachusetts	617	MCI Metro, ATS, Inc.	310

Table 4-13. Data After Factorize Encoding

Consumer_State	Consumer_Area_Code	Supplementary.carrier_name	Supplementary.mobile_country_code
0	0	0	0
0	0	0	0
1	1	2	2
1	1	3	1

As demonstrated in Tables 4-12 and 4-13, this method of encoding allows for consistent transformation of all data types. In the scenario of this phone call dataset, that applies to both the categorical and numeric data. Furthermore, the usage of this approach prevents the condition of dimensionality from occurring.

4.5.3 Feature Selection

The decision to remove or drop certain features while maintaining others, especially when facing the condition of dimensionality, is made possible by a modified dimensionality reduction method known as forward feature selection. This mechanism, as described in Section 3, is the process of determining the features that influence model performance by training the model on a single feature at a time. In this praxis, a modified

approach is employed that leverages feature importance. An example of how feature importance is used in the decision-making process is depicted in Tables 4-14 through 4-17.

Table 4-14. Feature Importance for Decision Tree Binary Prediction

(using one-hot encoding)

Feature	Importance
Supplementary.mobile_network_code	0.41
Licensee Type	0.14
carrier_name_IP Horizon LLC	0.08
carrier_name_Peerless Network	0.05
state_California	0.05
. . .	. . .
city_East New Market	0.00
city_East Northport	0.00

Table 4-15. Feature Importance for XGBoost Binary Prediction

(using one-hot encoding)

Feature	Importance
Supplementary.mobile_network_code	0.32
carrier_name_Commio LLC	0.07
carrier_name_Mediacom Communications Group	0.06
EON Telecom, Inc.	0.04
carrier_name_Peerless Network	0.04
. . .	. . .
city_Elberton	0.00
city_Elbow Lake	0.00

Table 4-16. Feature Importance for Decision Tree Multiclass Prediction

(using one-hot encoding)

Feature	Importance
Supplementary.mobile_network_code	0.23
Licensee Type	0.09
Consumer_Area_Code	0.06
carrier_name_Peerless Network	0.05
carrier_name_IP Horizon LLC	0.04
. . .	. . .
city_Emerald Isle	0
city_Emerson	0

Table 4-17. Feature Importance for XGBoost Multiclass Prediction

(using one-hot encoding)

Feature	Importance
carrier_name_Cincinnati Bell	0.07
carrier_name_Suddenlink Communications	0.03
city_Lucedale	0.01
city_Portland	0.01
city_Elgin	0.01
. . .	. . .
city_Eustis	0
city_Eutawville	0

As identified in Table 4-14 and Table 4-15, *Consumer_City,* which becomes *city_NAME_OF_CITY* after one-hot encoding, has an imperceivable percentage of importance despite a display of 6 significant values across the attributes for binary prediction. In Table 4-16 and Table 4-17, while there may be a few cities that are included in the top five important features, more than 2,000 cities still have an

imperceivable percentage of importance against the select ML model, despite a display of 6 significant values. This indicates that the feature relating to the cities across the United. States has no influence on model performance for multiclass prediction.

Table 4-18 and Table 4-19 offer a breadth of view into the distribution of the feature importance for binary prediction among the remaining features for both a Decision Tree classifier and an XGBoost classifier. The *carrier_name* and *mobile_network_code* features statistically play the greater significance regarding model performance. With each appearing in the top three features for each model.

Table 4-18. Feature Importance for Decision Tree Binary Prediction

(using factorize encoding)

Feature	Importance
Supplementary.carrier_name	0.61
Supplementary.mobile_network_code	0.25
Consumer_State	0.06
Regulatory Status	0.02
Licensee Type	0.02
Supplementary.type	0.02
Consumer_Area_Code	0.01
Foreign Ownership	0.00
Radio Service	0.00

Table 4-19. Feature Importance for XGBoost Binary Prediction

(using factorize encoding)

Feature	Importance
Supplementary.mobile_network_code	0.84
Supplementary.carrier_name	0.06
Supplementary.mobile_country_code	0.03
Regulatory Status	0.02
Licensee Type	0.01
Supplementary.type	0.01
Consumer_State	0.01
Foreign Ownership	0.01
ULS_Active_License	0.01
Radio Service	0
Consumer_Area_Code	0

Table 4-20 and Table 4-21 display the feature importance across the different models and the different prediction cases. Once the *Consumer_City* attribute is dropped and *factorize()* is leveraged as an alternative to one-hot encoding, there is a better distribution for the remaining features and their relevancy to the performance of a selected model.

Table 4-20. Feature Importance for Decision Tree Multiclass Prediction

(using factorize encoding)

Feature	Importance
Supplementary.carrier_name	0.55
Supplementary.mobile_network_code	0.20
Consumer_State	0.09
Consumer_Area_Code	0.08
Regulatory Status	0.03
Supplementary.Type	0.02
Radio Service	0.01
Licensee Type	0.01
Supplementary.mobile_country_code	0.00
Foreign_Ownership	0.00

Table 4-21. Feature Importance for XGBoost Multiclass Prediction

(using factorize encoding)

Feature	Importance
Supplementary.mobile_network_code	0.30
Supplementary.carrier_name	0.20
Licensee Type	0.12
Supplementary.Type	0.07
Regulatory Status	0.07
Radio Service	0.06
Consumer_State	0.04
ULS_Active_license	0.04
Foreign Ownership	0.04
Consumer_Area_Code	0.03
Supplementary.mobile_country_code	0.03

For Tables 4-20 and 4-21, *carrier_name* and *mobile_network_code* stand out as primary attributes that influence model performance. Common features between the

Decision Tree and the XGBoost classifiers that have a significant impact regarding model performance include *carrier_name, mobile_network_code*, and *type.*

The Decision Tree classifier favors the *Consumer_Area_Code* and *Consumer_State* attributes in comparison to the XGBoost classifier which favors the *Radio_Service* and *Licensee Type* attributed for multiclass classification. This observation plays an important part in the comparison of the performance of the models and will be further discussed in Section 4.6.

4.5.4 Data Balancing

The nature of the scam call detection problem in the real world is that while scam calls are persistent, most of the phone calls that are received by a user are legitimate (Xing et al., 2020). The call dataset developed in this praxis mimics this reality with a greater representation of authentic calls in comparison to the scam calls. This results in the dataset being imbalanced. This is more apparent during the model development for multiclass categorization. In order to rectify data imbalance and prevent any bias in model performance, the Synthetic Minority Over-sampling Technique (SMOTE) method, an oversampling technique used for data balancing, is utilized. As depicted in Figure 4-12, the minority classes, as it pertains to the multiclass classification, are Utilities, Medical, and Technical. This means that the representation of these classes or the number of data points for each of these classes are weaker, in contrast to the dominant class. The following graph, Figure 4-13, reproduces the distribution of the classes for the phone call dataset with the SMOTE technique applied.

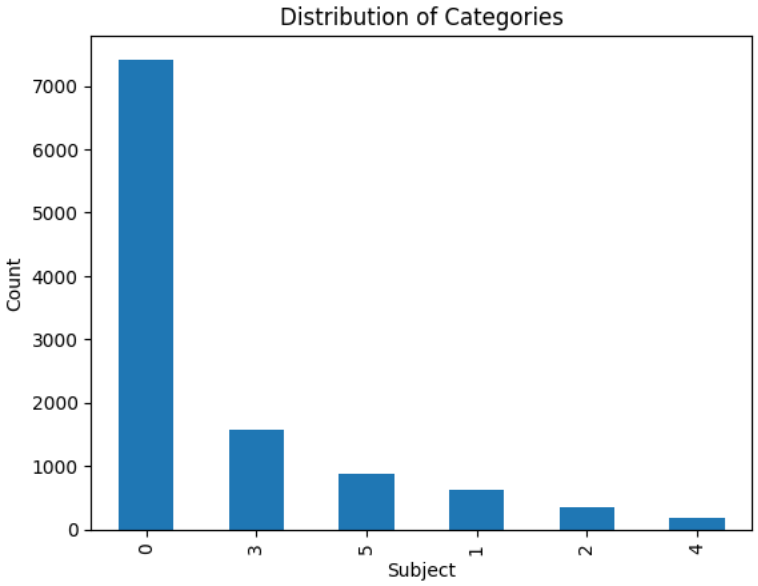

Figure 4-1. Original Category Distribution

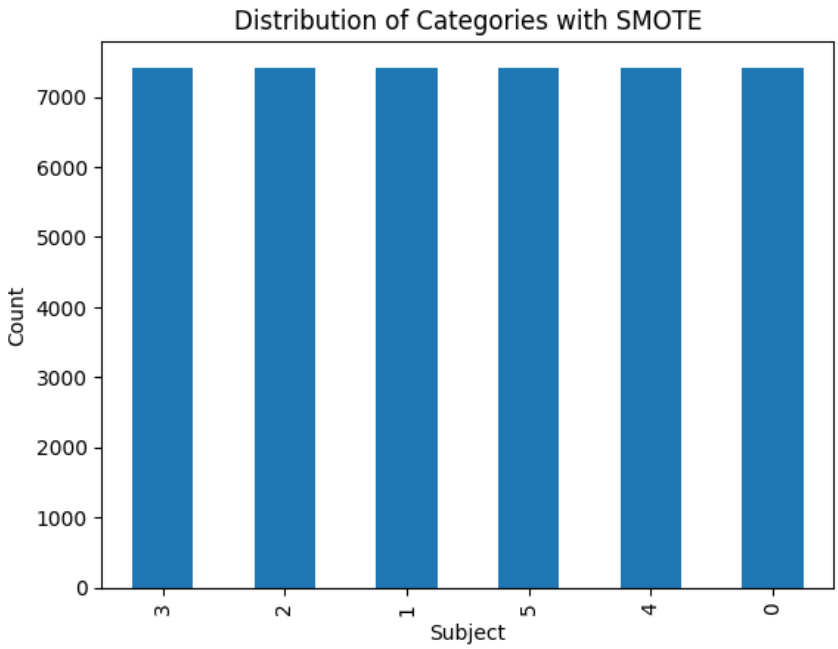

Figure 4-2. Category Distribution with SMOTE

Figure 4-2 showcases that all classes from which the call dataset is comprised have equal distribution or equal data points. This will help in preventing any bias towards

a particular category during classification, which can often occur due to the presence of more data points, because all categories will have equal representation.

4.5.5 Data Splitting

Using the refined phone call dataset, data splitting is conducted to begin training the selected models across the two prediction cases, which are the binary prediction and the multiclass classification. A split of 70/30 is used where 70% of the refined dataset is utilized for the creation of the training set, and 30% is used for the testing set. This split is applied to the dataset before oversampling and after oversampling. The results of the model performance in both cases will be analyzed in Section 4.6.

Table 4-22. Dataset Split

	Training Dataset	Testing Dataset
Number of Features	11	11
Number of Phone Calls	7721	3310

Table 4-22 lists the count for features in the training dataset as compared to the testing dataset. This is consistent for both the Decision Tree classifier and the XGBoost classifier for binary prediction and multiclass classification.

4.6 Hyperparameter Tuning

Achieving optimal model performance is necessary to determine the maximum capability of a given model to solve a specific problem. In this praxis, the objective is to detect voice phishing in the form of scam calls and categorize them accordingly. During the model development process, the models are tuned using grid search with cross-validation against two hyperparameters for Decision Tree: max_depth and

min_samples_split and two hyperparameters for XGBoost: max_depth and n_estimators using the logarithmic loss function (log loss function) and the accuracy scoring metric. The log loss function is used to understand how tuning these various hyperparameter configurations causes the model to perform as is similarly described by Kumar and Sastry (2018), where a loss function helps evaluate the performance of a model classifier. The accuracy scoring metric for grid search specifically evaluates how well-selected values for the model's hyperparameters perform. Both the log loss function and accuracy scoring serve as metrics to identify the optimal hyperparameters for the Decision Tree and XGBoost models.

In this praxis, the accuracy scoring metric for grid search better identifies those optimal values for the hyperparameters of Decision Tree and XGBoost models. Figure 4-3 through Figure 4-6 depict the output of the accuracy metric indicating the most favorable values for each of those model's hyperparameters. Figure 4-4 highlights the results of the grid search using that metric. The hyperparameter tuning results for binary classification using a Decision Tree model are 16 for max_depth and 14 for min_samples_split, with the values for the parameters of the XGBoost model being 5 for max_depth and 300 for n_estimators. The hyperparameter tuning results for multiclass classification using a Decision Tree model are 13 for max_depth and 14 for min_samples_split, with the values for the parameters of the XGBoost model being 3 for max_depth and 600 for n_estimators.

The output of the logarithmic loss function for Decision Tree and XGBoost for both binary and multiclass prediction are listed in Appendix A.

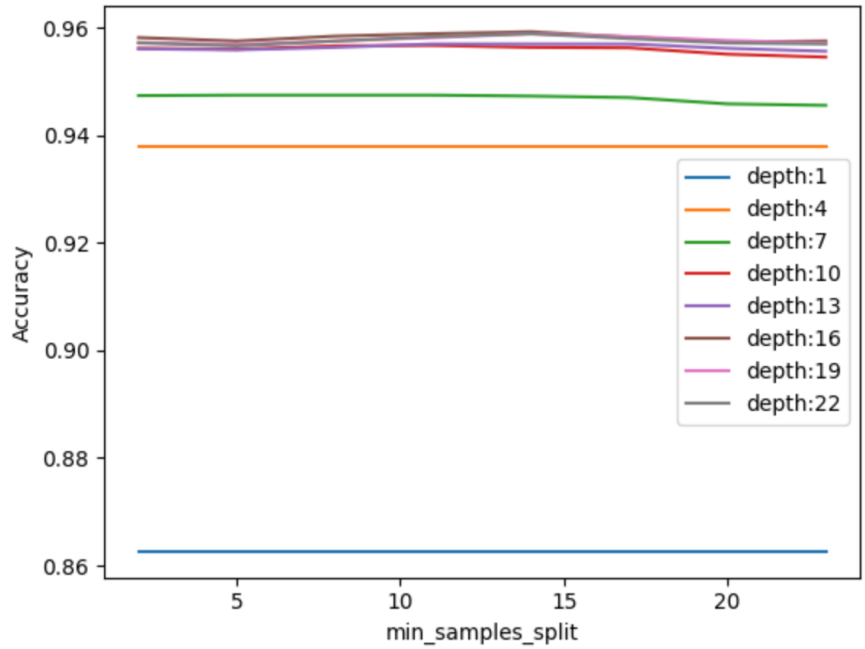

Figure 4-3. Hyperparameter Tuning for Binary Classification using Decision Tree

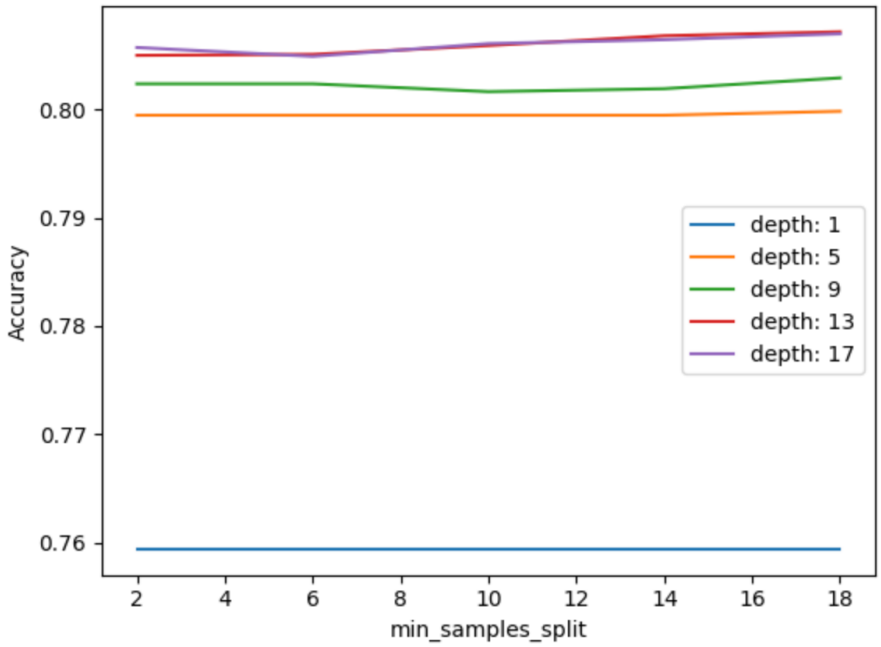

Figure 4-4. Hyperparameter Tuning for Multiclass Classification using Decision Tree

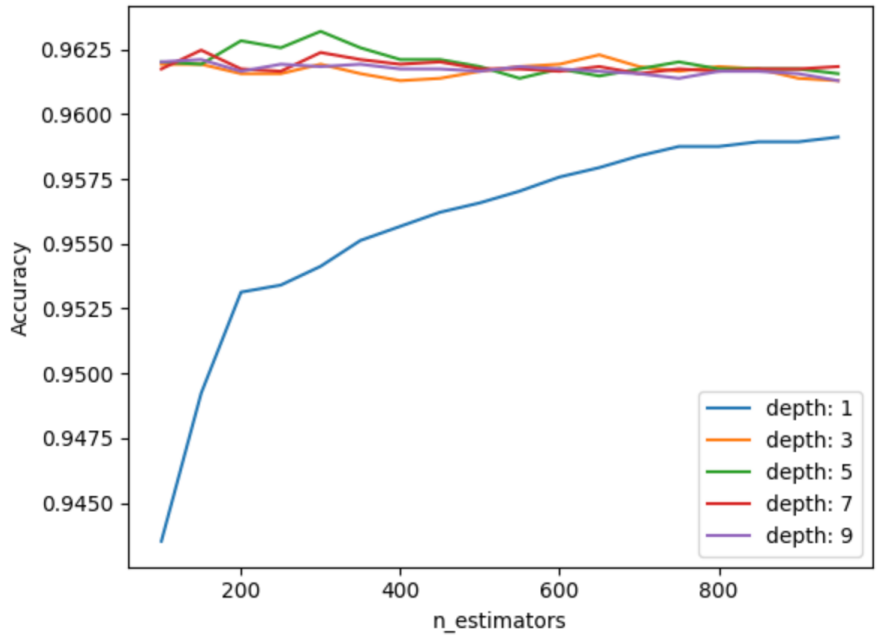

Figure 4-5. Hyperparameter Tuning for Binary Classification using XGBoost

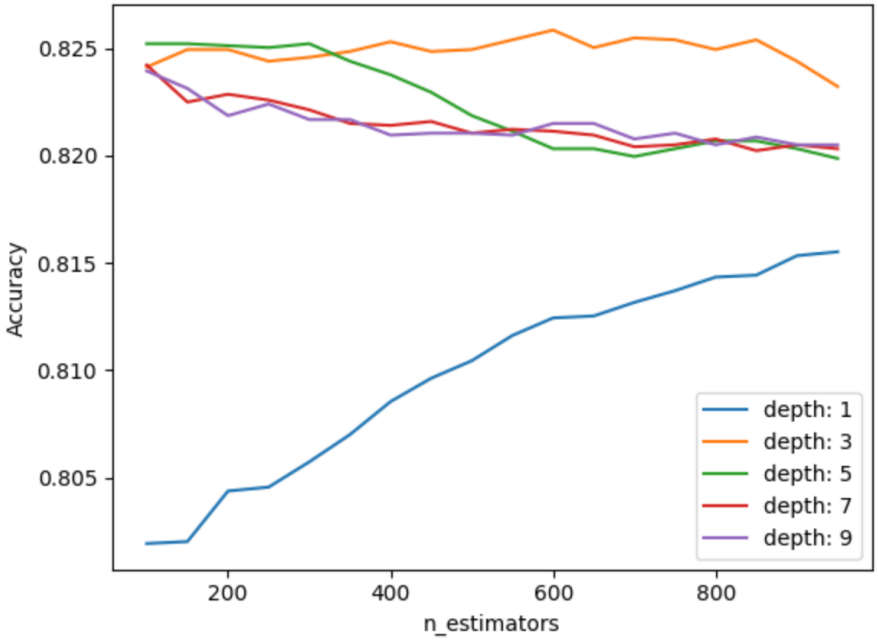

Figure 4-6. Hyperparameter Tuning for Multiclass Classification using XGBoost

4.7 Model Training & Testing

As noted throughout previous sections, the models selected for the binary prediction and multiclass categorization of scam calls are Decision Tree and eXtreme Gradient Boosting. During this final stage of the model development process, the split datasets will be used, and the evaluated optimal parameters will be configured to train and test the models.

Table 4-23. Decision Tree Binary Classification Report

(without SMOTE)

	Precision	Recall	F1-score	Support
0	95.09	98.44	96.74	2185
1	96.76	90.13	93.33	1125
Macro avg	95.92	94.29	95.03	3310
Weighted avg	95.66	95.62	95.58	3310
Accuracy	95.62			3310

Table 4-24. Decision Tree Binary Classification Report

(with SMOTE)

	Precision	Recall	F1-score	Support
0	97.00	99.45	98.21	2185
1	98.88	94.04	96.40	1125
Macro avg	97.94	96.75	97.31	3310
Weighted avg	97.64	96.61	97.60	3310
Accuracy	97.61			3310

Table 4-23 lists the classification report for the Decision Tree model used for binary prediction, which indicates the model achieved 95.62% accuracy without SMOTE sampling method. Table 4-24 showcases that with the sampling method applied, the

model achieves an accuracy of 97.61%. Further evaluation of the Decision Tree model for binary prediction is conducted through the use of a confusion matrix.

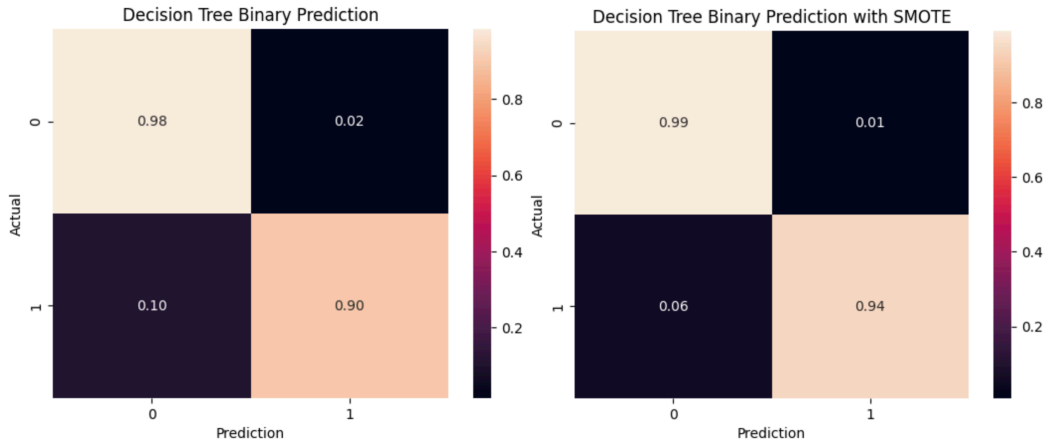

Figure 4-7. Decision Tree Binary Confusion Matrix

Figure 4-7 displays the confusion matrix before and after oversampling is applied. The results produced from both the classification report and confusion matrix, once sampling is applied, indicate that the Decision Tree Model is able to predict whether an incoming phone call is a scam call or a legitimate call 97.61% of the time.

Table 4-25. XGBoost Binary Classification Report

(without SMOTE)

	Precision	Recall	F1-score	Support
0	96.12	98.58	97.33	2259
1	96.31	91.43	94.03	1051
Macro avg	96.45	95.01	95.68	3310
Weighted avg	96.33	96.31	96.29	3310
Accuracy	96.31			3310

Table 4-26. XGBoost Binary Classification Report

(with SMOTE)

	Precision	Recall	F1-score	Support
0	97.86	99.34	98.59	2259
1	98.53	95.34	96.91	1051
Macro avg	98.19	97.34	97.75	3310
Weighted avg	98.07	98.07	98.06	3310
Accuracy	98.07			3310

Table 4-25 and Table 4-26 detail the classification report for the XGBoost classifier for binary prediction without and with SMOTE. There is about a 2% enhancement in the model accuracy after performing class balancing using SMOTE. Further evaluation of the XGBoost model for binary prediction is conducted through the use of a confusion matrix.

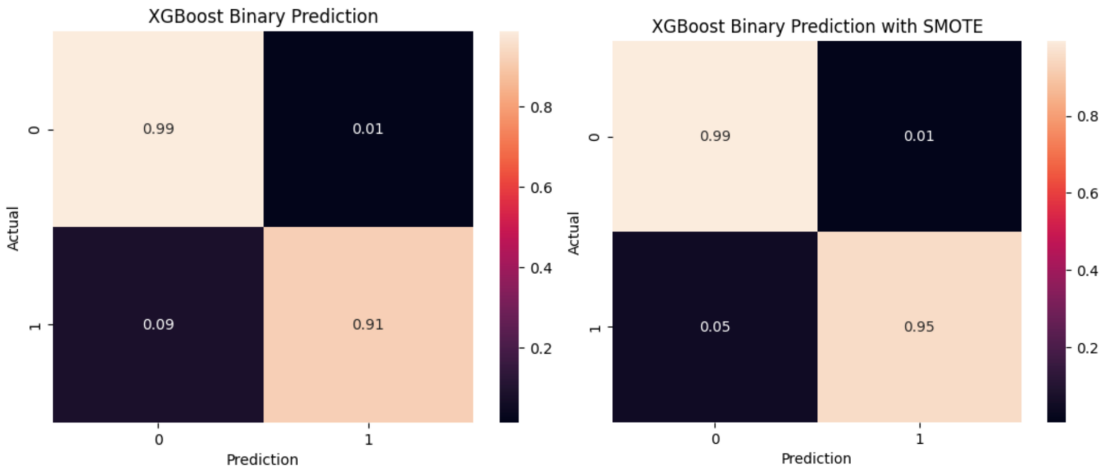

Figure 4-8. XGBoost Binary Confusion Matrix

Figure 4-8 reveals a confusion matrix for the XGBoost classifier before and after the SMOTE sampling method is applied, indicative of an accuracy of 96.31% and 98.07%, respectively. The results achieved between both the classification report and the

confusion matrix indicate that the eXtreme Gradient Boosting model can accurately distinguish between scam calls and legitimate calls 98.07% of the time.

Table 4-27 and Table 4-28 display the performance metrics for the Decision Tree model for multiclass classification. Prior to the application of the SMOTE method, the accuracy performance of the model was about 80.42%. The performance is likely lower due to the imbalanced nature of the phone call dataset. This observation is substantiated by the improvement in performance when the minority classes all have an equal distribution of data points. The Decision Tree model achieves an accuracy of 91.57% after applying the SMOTE method. In instances regarding scam call categorization, the Decision Tree model can accurately classify the incoming call about 92% of the time, providing about an 11% improvement in accuracy in comparison to the model that is not developed using the oversampling method.

Table 4-27. Decision Tree Multiclass Classification Report

(without SMOTE)

	Precision	Recall	F1-score	Support
0	94.62	99.07	96.80	2256
1	32.23	22.41	26.44	174
2	22.50	80.36	11.84	112
3	52.11	69.13	59.42	447
4	41.18	12.96	19.72	54
5	35.60	23.60	28.38	267
Macro avg	46.37	39.20	40.43	3310
Weighted avg	77.53	80.42	78.40	3310
Accuracy	80.42			3310

Table 4-28. Decision Tree Multiclass Classification Report *(with SMOTE)*

	Precision	Recall	F1-score	Support
0	97.88	98.40	98.14	2256
1	68.91	76.44	72.48	174
2	70.00	75.00	72.41	112
3	84.71	80.54	82.57	447
4	67.86	70.37	69.09	54
5	79.03	73.41	76.12	267
Macro avg	78.07	79.03	78.47	3310
Weighted avg	91.63	91.57	91.57	3310
Accuracy	91.57			3310

Figure 4-9 demonstrates that the presence of an imbalanced dataset can create difficulty for the model to accurately classify a phone call except for authentic phone calls. Once the SMOTE technique is applied, we see significant improvement. Based on the classification report and confusion matrix that evaluate the performance of the Decision Tree model as it applies to scam call categorization, this approach is a novel and viable solution to accurately classify incoming phone calls.

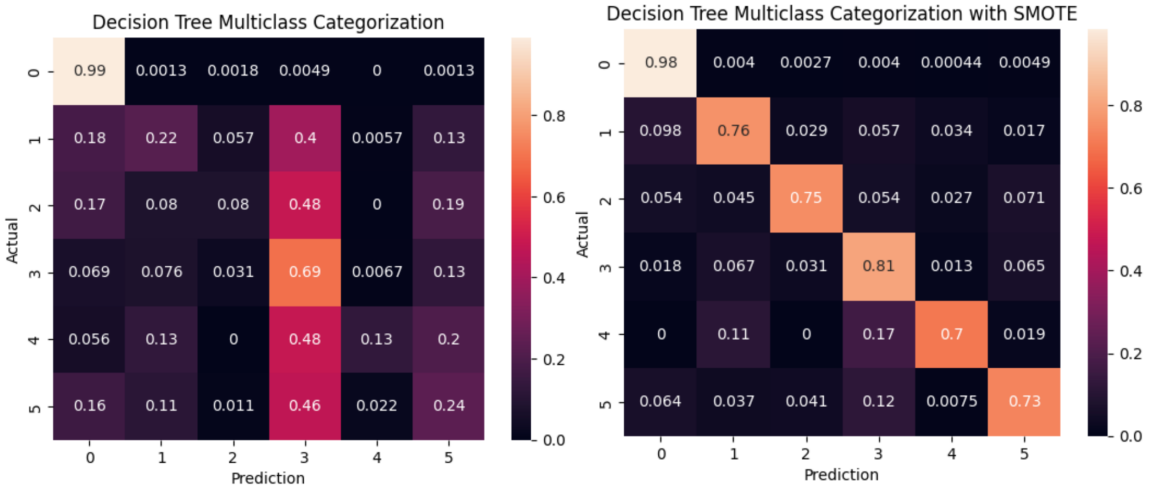

Figure 4-9. Decision Tree Multiclass Confusion Matrix

Table 4-29 and Table 4-30 showcase the evaluation of the XGBoost model as it relates to multiclass classification. Prior to oversampling, the model is able to achieve 81.96% accuracy in scam call categorization. Similar to the decision tree model, the dataset used to train and test during model development is imbalanced, which results in a lower performance in accuracy. Table 4-30 lists an accuracy of 89.34% when the SMOTE technique is applied to create a balanced dataset. There is about a 7% enhancement in the model accuracy after performing class balancing using SMOTE.

Table 4-29. XGBoost Multiclass Classification Report *(without SMOTE)*

	Precision	Recall	F1-score	Support
0	94.55	98.92	96.69	2212
1	46.43	26.26	33.555	198
2	40.30	28.13	23.13	96
3	58.78	69.64	63.75	471
4	23.33	12.07	15.91	58
5	48.47	40.36	44.05	275
Macro avg	51.98	45.90	47.85	3310
Weighted avg	79.94	81.96	80.59	3310
Accuracy	81.96			3310

Table 4-30. XGBoost Multiclass Classification Report *(with SMOTE)*

	Precision	Recall	F1-score	Support
0	96.67	98.46	97.56	2212
1	70.11	65.15	67.54	198
2	66.67	72.92	69.65	96
3	79.32	78.98	79.15	471
4	51.47	60.34	55.56	58
5	74.89	62.91	68.38	275
Macro avg	73.19	73.13	72.97	3310
Weighted avg	89.14	89.34	89.17	3310
Accuracy	89.34			3310

Figure 4-10 demonstrates how the use of the SMOTE technique can create a balanced dataset that can be used for training the XGBoost model. The improvement in performance makes this model a viable solution to accurately categorize incoming phone calls.

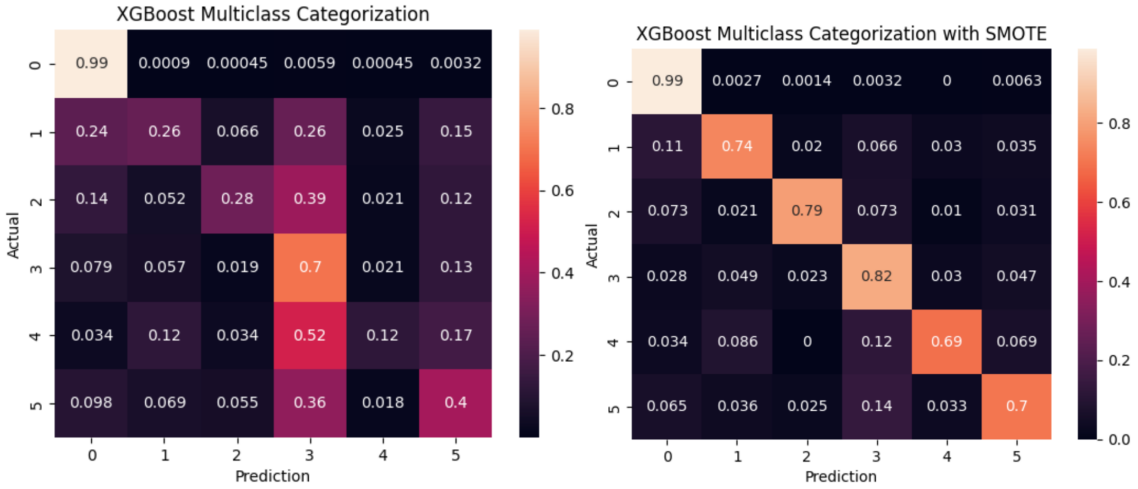

Figure 4-10. XGBoost Multiclass Classification Confusion Matrix

Among the models evaluated for scam call detection and classification, XGBoost outperforms the Decision Tree model for binary classification by just under 1%. On the other hand, the Decision Tree model outperforms the XGBoost model for multiclass classification by just over 1%. The Decision Tree model outperforming XGBoost, as it relates to the multiclass classification, is likely due to the native capabilities of each classifier and how they handle data. As highlighted by researchers Zhang et al. (2022), XGBoost models can struggle with classification performance when dealing with imbalanced data, as is the case with the original dataset constructed for model development. As previously discussed, to address this dataset imbalance, the minority classes in the training set are supplemented using the oversampling method, SMOTE. Researchers Velarde et al. (2023) have empirically proven that the use of sampling to

resolve imbalance does not always produce consistent improvements on training sets of 1,000 data points or 10,000 data points when using XGBoost. Similarly, this may be playing a factor in the performance being achieved by XGBoost for multiclass classification of scam calls as the training set consists of roughly 7700 data points. There is an improvement in the overall model's performance when sampling is introduced, but given the limited size of the training dataset, which is less than 10,000 data points, the peak performance of XGBoost is unable to be achieved, likely resulting in the Decision Tree model outperforming XGBoost. This can occur when working with smaller sample sizes; while these limitations may exist, the difference in performance between the two models is minimal.

Chapter 5—Discussion and Conclusions

5.1 Discussion

Receiving scam calls has grown into a common practice that causes interruptions to a user's day, which can have a greater impact, leading to financial fraud and identity fraud. The objective of this praxis is to develop a preventative solution using machine learning to accurately detect those scam calls and categorize them into specific classes. The research hypotheses that are presented and proven in this praxis serve as guidelines to address those goals.

5.1.1 Research Hypothesis 1

The first research hypothesis states, "Phone call metadata improves the performance of a predictive model for scam call detection." The goal of this research hypothesis (RH) is to prove that by introducing supplementary information about a given call, the ability to detect whether the call is malicious or not can be improved as compared to similar work conducted by researchers. The models trained and tested in this praxis, Decision Tree and XGBoost, leverage supplementary features beyond the call detail records highlighted in previous work and is not reliant on attributes that can only be collected if a suspected scam call is answered. As a result, the performance for binary prediction for the more performant model, XGBoost, is 98.07% accuracy in the detection of scam calls, which outperforms state of the art models that leverage similar features**.**

5.1.2 Research Hypothesis 2

The second research hypothesis states, "A scam call prevention tool accurately blocks scam calls without interruption to user experience and impact to user privacy." As highlighted in Chapter 2, which examines state-of-the-art scam call detection, novel

methods have been developed to detect a scam call using audio or audio-related features. This approach requires listening to a phone call, which can infringe upon a user's right to privacy as dictated according to various regulations. The solution presented in this praxis does not require any voicemail or audio-related data in order to accurately detect a malicious call.

5.1.3 Research Hypothesis 3

The third research hypothesis states, "The use of a scam call prevention tool accurately categorizes the type of scam call for user awareness." With the advent of tools from mobile carriers that can label a suspected phone call as *scam likely,* yearly reports indicate that these malicious calls are still prevalent and effective today. Among other considerations, familiarity with the origin of the phone number or associated area code plays a role in such scams being successful. The novel solution introduced in this praxis addresses this reality by categorizing the types of scam calls into six classes: Authentic, Medical, Utilities, Impersonator, Technical, and Financial, with an accuracy of 91.57%. The class or label that becomes associated with a given scam call provides an indication of the caller's intention, which reduces the likelihood that a scam phone call will be answered.

5.2 Conclusions

As revealed throughout this praxis, and particularly in Chapter 2, the problem of scam calls and the ability to prevent them has existed for more than 20 years. The solution presented in this work acknowledges this reality by taking a novel approach, as outlined in the research hypotheses, to address the issue. The anti-scam call solution

accurately detects the occurrence of malicious phone calls and further categorizes and classifies those calls into classes for better insight into the intention of a given caller.

5.3 Contributions to Body of Knowledge

The literature regarding existing solutions to scam calls that are examined in this praxis exposes a gap in those state-of-the-art approaches. Particularly, there are three key things that are revealed during the literature review: the use of blacklisting and assigning trust is insufficient, the use of audio data such as phone calls and voicemails can violate someone's right to privacy, and the use of audio features, including call duration, require a scam call to successfully go through to be detected. The solution introduced in this praxis addresses each of these gaps.

The practice of blacklisting and whitelisting or assigning trust is predicated on the maintenance of a list or table comprised of known good actors or known bad actors. This process is unsustainable and incapable of easily scaling as technology has evolved, and the ability to generate new phone numbers has become easier. The methodology employed as part of the solution in this praxis leverages predictive modeling to decide whether a phone call is malicious or authentic using phone call metadata.

The use of voicemails or active phone calls to detect whether a call is malicious or authentic introduces the infringement of privacy. While work conducted in this area has showcased this method's effectiveness, the privacy issues have not been addressed. The solution discussed in this praxis prioritizes the privacy of users and does not require or leverage any audio to detect or categorize a malicious phone call.

Audio features are characteristics of a phone call that has occurred. Such features include call duration, the presence of a voicemail, and more. In the related work, these

features are used in the creation of the dataset during model development. In order for these attributes to be collected, they actually require a suspected malicious phone call to be answered. In such a case, where a phone call is answered, unknowing if it is malicious or authentic, the likelihood of a user falling victim to a scam is significantly increased. The solution presented in the praxis is developed using phone call metadata that can be collected *without* the call having to be answered by the user, and it outperforms existing solutions by an average of about 6% in detection accuracy. This mechanism can determine whether a call is malicious or authentic and offer six different classifications of the type of call, all within seconds of the phone call occurring.

5.4 Recommendations for Future Research

The study of scam call prevention has evolved as technology has advanced. As such, areas for future work relating to the solution presented include a performance review of more modern machine learning models, collecting phone call metadata and features from a longitudinal study, and conducting a review of demographic correlation to understand the targets of scam calls.

The models utilized in this research include Decision Tree and eXtreme Gradient Boosting for binary prediction and multi-class categorization. Future research can investigate how newer methods such as deep learning models can specifically improve on the performance of the classification of scam calls. Considering that each model required supplementing minority classes via Synthetic Minority Over-sampling Technique (SMOTE), deep learning models can be leveraged as they operate well with sparse features and may not require the process of oversampling.

The data collected for this study spans over about a month and a half between October and December of 2023. A longitudinal study would allow phone call metadata to be collected over a longer period which could help in the generation of a more balanced dataset across all the classes during the initial model development. This can also remove the need to oversample the minority classes via the SMOTE method.

The dataset developed for this solution offers information about the cities and states impacted by scam calls during the months of October 2023 and December 2023. This information can be used to analyze the demographic composition of those locations to identify patterns or any correlation of the frequency of malicious calls to specific communities.

References

Adam, W. A. B., Tan, I. Y. X., Lai, C. S., Rahim, N. T., Bin Tham, B. Y., and Guo, H., "VishingDefender: An Advanced Vishing Defence System Against Vishing Attacks," 2023 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), Singapore, 2023, pp. 1-8, doi: 10.1109/SOLI60636.2023.10425272.

Akinyemi, B.O., Odukoya, O. H., Sanni, M.L., Sewagnon, G., Aderounmu, G. A., "Performance Evaluation of Machine Learning-Based Robocalls Detection Models in Telephony Networks", International Journal of Computer Network and Information Security (IJCNIS), Vol.14, No.6, pp.37-53, 2022. DOI:10.5815/ijcnis.2022.06.04

Alsarhan, H. "Real-Time Machine Learning-Based Intrusion Detection System (IDS) for Internet of Things (IoT) Networks." George Washington University, 2023.

Anyanwu, G.Oluchi, Nwakanma, C. I., Lee, J. -M., and Kim, D. -S. "Optimization of RBF SVM Kernel Using Grid Search Algorithm for DDoS Attack Detection in SDN-Based VANET," in IEEE Internet of Things Journal, vol. 10, no. 10, pp. 8477-8490, 15 May15, 2023, doi: 10.1109/JIOT.2022.3199712.

Azam, Z., Islam, Md. M., & Huda, M. N. (2023). Comparative Analysis of Intrusion Detection Systems and Machine Learning-Based Model Analysis Through Decision Tree. IEEE Access, 11, 80348–80391. https://doi.org/10.1109/ACCESS.2023.3296444

Benjamin, E. and Emmanouilidou, D. "Detection of Robocall and Spam Calls Using

Acoustic Features of Incoming Voicemails.” Proceedings of Meetings on Acoustics, vol. 45, no. 1, 2021, https://doi.org/10.1121/2.0001533.

Chaisamran, N., Okuda, T., Blanc, G., and Yamaguchi,S., "Trust-Based VoIP Spam Detection Based on Call Duration and Human Relationships," 2011 IEEE/IPSJ International Symposium on Applications and the Internet, Munich, Germany, 2011, pp. 451-456, doi: 10.1109/SAINT.2011.84.

Chu, G. et al., "Exploiting Spatial-Temporal Behavior Patterns for Fraud Detection in Telecom Networks," in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 6, pp. 4564-4577, Nov.-Dec. 2023, doi: 10.1109/TDSC.2022.3228797.

Cialdini, R. B. (2007). Influence: the psychology of persuasion. Rev. ed.; 1st Collins business essentials ed. New York, Collins

Clare, D.D., & Levine, T.R. (2019). Documenting the truth-default: the low frequency of spontaneous unprompted veracity assessments in deception detection. Human Communication Research, 45, 286-308. https://doi,org/10.1093/hcr/hqz001

Dash, P. B., et al. “Model Based IoT Security Framework Using Multiclass Adaptive Boosting with SMOTE.” Security and Privacy, vol. 3, no. 5, 2020, https://doi.org/10.1002/spy2.112.

Dhaliwal S.S., Nahid, A-A., Abbas R. Effective Intrusion Detection System Using XGBoost. Information. 2018; 9(7):149. https://doi.org/10.3390/info9070149

Dietrich, C. and Hesse, M. (2007). Evaluation of the Possible Utilization of anti-spam Mechanisms Against spit. In: ISSE/SECURE 2007 Securing Electronic Business Processes. Vieweg. https://doi.org/10.1007/978-3-8348-9418-2_21

Dissanayake, D., Gamage, P., Kumarasinghe, N. , and Ganegoda, G. U., "Leveraging Artifact Reputation Analysis and Contextual Sentiment Analysis for Advanced Detection of Vishing and Smishing Attacks," 2023 8th International Conference on Information Technology Research (ICITR), Colombo, Sri Lanka, 2023, pp. 1-5, doi: 10.1109/ICITR61062.2023.10382862.

Doherty, J., and Anderson, N. Internet Phone Services Simplified: An Illustrated Guide to Understanding, Selecting, and Implementing VolP-Based Internet Phone Services for Your Home. 1st edition, Cisco Press, 2006.

Edwards, Gregory W., Gonzales, Michael J., and Sullivan, Marc A. 2020. Robocalling: STIRRED AND SHAKEN! - An Investigation of Calling Displays on Trust and Answer Rates. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20). Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376679

Fischer, P., Lea, S. E. G., & Evans, K. M. (2013). Why do individuals respond to fraudulent scam communications and lose money? The psychological determinants of scam compliance. Journal of Applied Social Psychology, 43(10), 2060–2072. https://doi.org/10.1111/jasp.12158

Gamage, P., Dissanayake, D., Kumarasinghe, N., and Ganegoda, G. U. "Acoustic Signature Analysis for Distinguishing Human vs. Synthetic Voices in Vishing Attacks," 2023 8th International Conference on Information Technology Research (ICITR), Colombo, Sri Lanka, 2023, pp. 1-6, doi: 10.1109/ICITR61062.2023.10382846.

Gowri, S. M., Ramana, G. S., Ranjani, M. S., and Tharani, T. "Detection of Telephony

Spam and Scams using Recurrent Neural Network (RNN) Algorithm," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 1284-1288, doi: 10.1109/ICACCS51430.2021.9441982.

Hadnagy, C. (2018). Social Engineering: The Art of Human Hacking (Ed.2). Indianapolis: John Wiley and Sons.

Heras, Gema de las, et al. “Robocall Scam Examples.” Consumer Advice, 12 July 2023, consumer.ftc.gov/features/robocall-scam-examples.

Hoffstadt, D., Rathgeb, E., Liebig, M., Meister, R., Rebahi, Y., and Thanh, T.R.,"A comprehensive framework for detecting and preventing VoIP fraud and misuse," 2014 International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, USA, 2014, pp. 807-813, doi: 10.1109/ICCNC.2014.6785441.

Hong, B., Connie, T., and Goh, M. (2023). Scam Calls Detection Using Machine Learning Approaches. 442-447. 10.1109/ICoICT58202.2023.10262695.

Jones, K.S., Armstrong, M.E., Tornblad, M.K. and Siami Namin, A. (2021), "How social engineers use persuasion principles during vishing attacks", Information and Computer Security, Vol. 29 No. 2, pp. 314-331. https://doi.org/10.1108/ICS-07-2020-0113

Khonji, M., Iraqi, Y., and Jones, A., "Phishing Detection: A Literature Survey," in IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091-2121, Fourth Quarter 2013, doi: 10.1109/SURV.2013.032213.00009.

Koilada, D. V. S. R. K., "Strategic Spam Call Control and Fraud Management:

Transforming Global Communications," in IEEE Engineering Management Review, vol. 47, no. 3, pp. 65-71, 1 thirdquarter,Sept. 2019, doi: 10.1109/EMR.2019.2924635.

Kumar, H., and Sastry, P. S., "Robust Loss Functions for Learning Multi-class Classifiers," 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 2018, pp. 687-692, doi: 10.1109/SMC.2018.00125.

Li, H., et al., "A Machine Learning Approach to Prevent Malicious Calls over Telephony Networks," 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2018, pp. 53-69, doi: 10.1109/SP.2018.00034.

Lieto, A., et al., ""Hello? Who Am I Talking to?" A Shallow CNN Approach for Human vs. Bot Speech Classification," ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 2019, pp. 2577-2581, doi: 10.1109/ICASSP.2019.8682743.

Malhotra, S., Arora, G., and Bathla, R. "Detection and Analysis of Fraud Phone Calls using Artificial Intelligence," 2023 International Conference on Recent Advances in Electrical, Electronics & Digital Healthcare Technologies (REEDCON), New Delhi, India, 2023, pp. 592-595, doi: 10.1109/REEDCON57544.2023.10150631.

Munyaka, Imani NS., Delgado, Daniel A., Gilbert, Juan E., Ruiz, Jaime, and Traynor, Patrick. (2024). “I used to live in Florida”: Exploring the Impact of Spam Call Warning Accuracy on Callee Decision-Making. In Proceedings of Symposium on Usable Security and Privacy (USEC) 2024

Pandit, S., Liu, J., Perdisci, R., and Ahamad, M., "Applying Deep Learning to Combat

Mass Robocalls," 2021 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 2021, pp. 63-70, doi: 10.1109/SPW53761.2021.00018.

Patankar, P., Nam, G., Kesidis, G., and Das, C. R. "Exploring Anti-Spam Models in Large Scale VoIP Systems," 2008 The 28th International Conference on Distributed Computing Systems, Beijing, China, 2008, pp. 85-92, doi: 10.1109/ICDCS.2008.71.

Peltier, T. R. (2006). Social Engineering: Concepts and Solutions. Information Systems Security, 15(5), 13–21. https://doi.org/10.1201/1086.1065898X/46353.15.4.20060901/95427.3

Pohlmann, N**.** & Reimer, H**.** & Schneider, W. (2007). **"Mechanisms Against SPIT,"** ISSE/SECURE 2007 Securing Electronic Business Processes. 10.1007/978-3-8348-9418-2.

Prasad, S., Bouma-Sims, E., Mylappan, A.K., & Reaves, B. (2020). Who's Calling? Characterizing Robocalls through Audio and Metadata Analysis. USENIX Security Symposium.

Quittek, J., Niccolini, S., Tartarelli, S., and Schlegel, R. , "On Spam over Internet Telephony (SPIT) Prevention," in IEEE Communications Magazine, vol. 46, no. 8, pp. 80-86, August 2008, doi: 10.1109/MCOM.2008.4597108.

Rimal, Y., Sharma, N., & Alsadoon, A. (2024). The accuracy of machine learning models relies on hyperparameter tuning: student result classification using random forest, randomized search, grid search, bayesian, genetic, and optuna algorithms. Multimedia Tools and Applications. https://doi.org/10.1007/s11042-024-18426-2

Ruan, N., Wei, Z., and Liu, J., "Cooperative Fraud Detection Model With Privacy

Preserving in Real CDR Datasets," in IEEE Access, vol. 7, pp. 115261-115272, 2019, doi: 10.1109/ACCESS.2019.2935759.

Sanchez, O. R., Repetto, M., Carrega, A., and Bolla, R., "Evaluating ML-based DDoS Detection with Grid Search Hyperparameter Optimization," 2021 IEEE 7th International Conference on Network Softwarization (NetSoft), Tokyo, Japan, 2021, pp. 402-408, doi: 10.1109/NetSoft51509.2021.9492633.

Sandhya, S., Karthikeyan, N., Sruthi, R., "Machine learning method for detecting and analysis of fraud phone calls datasets" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

Sipper, M. (2022). High Per Parameter: A Large-Scale Study of Hyperparameter Tuning for Machine Learning Algorithms. Algorithms, 15(9), 315-. https://doi.org/10.3390/a15090315

Song, J., Kim, H. and Gkelias, A. (2014), iVisher: Real-Time Detection of Caller ID Spoofing. ETRI Journal, 36: 865-875. https://doi.org/10.4218/etrij.14.0113.0798

Stajano, F. and Wilson, P. (2011), "Understanding scam victims: seven principles for systems security", Communications of the ACM, Vol. 54 No. 3, pp. 70-75, doi: 10.1145/1897852.1897872.

Tu, H., Doupé, A., Zhao, Z., Ahn, G. (2016). SoK: Everyone Hates Robocalls: A Survey of Techniques Against Telephone Spam.320-338. 10.1109/SP.2016.27.

United States Congress, Consumer and Governmental Affairs. (2023). "Scam Glossary." Federal Communications Commission.www.fcc.gov/scam-glossary.

United States Congress, Consumer and Governmental Affairs. TRACED Act

Implementation, Federal Communications Commission, May 2023. https://www.fcc.gov/TRACEDAct.

United States Congress, Telephone Robocall Abuse Criminal Enforcement and Deterrence Act. Telephone Robocall Abuse Criminal Enforcement and Deterrence Act, U.S. Government Publishing Office, 2019. 2019 Congress, law.

Velarde, G., Sudhir, A., Deshmane, S., Deshmunkh, A., Sharma, K., & Joshi, V. (2023). Evaluating XGBoost for balanced and imbalanced data: application to fraud detection. arXiv preprint arXiv:2303.15218.

Wang, S., Delavar, M., Azad, M. A., Nabizadeh, F., Smith, S.,and Hao, Feng. (2023). Spoofing Against Spoofing: Toward Caller ID Verification in Heterogeneous Telecommunication Systems. ACM Trans. Priv. Secur. 27, 1, Article 1 (February 2024), 25 pages. https://doi.org/10.1145/3625546

Xing, J., Yu, M., Wang, S., Zhang,Y., Ding, Y., "Automated Fraudulent Phone Call Recognition through Deep Learning", Wireless Communications and Mobile Computing, vol. 2020, Article ID 8853468, 9 pages, 2020. https://doi.org/10.1155/2020/8853468

Zhang P, Jia Y, Shang Y. Research and application of XGBoost in imbalanced data. International Journal of Distributed Sensor Networks. 2022;18(6). doi:10.1177/15501329221106935

Zhen, Z. and Gao, J."Cdr2img: a bridge from text to image in telecommunication fraud detection," Computer Systems Science and Engineering, vol. 47, no.1, pp. 955–973, 2023.

Appendix A

Table A-1. Dataset Feature List

Features	Definition/Relevance
Company_Phone_Number	Number used by threat actor to execute scam calls
Consumer_State	Target destination or state where scam call was received
Consumer_City	Target destination or city where scam call was received
Subject	Core objective of threat actor's actions ***(Ex:*** *Utility scam call, Medical scam call)*
Recorded_Message_or_Robocall	Indicates whether a recorded voicemail message was left for the receiver or if robocall present when call answered
Created_Date	When the complaint was made with FTC
Violation_Date	When scam the scam call occurred
Supplementary.carrier_name	Name of the telecommunication company or mobile carrier
Supplementary.mobile_country_code	Country code for mobile carrier
Supplementary.mobile_network_code	Network code for mobile carrier
Supplementary.type	Type of phone line used to execute the scam call

	(***Ex:*** *landline, mobile*)
ULS_Active	Indicator for valid registered ULS license with the FCC
Foreign_Ownership	Is the company owned or partially owned by a foreign entity?
Licensee Type	What type of organization registered this license? (***Ex:*** *Corporation, LLC)*
Radio Service	Indicates which communication medium or radio service is in use
Regulatory Status	Indicates which regulations the organization needs to adhere to
Label	Ground truth to indicate if a call entry is malicious or authentic

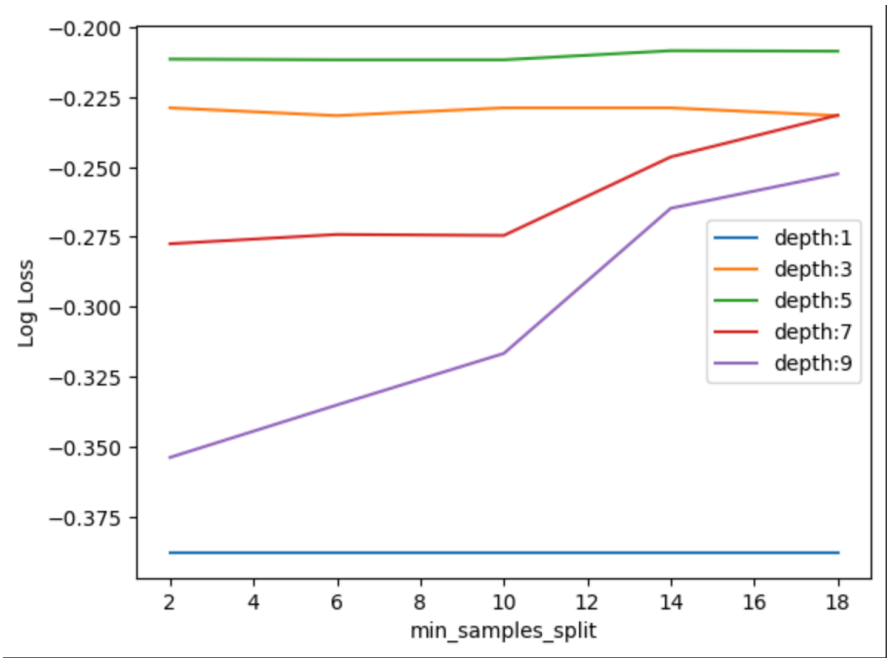

Figure A-1. Hyperparameter Tuning for Binary Classification using Decision Tree

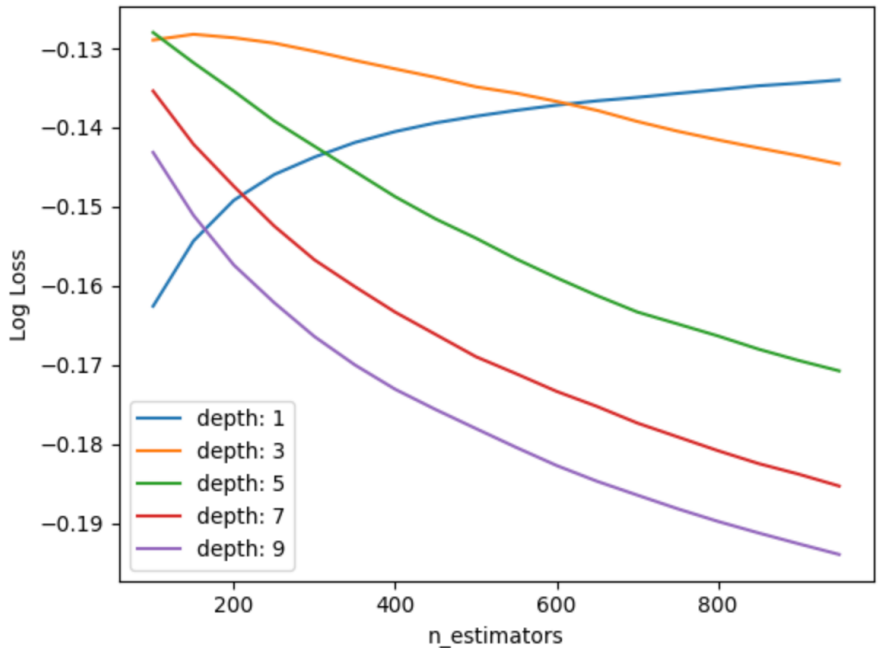

Figure A-2. Hyperparameter Tuning for Binary Classification using XGBoost

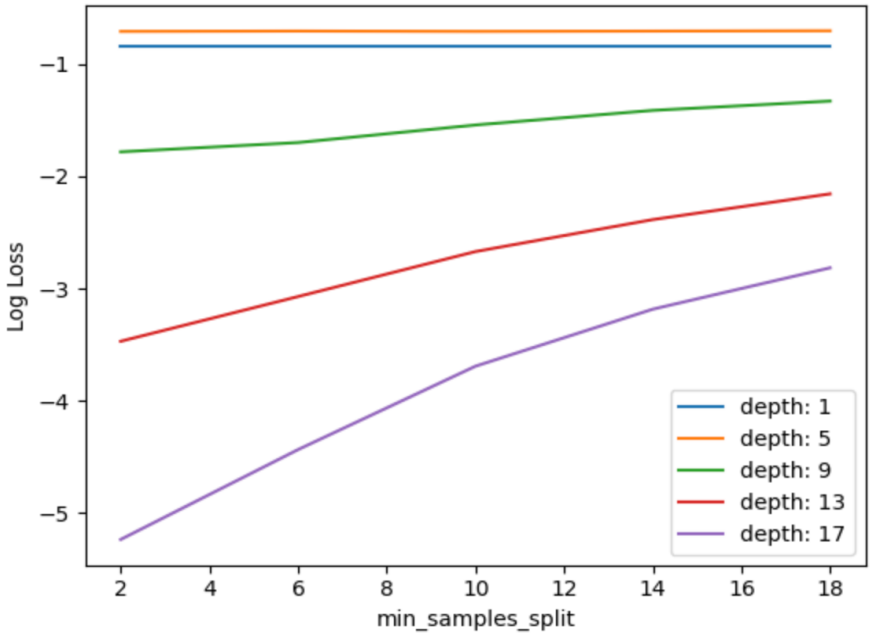

Figure A-3. Hyperparameter Tuning for Multiclass Classification using Decision Tree

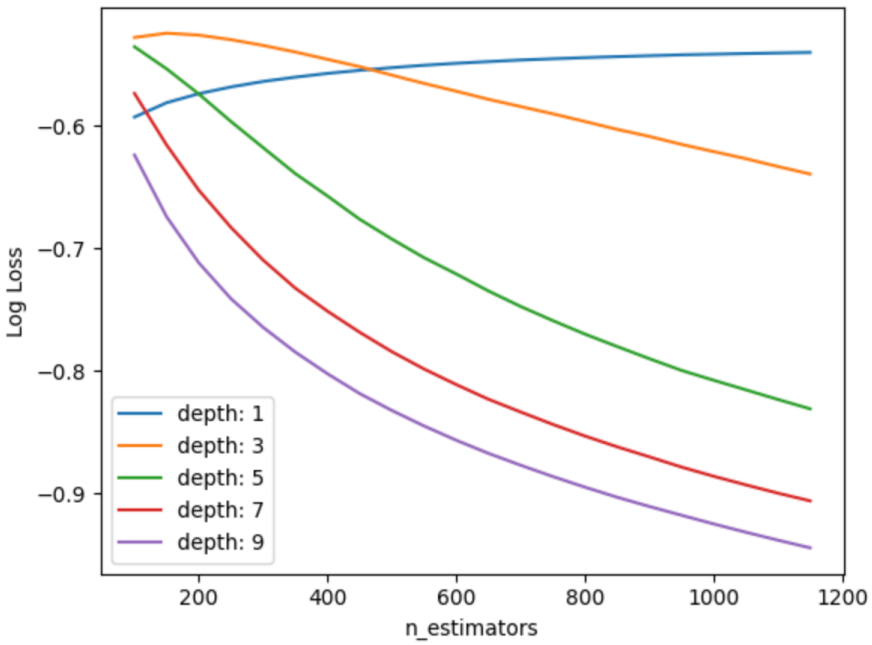

Figure A-4. Hyperparameter Tuning for Multiclass Classification using XGBoost