ENSEMBLE LEARNING FOR URL PHISHING DETECTION

Igwilo, Chiamaka Mary

(40806)

A Thesis submitted to the Department of Computer Science at the African University of Science and Technology

In Partial Fulfilment of the Requirements for the degree of Master of Science in the Computer Science Department

July 2021

CERTIFICATION

This is to certify that the thesis titled "Ensemble Learning for URL Phishing Detection" submitted to the School of Postgraduate Studies, African University of Science and Technology (AUST), Abuja, Nigeria for the award of the Master's degree is a record of original research carried out by Igwilo, Chiamaka in the Department of Computer Science.

African University of Science and Technology [AUST]

Knowledge is Freedom

<u>APPROVAL BY</u>

Supervisor

Surname: Odumuyiwa

First name: Victor

Signature

The Head of Department

Surname: Prasad

First name: Rajesh

Signature

COPYRIGHT

ABSTRACT

Phishing is a social engineering attack that has been perpetuated for long and is still a prominent attack with an attending high number of victims. The adverse effect of this allows phishers easy access to sensitive information about a company or an individual. This research compares the import of features such as lexical features, Domain Name Based features, HTML Features, and tokenization of URLs in detecting phishing URLs. Experimental procedures were designed to compare the efficiency of the four different approaches used separately on three machine learning models and five ensemble learning classifiers. The classification of URLs is done using K-Nearest Neigbour, Decision Tree, Logistic Regression, Random Forest, Bagging, Stacking, Ada Boost, Gradient Boost. The research shows that using URL tokenization performs better for both machine learning and ensemble learning classifiers.

Keywords: Ensemble Learning. Phishing detection, machine learning, URL features, classification

DEDICATION

This work is dedicated to God, to my parents, Mr and Mrs Chris Igwilo, to my siblings, and to myself.

ACKNOWLEDGEMENT

I acknowledge God for life and this opportunity. I will like to appreciate my supervisor, Dr. Victor Odumuyiwa for guidance and mentorship during this project. I thank the staff and faculties of Computer science who have taught and mentored me during this program. I also acknowledge the Institution for their full sponsorship throughout my master's program. I want to thank Dr. Rajesh Prasad for his teaching assistance and supervision during my program. In addition, I will like to appreciate my family, Mr and Mrs Chris Igwilo, Chisom, Henry, Collins and Mmesoma for being the best support system. I will also like to thank my friends.

TABLE OF CONTENTS

List of Tables

List of Figures

CHAPTER ONE

INTRODUCTION

With the advancement of technology over the years and the tremendous growth in data through the aid of social networks, internet activities, IoT devices; the need for data privacy, protection, and security against cyber-attacks cannot be over-emphasized. While attackers keep developing new ways to gain unauthorized access to networks, programs, and data, phishing remains one of the oldest methods used.

Due to the COVID-19 pandemic, a large amount of workload and business-related projects are being carried out over the internet from home. Cybercriminals are upgrading tactics and exploring the poor technological challenges faced in securing data while working away from the offices. Working from home has become an avenue for increasing data theft, fraudulent emails, spam, and phishing attempts. The pandemic forced businesses, companies, and employees to work remotely from the office. Cybercriminals are capitalizing on the pandemic which is leading to a significant increase in the number of cyber-attacks. With the era of the Internet of Things (IoT), the number of devices connected to the internet is still growing, so are the dangers of cyber-attacks.

According to reports and the article published by Kuala Lumpur in Deloitte, it states that "***91% of all cyber-attacks begin with a phishing email to an unexpected victim and 32% of all successful breaches involve the use of phishing techniques*** " (Kuala Lumpur, 2020). Despite the several phishing attacks over the years, individuals are still falling victims to this oldest form of cyberattack.

Phishing is a form of fraud whereby an attacker tries to access sensitive information such as account and login details by sending an email to a person disguising the source of the email as though it is from a reliable organization. Usually, a victim of phishing isn't aware that the email sent contains malicious software or would redirect them to fraudulent websites tricking them into divulging personal or financial information such as credit card details, account IDs. In phishing, the attackers trick people into clicking a malicious link that would appear legitimate. Jang-Jaccard and Nepal(2014) explain how attackers are adopting increasingly sophisticated tools to phish and the need to address cybersecurity challenges. Over the years experiments carried out with machine learning (ML) show that ML techniques can effectively serve as anti-phishing tools. (Abdelhamid et al., 2017). A common technique in phishing attacks is where they contact a user offering genuine support to help them in resolving web-based issues or resolve bank-related issues thereby gaining access to steal bank security codes, personal details, and much more. Past literature discusses the use of classification algorithms like Artificial Neural Network (ANN), K-Nearest Neighbors (KNN), Decision Tree as a strategy to mitigate phishing attacks. Alkhalil et al., (2021) discusses the five various phases carried out in the lifecycle of a phishing attack. His report reviews the anatomy of these attack phases, what characteristics are connected with phishing victims, threats and vulnerabilities, and novel phishing strategies. The author buttresses the importance of phishing awareness and finding an improved anti-phishing system.

Pre-existing measures include the Internet Service Provider (ISP) tasked with shutting down malicious websites (Hutchings et al., 2016) and the use of warning tools embedded in browsers to indicate malicious sites once they are being accessed by the user.

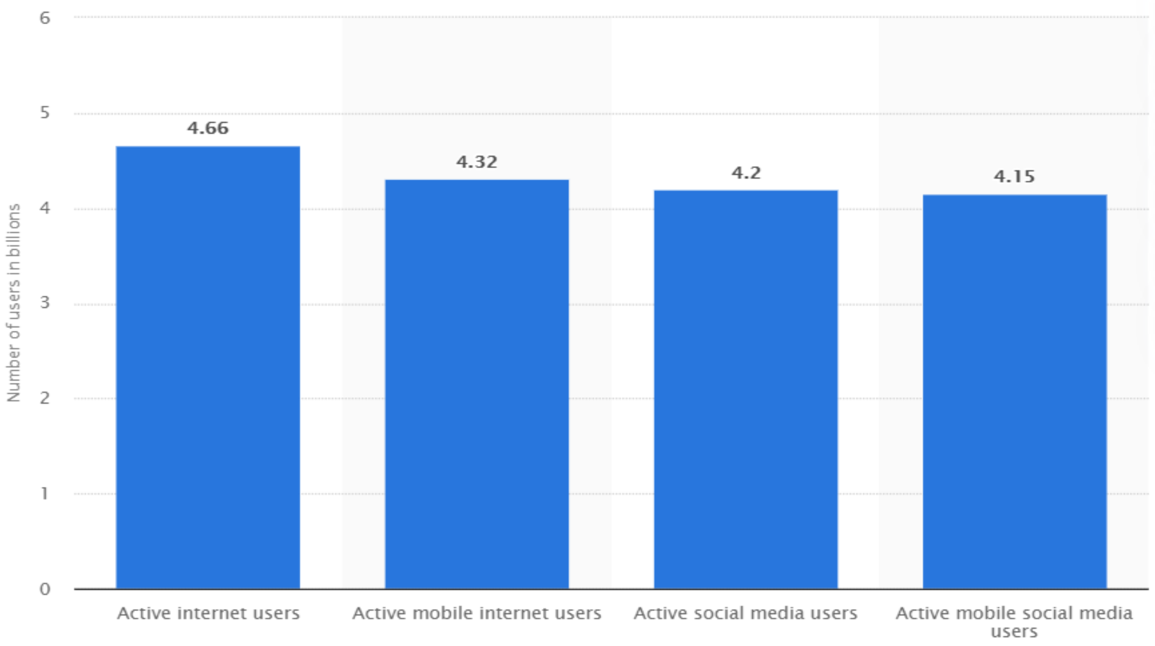


Figure 1.1: Number of internet users as of January 2021 (in billions)

The rapid growth of users accessing the internet shows the need to combat cyber threats to usher in better cyberspace. Figure 1.1 shows the number of internet users as of January, 2021(Johnson, 2021). The evolution of phishing attacks has created techniques that prey on the vulnerability of both the computer systems and users. Therefore researchers need to develop proactive measures to tackle this menace (Lim et al., 2020). An ensemble learning system uses multiple models combined to improve accuracy and reduce bias.

1.1 Ensemble learning systems

Ensemble learning is a method of combining individual models to improve the model's stability and productivity. This approach permits higher productive performance, lowers bias and reduces overfitting. It combines multiple machine learning models into one productive model to improve production using stacking and other learners (Brown, 2011). The main aim of this work is to detect phishing URLs and compare the rate of accuracy of an ensemble system with conventional machine learning classification algorithms. Algorithms like Stacking classifier, Bagging Classifier, Random forest, Adaboost, and Gradient boosting were used in the ensemble learning and Decision tree, KNearestNeigbors, Logistic Regression were used for the traditional machine learning process. The performance of an ensemble system depends on Robustness and Accuracy, model averaging where each ensemble learner contributes an equal amount to the prediction of the entire model (Jason Browniee, 2021). Ensemble models are used to lower the error that occurs in using an individual classification model thereby reducing overfitting.

The comparison is performed using various metrics of classification such as precision, rate of accuracy, recall, and F1 score (Steinki & Mohammad, 2015). (Xu Ying, 2014) describes ensemble learning as a constituent learning algorithm that uses the combination of different learning algorithms to obtain better performance and predictive inference.

Ensemble learning can be split into two methods:

1.1.1 Sequential ensemble methods

The base learners are generated serially. The basic motivation is to use the dependence between the basic learners. The performance of this model is increased by assigning higher weights to the previous learners. Example: Adaptive Boosting (AdaBoost).

1.1.2 Parallel ensemble method

When the basic learners are generated in parallel, the parallel technique is used. E.g.: Random forest. The basic motivation is to use independence between the learners and significantly reduces the error by averaging the error of the application.

1.2 Problem Statement

Phishing attacks have become more advanced and sophisticated over the years. We seek to solve these issues by providing strategies associated with using the ensemble learning model by mitigating and eliminating the occurrence of phishing attacks and comparing the accuracy of using different classification algorithms. This research addresses the following questions:

Which feature or classification model gives better prediction and higher accuracy score? In what ways can phishing attacks be detected before a person or organization is affected?

1.3 Aim and Objectives

This research aim is to develop a system using ensemble learning that detects phishing URLs. As a developing country, Nigeria still battles with numerous phishing attacks. We seek to find out ways to mitigate these attacks and increase accuracy score of detecting phishing URLs using ensemble models. Specific objectives include:

- Reviewing different phishing techniques used by attackers.
- Identifying relevant features of a phishing URL.
- Detecting benign and malicious URLs.
- Comparing accuracy score of different classifiers.

1.4 Scope

The scope of this work includes:

- Building classification models to detect malicious and benign URLs.
- Comparing classification results using feature extraction against the results of URL tokenization.
- Comparing accuracy between ensemble learners and other traditional classification methods.

1.5 Arrangement of Dissertation

Chapter One covers the introduction, background study, problem statement of the project, and the objectives of the project. Chapter Two discusses the literature review of related researches on machine learning on URL detection. Chapter Three covers the research methodology. This chapter demonstrates the methods used in implementing the research. Chapter Four discusses our experimentation procedures and the results obtained. Finally, Chapter Five provides concluding remarks and summary of work done in the thesis. This chapter also highlights some recommendations, contributions, and future work.

CHAPTER TWO

LITERATURE REVIEW

This section presents a review on phishing trends and types of phishing techniques. A focus on some traditional machine learning algorithms like Logistic Regression (LR), K-Nearest Neighbor (KNN), Decision Tree (DT) and ensemble learning methods like stacking, bagging and boosting for classification problems. This chapter will discuss past works from different researchers.

2.1 Phishing Trends

In Nigeria, the issue of phishing and cybercrimes are still ravaging the economic sector of the country. Microsoft reports that phishing attacks have increased by 250 percent in Nigeria and globally (Adepetun, 2019).

In the report published by (Ogbonnaya, 2020), it shows that in 2018, Nigeria's commercial banks lost a total of $39 million to cybercrimes and electronic fraud. According to Ogbonnaya's study, the majority of these crimes were done through phishing and identity theft, contributing to an increase over the previous year's loss of 2.37 billion to the same offences. While Deloitte Nigeria indicates that cybercriminals are taking advantage of the pandemic and tricking individuals into downloading ransomware disguised as legitimate COVID applications. According to Serianu, the Nigeria Cybersecurity report, cyber-attacks including phishing, SQL injection attacks cost Nigerian businesses about $649 million yearly (Serianu, 2017). Over the years, cyber-attacks in Nigeria have evolved. In other to adapt to the new techniques developed, some criminals are creating phishing websites to defraud victims while others are using malware to have access to private data. In respect of several warnings of phishing attacks and awareness campaigns, cyber attackers have developed other new techniques to collect personal details.

In phishing, the attackers do not target a particular victim whereas, in spear-phishing, a specific individual with a specific goal in an organization is targeted. In the past years, organizations have shown significant efforts in controlling cyber-crimes although there's a need for more effort. With the significant increase in IT devices, one could ask the question if organizations are prepaid to handle breaches and phishing attacks? In (FireEye, 2020) report, it states that 51% of organizations don't believe they are ready to handle cyber or phishing attacks. The number of phishing sites recognized by Anti-Phishing Working Group (APWG) in the 3rd quarter of 2019 was 266,387 (Anti-Phishing Working Group, 2020). This was up 46 percent from the 182,465 seen in Q2, and 138,328 seen in Q4, 2018 respectively. Figure 2.1 represents the APWG report on phishing activities in the 3rd quarter of 2019.

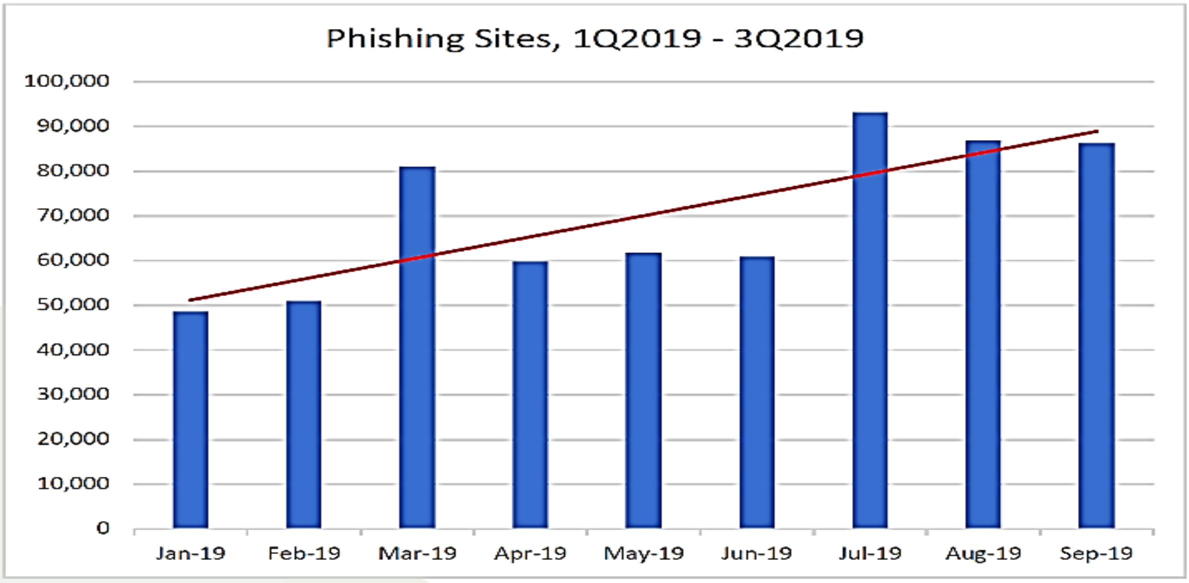


Figure 2.1: Phishing Activity Trends Report, 3rd Quarter 2019 (APWG report)

The total number of phishing websites detected in the second quarter of 2020 during the pandemic was 165,772. That was up from the 162,155 in Q4 2019. Phishing report from APWG, shows 132,553 phishing attacks occurred in the Q4 2020 and 122,359 in 3Q 2020, and 112,163 in 2Q 2020. These reports have demonstrated that the current anti-phishing techniques are not giving the desired results and are not as effective as they seem.

In the 3rd quarter of 2020, the phishing attacks trend shows that 31.4% of all attacks are SAAS and webmail sites attacks against social media companies. There was a rise from 10.8 to 12.6 percent in phishing attacks against social media companies. Scammers demanded payments in the form of gift cards in 71 percent of business email compromise (BEC) assaults in the third quarter, according to the research (Anti-Phishing Working Group, 2020). In 6% of attacks, the attackers requested payroll diversions, there was a significant decrease of 25% from the value gotten in the third quarter of 2019.

The APWG 2020 report states that during wire transfer attacks the average was $48,000 in Q3, a significant decrease from $80,000 in Q2 and $54,000 in Q1. Agari is an organization against phishing, BEC scams, and other advanced email threats. In Agari’s report, it identifies BEC countries where scammers are located and list Nigeria as a traditional epicenter of social engineering scam amongst 50 countries (APWG, 2020). F5 labs report about the evolution of attackers in the past years and identify the 15% increase in phishing incidents as compared to 2019. (Warburton & F5 Labs, 2020). The effect of the global pandemic contributed to the rise of phishing incidents to about 220% as compared to the past years. While attempting to evolve in 2020, Warburton & F5 Labs(2020) reports that phishers were targeting 55% of the sites relating to brand names and their URLs. Europol's organized crime threat assessment (IOCTA) states that social engineering and phishing remain a big threat and both demonstrate a significant increase in volume. Figure 2.2 shows the phishing report in the 3rd quarter of 2020.

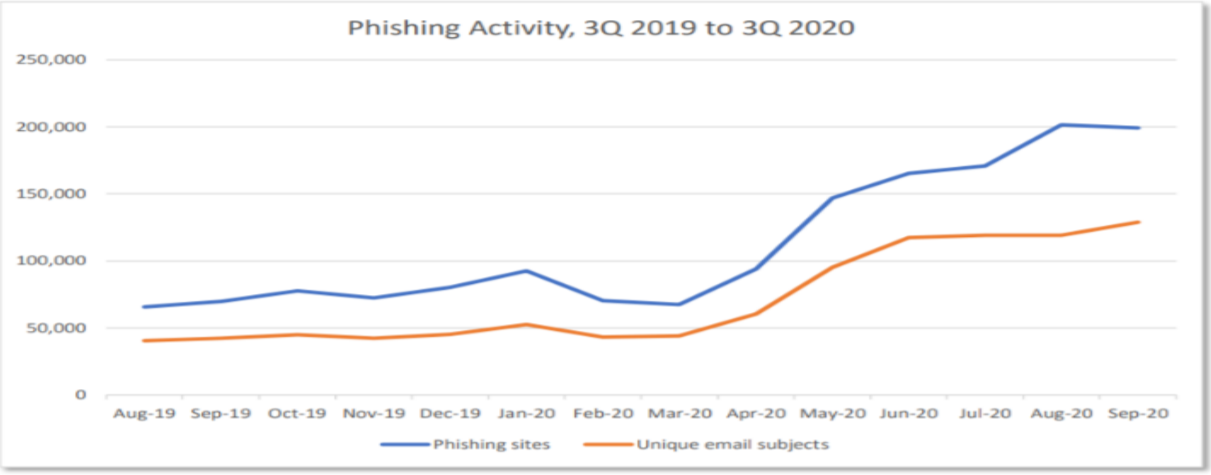


Figure 2.2: The Phishing Activity Trends Report, 3rd Quarter 2020 (APWG, 2020)

APWG report analyzes phishing scams relating to both social engineering attacks and identity theft. The number of phishing attacks declined during the first quarter of 2021 but Business e-Mail Compromise (BEC) attacks increased from $48,000 previously recorded in Q3 2020 to $85,000. However, in January 2021 the highest rate of phishing attacks in this quarter was 245,771 attacks as stated by the APWG records. With financial institutions, webmail, and social media sectors still predominately attacked by phishers in this quarter. (Phishing Activity Trends Report 1st Quarter 2021, 2021). In 2020, the number of phishing attacks observed by the APWG peaked in January 2021, with 24,771 new phishing sites debuting in that month alone, setting a new high. The number of attacks then began to reduce in February and March, giving consumers hope for the future. Despite this, March 2021 was the fourth-worst month in

APWG history, with over 200,000 attacks. Figure 2.3 shows the phishing report of the 1st quarter of 2021.

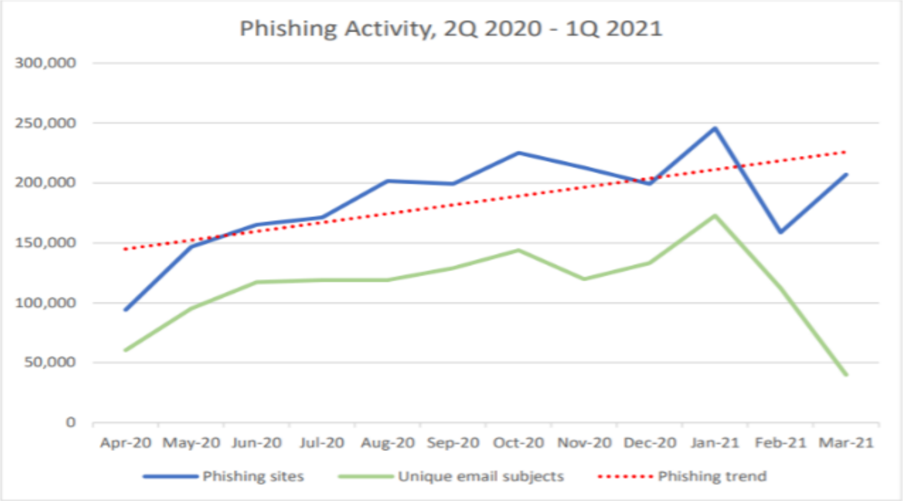


Figure 2.3: Phishing Activity Trends Report, 1st Quarter 2021 (APWG Report)

Rasool & Jalil (2020) refers to phishing as a large-scale attack strategy. In March 2020, ZOOM, a video-conferencing platform received eight reports from APWG relating to phishing attacks. The month of April shows a significant increase in ZOOM phishing-related attacks. (APWG, 2020).

2.1.1 How Phishers use Encryption to fool victims

PhishLabs provides solution to digital risk protection and cyber-attacks. PhishLabs explains how phishing websites deceive internet users by exploiting the HTTPS internet security mechanism (APWG, 2020). The track numbers of phishing sites show that phishing sites are now protected by HTTPS encryption protocol. The HTTPS means are used to ensure secure communication by encrypting data being exchanged between a person's browser and a website visited. It protects credit card information and online sales both during and after a transaction.

APWG (2020) reports that 80% of phishing sites have SSL encryption enabled. Figure 2.4 shows the typical structure of the phishing attack cycle.

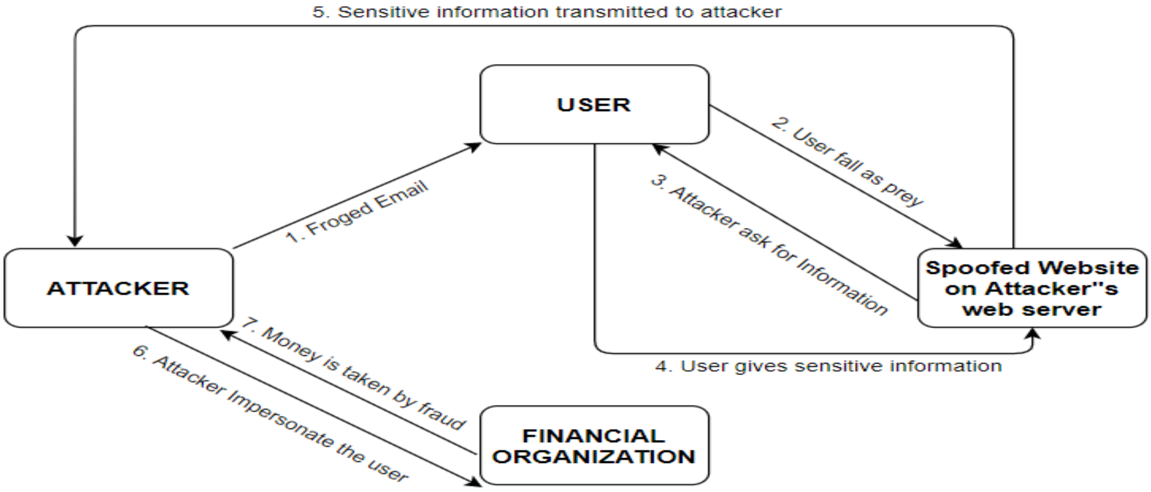

5. Sensitive information transmitted to attacker
USER
2. User fall as prey
3. Attacker ask for Information
1. Froged Email
ATTACKER
Spoofed Website on Attacker"s web server
4. User gives sensitive information
7. Money is taken by fraud
6. Attacker Impersonate the user
FINANCIAL ORGANIZATION

Figure 2.4: A typical phishing attack scenario

(Basit et al., 2020)

In Figure 2.4, an attacker creates websites by mincing or copying the content of a legitimate website to target victims through any on the phishing methods used in phishing attacks.

2.1.2 Use of Domain Names for Phishing

Rusk IQ, a member of APWG analyses incidents where phishing is happening in the domain name system. Risk IQ reports that 2019 confirmed phishing URLs show that 1,274 were hosted on a unique second-level domain and 15 were hosted on unique IP addresses without a domain. (APWG, 2020)

In risk IQ analyses, it shows that 1,565 unique domain names used for phishing during Q3 out of 13,567 confirmed phishing URLs were involved in attacks against the financial sector. According to Risk IQ, hosting providers seem to be more proactive at the beginning of the year since 1.5% of unique phishing domains active in Q1 were still active when scanned for Q2. In recent research, APWG members at the Interisk consulting group show that most phishing is concentrated on small numbers of hosting providers.

2.2 An overview of uniform resource locator

The objective of the project is to identify a phishing or legitimate URL. Therefore, it is important to understand the components of a URL. In this section, we will discuss the structure of the URL.

The Uniform Resource Locator (commonly known as a web address) is the global address of documents and other resources on the World Wide Web. The main goal is to find a document or other resource on the internet and to specify how to access it in a browser. The author describes a URL as the unique identifier used to find an online resource. The URL is made up of different parts, which tells a web browser how to acquire a resource and where to get it.

2.2.1 The structure of a URL

The URL is made up of the protocol, domain name, and path. The first section of the URL is the protocol that is separated by a double slash from the remaining complete URL. The domain name or the Internet Protocol (IP) address are separated by a dot and the third part, known as the path is separated by a single slash. The protocol includes HTTP (Hypertext Transfer Protocol) and HTTPS (HTTP Secure) for web resources. The domain might define a URL for a specific website or a network port to connect with. The parts of a URL is shown in figure 2.5 below.

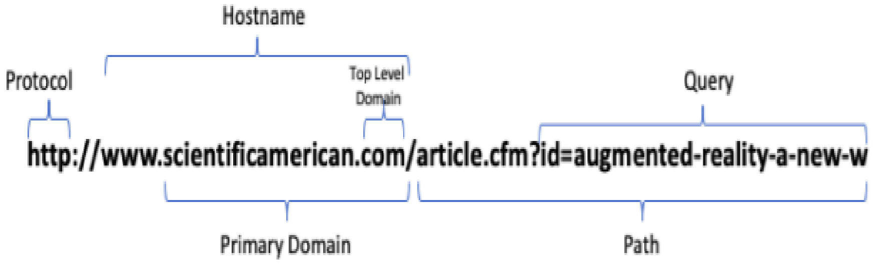

Hostname
Protocol
Top Level Domain
Query
http://www.scientificamerican.com/article.cfm?id=augmented-reality-a-new-w
Primary Domain
Path

Figure 2.5**:** Example of a URL

Using the URL https://home.techgiant.com/search/query?q=URL as an example, components of a URL can include:

i. **The protocol or scheme**: The protocol is used to access internet resources by identifying the type of protocol being used. HTTP, HTTPS, File Transfer Protocol Secure (FTPS), Mailto, and file are examples of protocols.

ii. **Domain name:** The domain name system (DNS) name is used to access the resource. The unique reference that represents a webpage.

iii. **Port name:** This is a necessary part of the URL and it is usually not visible. If a port number is indicated, that number is followed by the hostname and separated by a colon. Port 80 is the default port for web servers.

iv. **Path:** A path is referred to a file or location on the webserver. For this example, search/query in the URL above is the path. This is used to identify the specific resource in the hostname that the web client wants to access. The pathname begins with a single forward slash.

v. **Query:** A query string is a string that is used to search for information. If a query string is used, it follows the path component and returns a string of data that the resource can utilize for whatever reason (for example, as parameters for a search or as data to be processed). A question mark precedes the arguments in the query. The query string is usually a string of name and value pairs; for example, term=bluebird. Name and value pairs are separated from each other by an ampersand (&); for example, term=bluebird&source=browser-search.

2.3 Types of Phishing

Phishing is a type of fraud in which hackers send messages posing as reputable organizations to different people. Despite the fact that these attackers all have the same goals, their attack techniques differ. Organizations can better safeguard their users and data if they have a better awareness of the types of phishing scams and how to recognize them. We will discuss the types of phishing method used by attackers in this section.

2.3.1 Spear phishing

In Spear phishing a particular individual is targeted and basic information about such victim is usually gathered before sending a phishing email. The most prevalent platforms for spear-phishing are social media sites like LinkedIn, where they may easily get information about a person's occupation.

2.3.2 Whaling

Whaling is an even more targeted type of phishing than just spear phishing as it goes after the whales as the name implies. The target of these whaling attacks is the CEO, CFO, or board members within an organization or specific business. Whaling is the most serious attack since executive bands have access to the most sensitive information in the business. (Fahmida Y. Rashid, 2020)

2.3.3 Pharming

Pharming is another variation of phishing. In this case, no particular person is targeted instead the attack is directed towards a large number of people. Pharming is done through a technique called DNS Poisoning. The system's localhost files are not corrupted and the domain name system table is modified. These results retrieved from the attack are been redirected to malicious websites without the victim's knowledge. The victim would assume the target is accessing legitimate websites but the DNS poisoning is included in the domain. Another method for pharming occurs where an attacker sends a code through email whose target is to modify and access localhost files in the system. The URLs would be converted by the host files to number strings, used by the system to access websites. This leads the target to the malicious site despite originally heading to a legitimate site.

2.3.4 Smishing

This type of phishing attack makes use of text messaging or SMS (short message service) to get the attention of its victim. A common scenario on a smishing attack would be when a message containing a link is sent through a phone and the users click the malicious link. Then the attacker requests that the victim inputs their bank account number, SSN, etc. This gives the attacker access to control their bank account.

2.3.5 Vishing

A vishing attack occurs through a phone call. The attackers are still pursuing the user's personal information or critical corporate information, as in all of these phishing assaults. (Fahmida Y. Rashid, 2020)

2.3.6 Email phishing

These are the emails that a hacker can send to anyone's email account if they have it. The email usually tells the recipient that their account has been compromised and they need to respond immediately by clicking on this link. It is easy to recognize a dubious source if you verify the email source and the actual link that you are being pointed to.

2.3.7 Search engine phishing

Hackers use search engine phishing, also known as SEO poisoning or SEO trojans, to try to become the first result on Google or other search engines. When their link is clicked on, the person is redirected to their (hacker) website. The attacker tricked the victims into divulging sensitive data.

2.3.8 Deceptive phishing

Deceptive phishing is the most prevalent type of phishing attack. It entails impersonating a legitimate website and sending an email to the intended recipient that appears to be legitimate. The email sent would contain a malicious URL or link. It would instruct the target to click on the URL. When the user follows the instructions, the phishing website collects all of the target's login credentials and other sensitive information and sends it to the attacker. For example, games@cartoonnetwork.com uses a lowercase 'c' that could be removed. Hence, games@artoonnetwork.com could trick the target and thereby obtain data. If the outcome of a phishing assault is to make sure that the victim is redirected to the phishing internet site without being aware of it, different strategies such as URL hiding, Homograph spoofing are used. In Homograph, the attacker replaces characters in a domain name with different visually comparable characters. (Rouse et al., 2019). Other techniques may include Typosquatting where the objective of the attacker is to make typographic mistakes in domain names. An instance is an example in which an attacker uses a domain name similar to a popular domain to prey on users who can't spell correctly.

2.4 Past works

Phishing is one of the most predominant methods of cybercrime and was first discussed in a newsletter in 1996 after an attack on American Online (AOL) accounts (Ollmann, 2008). According to ("Verizon: 2019 Data Breach Investigations Report," 2019), phishing amounts to 78% of all Cyber-Espionage. Widup (2018) reported that 22% of phishing victims that year clicked on the phishing links sent and only 17% reported the incident. Thabtah & Abdelhamid (2016) research work compares multiple datasets that are used in various phishing techniques. Here, (Thabtah & Abdelhamid, 2016) bases this research on using various machine learning methods as techniques for phishing detection showing the performance and effectiveness of the learning models. The paper on “Discovering Phishing Target Based on Semantic Link Network” by Wenyin et al., (2010), proposes a new method for predicting malicious websites by combining the linkage, search relationship, and text relationships between phishing webpages and the connected web pages. The model Link network proposes a way to recognize the suspicious website as phishing based on four convergent scenarios. This method takes a long time and requires a lot of association to be carried out.

The Evolving Fuzzy Neural Network for Phishing Emails Detection addresses and emphasizes zero-day phishing. It differentiates online phishing and ham e-mails. The author's approach is based on feature and grade fetch, and clustering of attributes of related emails. This methodology is based on the binary classification to yield the results for all features implemented in this method in which 1 is malicious and 0 is a legitimate characteristic. This model does not consider the dynamic system in this procedure, therefore, making it less efficient to achieve accurate results (Almomani et al., 2012). The Link Guard algorithm performs some tests, such as a comparison of the DNS of current and visual links while examining the number of decimal points of the IP address. The demerit of the Intelligent Phishing Website Detection and Prevention System presented by (Naresh, 2013) returns false-positive results for any legitimate site with an IP address instead of a domain name. This leads to erroneous errors and negative findings of the model (Madhuri et al., 2013). In this paper, the authors (Preethi & Velmayil, 2016) introduce a PrePhish method that allows real-time detection of phishing URLs. Malicious websites are identified using four different types of URL features: domain-based, address-based, abnormal-based, and HTML-based. These features are extracted from the sample dataset used by the author and analyzed using the machine learning technique. The presence or absence of an attribute is considered in generating threshold values for the features that classify whether a URL is phishing and legitimate. The author achieved 97.83% for correctly classified phishing URLs and the 2.17% rate for incorrectly classified phishing URL recorded as legitimate. The PrePhish method was implemented in MATLAB and the classification was done using Naïve Bayes, Random Forest, and Support Vector Machine classifiers.

The proposed strategy for finding phishing by appropriately identifying and using structural aspects of e-mail is explained in e-mail detection based on structural properties by (Chandrasekaran et al., 2006). The experiment is performed using SVM and phishing mail classification algorithm. The method used for this classification process is insufficient and employs only one strategy to identify the efficiency and scalability of phishing emails. This is primarily dependent on the structural aspects of email. To reduce incorrect results, it must expand further structural or content properties.

Baykara & Gürel (2018) propose a model called an Anti-phishing simulator that doesn't allow a phishing attack to occur. This model gives detailed information about the occurrence of phishing attacks and how to identify them. The database model is taken as an input for detecting which of its samples consist of spam while using the Bayes algorithm. The Anti-phishing simulator was developed as a method of identification involving the process of carrying out complex kinds of word processing where keywords are gotten from sample text of the phishing mechanism. Han et al., (2012) proposes a solution that allows the system to defend against phishing attacks by combining visual similarity-based techniques and white lists.

In (Sahoo, & Hoi, 2017), the authors discuss the works done during the early stages of malicious URL detection where blacklisting, regular expression, and signature matching were mostly used for URL detection. The problem this model faced was that they were unable to detect new or variants for previous URLs and it required that the database needed to be updated regularly. Due to these challenges, machine learning algorithms were introduced to detect malicious and phishing URLs efficiently. The use of feature engineering to extract good features from the URL requires extensive domain knowledge of the URL. In (Sahoo et al., 2019) research, they propose the use of lexical features (Ma et al., 2009b), host-based features (Ma et al., 2009a), blacklist features (Felegyhazi et al., 2010), content features (Canali et al., 2011), and popularity features as a combination of features used in the classification model (J. Cao et al., 2016). Felegyhazi et al., (2010) use the blacklist features to predict by checking the presence of a URL in a blacklist database while lexical features (Ma et al., 2009b) are gotten through the string properties of a URL. Example of these are: checking for redirection, the number of special characters, length of URL, existences of HTTPS tokens in the domain or URL, etc. Host-based features are gotten from the hostname properties of the URL such as the WHOIS information, IP Address, Geographic location. When an unknowing user views a webpage through a malicious URL, content features are extracted from HTML and JavaScript. Another feature needed in feature extraction is the information relating to their ranking, popularity scores, and source of sharing which are considered as the content features. (Le et al., 2018). This approach discusses the challenge faced by feature selection because it requires extensive domain knowledge to obtain features from a raw URL. Blum et al., (2010) research results show that lexical features give good performance and it is easier to use for comparison than other features. According to (Srinivasan et al., 2021) statistical properties of the website string such as length of the URL, the total amount of special characters (Kolari et al., n.d.) were the most widely utilized features. While Bag of Words (BOW), term frequency approaches such as term-document matrix (TDM) and term frequency, and inverse document frequency(TF-IDF) and n-gram features (Ma et al., 2009a) were also popular. All of these attributes are ineffective at extracting the URL's sequential order and semantics (Srinivasan et.al, 2021). Recent research looks into using deep learning with character-level embedding to detect malicious URLs. Vinayakumar et al., (2019) compares the detailed analysis of deep learning with character level embedding and traditional machine learning with feature engineering methods for URL phishing detection. Traditional algorithms performed worse than deep learning architectures. Bahnsen et al., (2017) employ the use of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) for URL detection. Bahnsen et al., (2017) use random forest classifier and a combination of lexical features and statistical URL analysis. Although both models performed admirably, the LSTM outperformed the traditional machine learning models. Content analysis and blocklists were used in traditional detection classifiers with a focus on predictive filtering of URLs. This study developed a model that detects malicious URLs within a platform called Splunk. Support Vector Machine (SVM) and RF algorithm was used to classify malicious and benign URLs. The RF algorithm has a precision of 85% and recall of 87% while SVM has a 90% precision score and 88% recall score. (Christou et al., 2020)

Alazab & Fellow (2020) offer DeepURLDetect (DUD) in which raw URLs are encoded using characters level embedding rather than feature engineering resources, which must be updated regularly to handle new URLs or changes to existing URLs. Character Level Embedding (CLE) is a cutting-edge Natural Language Processing (NLP) method for encoding characters in a numeric format that can be used to extract optimal attributes from raw URLs in order to improve performance. Hidden layers in deep learning architectures record character-level embedding features and then assess the possibility that the URL is fraudulently constructed using a feed-forward network with a non-linear activation function. In Alazab & Fellow's (2020) paper, they compared and contrasted multiple state-of-the-art deep learning-based character level embedding approaches for detecting malicious URLs. Furthermore, because the embedding captures the sequence of URL characters, deep learning architectures based on character-level embedding models outperformed n-gram representations. More experiments are performed to attain the best learning-based character level embedding model.

By providing a feature selection algorithm, this research contributed to enhancing the accuracy of phishing website detection (Ubing et al., 2019). This algorithm is combined with an ensemble learning model associated with majority voting. Then compared against different classification models including Random Forest, Logistic Regression, Prediction model. The experiment yielded a 95% accuracy rate, and the learning model utilized suggests that the proposed technique could be more effective at detecting URL phishing.

Sahingoz et al., (2019) use seven different classification algorithms and NLP based features, for the proposed anti-phishing model than the use of blacklists, heuristics, visual and machine learning-based approaches. According to results obtained from experiments, the Random Forest algorithm with only NLP-based features gives a 97.98% accuracy rate for the detection of malicious URLs. These are machine learning techniques that are based on the traditional model. The authors developed a methodology based on the Alexa and PhishTank datasets. (Basit et al., 2020). Naïve Bayes (NB), DT, KNN, and SVM were the four classifiers employed. (Abdelhamid et al., 2017) paper compares model content and features in the

PhishTank dataset with 11055 URLs. Dynamic rule induction (eDRI) is the first machine learning and deep learning algorithm to be employed in an anti-phishing tool, according to the inventors. Another technique in (Mao et al., 2019) research that considers the page layout attribute and 49 phishing websites gotten from the PhishTank.com dataset were used. Analyses from over 20000 messaging samples were tested on the four machine learning classifiers listed: Support vector machine (SVM), AdaBoost, DT and RF. (Jain & Gupta, 2018) reports phishing attack detection accuracy was improved up to 99.09% by applying RF, SVM, LR, and NB on multiple datasets. The first dataset was from PhishTank, which contained 1528 phishing sites, followed by OpenPhish, which contained 613 phishing sites, and Alexa, which contained 1600 URLs. Another work based on ensemble learning was proposed by (Ubing et al., 2019), where three techniques bagging, boosting, stacking were used. There were 30 features in the dataset, with 5126 records in the result column. (Hota et al., 2018) introduced a remove replace feature selection approach (RRFST) and tested it on a phishing email dataset from the khoonji's anti-phishing website, which contained 47 features. Their suggested model reduced 30 to 11 features. Another popular approach is the Random forest method which is used for the detection of a phishing attack. RF has been employed by various researchers in the past, with encouraging results. The authors (Subasi et al., 2017) used six classifiers which are Artificial Neural Network (ANN), KNN, SVM, RF, Rotation Forest, and C4.5. In another work, (Tyagi et al., 2018) used a dataset from the UCI (University of California, Irvine) machine learning repository and utilized a strategy that involved extracting 30 attributes from URLs. Those features are used to forecast phishing attempts. A decision tree, random forest, Gradient Boosting (GB), a Generalized Linear Model (GLM), and Principle Component Analysis were all used (PCA).

Joshi & Pattanshetti (2019) proposed their study based on the Random Forest algorithm as a binary classifier and Relief Feature Selection (reliefF) algorithm used as a forward selection approach using 48 features. Mao et al., (2019) proposed a method using a page layout feature on 49 phishing website datasets from Phishtank.com and over 20000 texting sample was used in their research. They used four learning classifiers namely SVM, Decision Tree (DT), Attribute Bagging (AB), and RF. Sahingoz et al., (2019) created a dataset with 73575 URLs, including 36400 authentic and 37175 phishing URLs, using seven classification algorithms and NLP-based features.

2.4.1 Scenario-based techniques

At a workplace, scenario-based approaches to detect phishing assaults were utilized to improve phishing detection results. Yao, W. described an approach that involved two steps: logon extraction and identity detection. (Yao et al., 2019). The logon extraction process takes the image from the two-dimensional code and extracts the logo from it. The identity detection technique evaluates the similarities between the two websites to determine whether it is a legitimate or phishing website. They used 726 web pages to construct two non-overlapping datasets.

Curtis et al., (Curtis et al., 2018) suggested and worked on many people ranging from 50 to 2885 characters, as well as Dark Traid attacker conceptions. With both attackers, they used the Dark Traid score to create a 27-item short dark triad. The scenario required the participation of end-users. Psychopathy, narcissism, and Machiavellianism were used to calculate the score. (Williams et al., 2018) research was set in a workspace environment and didn't use a dataset; instead, they conducted two studies that looked at different characteristics of emails, such as the emails that were received, the person who got them, and the context of the emails. Theoretical development was facilitated by these theoretical approaches. For six weeks, the authors observed 62 employers and their employees, as well as targeted phishing emails, which are known as spear phishing. The authors planned and worked with 985 people who performed a role in scenario-based phishing research. This is a two-way repeated-measures analysis of variance (ANOVA) that was used to determine the effect of email legitimacy and influence. (Parsons et al., 2019).

2.4.2 Hybrid Techniques

Hybrid approaches work by integrating different classifiers to improve the accuracy of phishing attack detection. In a hybrid context, (Patil et al., 2018) suggested a hybrid approach that combined blacklist and whitelist, heuristics, and visual similarity. Suspicious websites, phishing websites, and benign websites were the three outcomes used for monitoring every traffic on the end-user machine and compares each URL to a trusted domain whitelist. During classification, LR, DT, and RF were used to predict and determine the threshold value. 9076 test websites were collected and used to detect phishing attacks and generate an accuracy score. The ensemble strategy based on voting and stacking methods with decreased characteristics was proposed and applied by (Deepa et al.,2018). They used a dataset from the UCI machine learning repository and used 23 out of a total of 30 features. They employed a hybrid EKRV model that combines KNN techniques. The authors used URLs from PhishTank and OpenPhish to create 5000 phishing web pages (Abdelhamid et al., 2017). They employed RF and SVM hybrid models to predict accuracy on a dataset with 10 attributes and 1353 examples. (Pandey et al., 2020). Existing phishing assault detection solutions have some drawbacks in terms of accuracy and usability. Some of these techniques are computationally complex, time-consuming, and efficient. The results of ensemble-based approaches were good, but they may be even better with fewer features. In this study, an unique ensemble model was utilized to evaluate and enhance results given by several classifiers. This study proposes a hybrid model that uses KNN, ANN, and Decision Tree (C4.5) with Random Forest Classifier to detect phishing attacks on a website. This model has a high accuracy of 97.3% which shows that it performs better than previous models. The result shows that a combination of KNN and RF produces the best accuracy score (Basit et al., 2020).

2.5 Traditional Machine Learning

A brief review of the different machine learning algorithm used for classification problems in this project are discussed in this section.

2.5.1 Logistic Regression (LR)

Logistic regression also called the logistic model or logit model. LR analyzes the relationship between a categorical dependent and many independent variables, and calculates the probability of an event occurring by fitting data to a logistic curve. The logistic curve is S-shaped or sigmoid curve that begins with moderate, linear growth and then accelerates exponentially. LR is a method of fitting a regression curve, y=f(x),that estimates the probability of a particular outcome depending on individual variables. Logistic regression fits a logistic curve to the relationship between x and y when the output is a binary variable and x is numerical. (Park, 2013)

2.5.2 Decision Trees (DT)

The structure of DT is similar to that of a normal tree, and it contains multiple branches. One of the algorithm used in statistics, data mining, and machine learning is decision trees. The data analysis technique in this case divides the data into many possible entities related to a given parameter. Nodes and leaves are the entities that make up a decision tree. Each leaf is assigned to one class representing the most appropriate outcome or target value to an issue. (Rokach & Maimon, 2006). The assignment of attributes to the parent or root node is a major challenge in this statistical modelling. Furthermore, the tree's farthest branches symbolize the final findings. These trees are extremely important in decision-making. They present all possible outcomes at any time, allowing you to visualize all possible scenarios.(Charbuty & Abdulazeez, 2021)

2.5.3 K-Nearest Neighbor (KNN)

The central concept behind KNN is to predict a query instance's label based on the labels of k nearest instances in the stored data, assuming that an instance's label is close to that of its KNN instances. In terms of prediction performance, KNN is quite effective, and it makes no assumptions about the data distribution.(Kang, 2021). The theory behind KNN is as follows: first, calculate the distance between the new sample and the training sample; then discover the nearest Kneighbors; finally, determine the category of the new sample based on the category to which the neighbor belongs, if they all belong to the same category. The new sample is then placed in this category as well; alternatively, each post-selection category is recorded and the new sample category is selected using specified rules. (Wang, 2019)

2.6 Types of Ensemble Methods

In this section, a concise overview of the different ensemble learning algorithm used for classification problems in this project are discussed.

2.6.1. Bagging

Bagging is gotten from bootstrap aggregating. Bagging is used in ensemble system for machine learning classification algorithm. It increases the accuracy of models through the use of decision trees, which reduces variance and increases accuracy hence eliminating overfitting. Bagging is classified into two types, i.e., bootstrapping and aggregation. The advantage of bagging over other models is that the combination of weak base learners allows the development of a single strong model that gives better prediction and accuracy. Figure 2.6 represents the structure of a bagging algorithm. (Charu Makhijani, 2020)

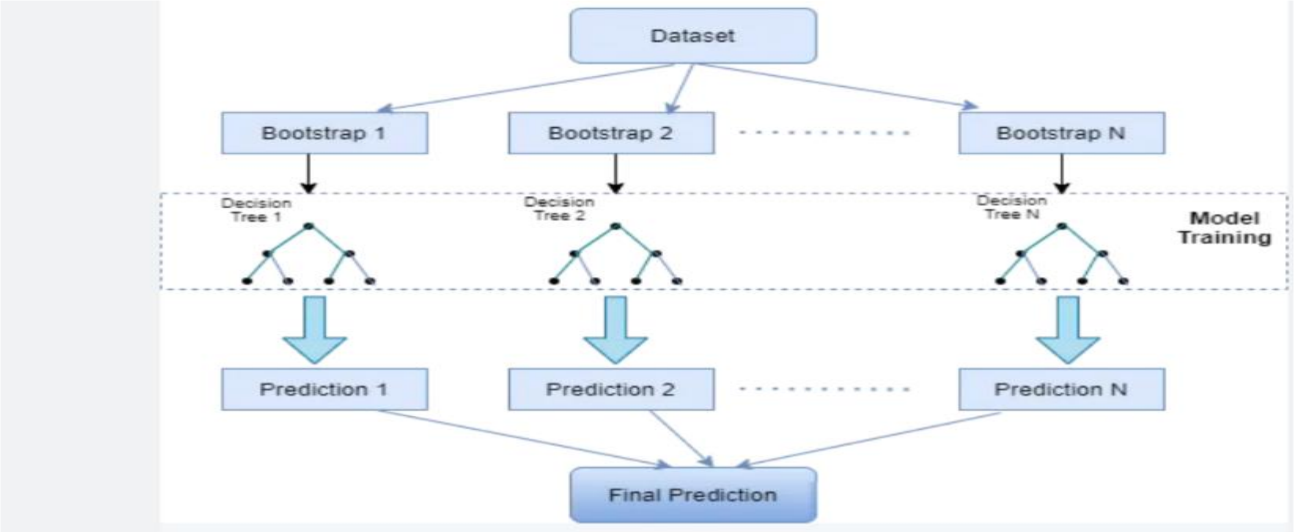
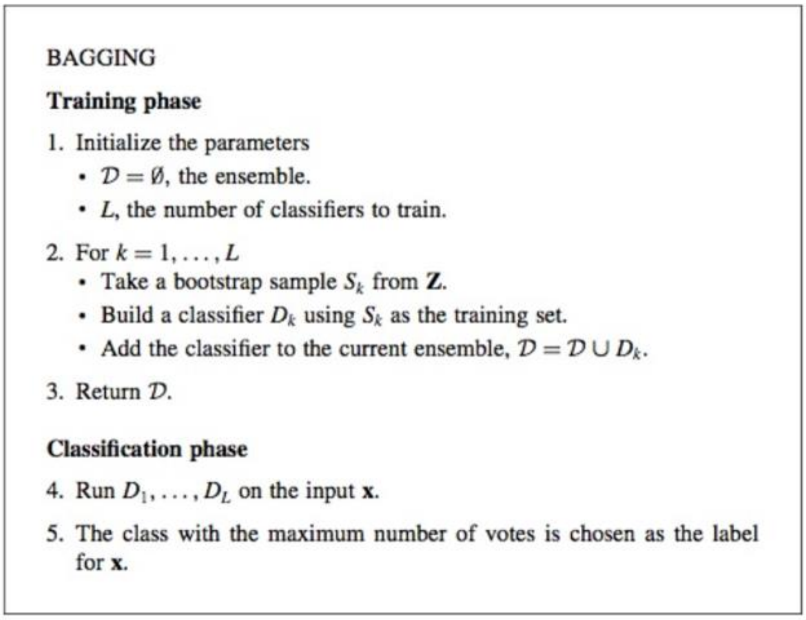


Figure 2.6: Structure of the bagging algorithm

In bagging, we create bootstrap samples of a training set using sampling with replacement. A distinct type of base classifier is trained with each bootstrap sample generated. Bagging is useful for classifiers that are inherently unstable, such as Neural Networks and Decision Trees. Bagging reduces variation and lowers the above-mentioned errors, according to past research. This is a method for combining more than one base classifier whose mixed overall performance is appreciably higher than that of any of the bottom classifiers. Each classifier votes to obtain an outcome of the model. (Polikar, 2009). The bagging algorithm is shown below in Figure 2.7.



```
BAGGING
Training phase
1. Initialize the parameters
    • $\mathcal{D} = \emptyset$, the ensemble.
    • $L$, the number of classifiers to train.
2. For $k = 1, \dots, L$
    • Take a bootstrap sample $S_k$ from $\mathbf{Z}$.
    • Build a classifier $D_k$ using $S_k$ as the training set.
    • Add the classifier to the current ensemble, $\mathcal{D} = \mathcal{D} \cup D_k$.
3. Return $\mathcal{D}$.
Classification phase
4. Run $D_1, \dots, D_L$ on the input $\mathbf{x}$.
5. The class with the maximum number of votes is chosen as the label
   for $\mathbf{x}$.
```

Figure 2.7: Bagging Alogrithm (Polikar, 2009)

2.6.2 Random Forest

Random Forest (RF) model is quite similar to bagging. Bagged Decision Trees have a wide range of features to pick from when deciding where to split and how to make decisions. As a result, the bootstrapped samples may change slightly, the data will typically break off at the same features in each model. On the other hand, the RF model decides where to split based on a random collection of features. RF model implements a level of difference because each tree will divide based on different features, rather than dividing at comparable features at each node throughout. As a result of the higher level of differentiation, there is a larger ensemble to aggregate over, resulting in a more accurate predictor. On each subsample, a decision tree is created. (Evan Lutins, 2017)

2.6.3 Boosting

Boosting is an ensemble technique that learns from previous predictors' mistakes to make better predictions. Each new tree is a fit on a modified version of the original dataset and focuses on reducing the bias by joining several weak base learners to make one stronger learner, therefore improving the predictability of models. Each learner is trying to correct the predecessor. In boosting, weak learners are arranged in sequence, therefore, allowing weak learners to learn from the next to create better predictive models. Boosting can be gradient boosting, Adaptive Boosting (AdaBoost), and XGBoost (Extreme Gradient Boosting). (Polikar, 2009). The figure 2.8 (Tyagi et al., 2018) represents the structure of a boosting classifier

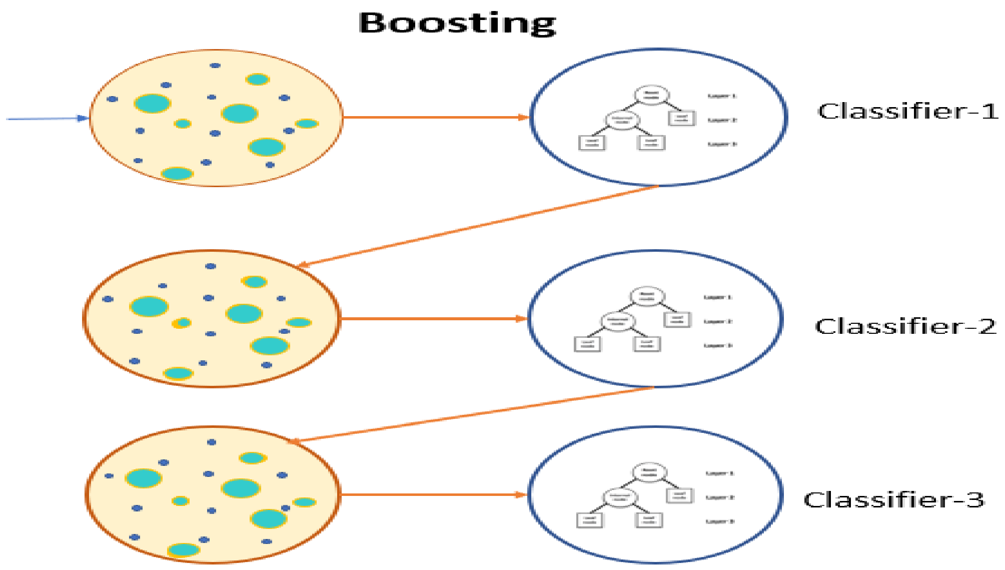


Figure 2.8: Structure of boosting algorithm

2.6.3.1 Adaboost

This boosting algorithm only supports binary classification. A weak classifier is prepared on training data by adding weighted samples and mixing multiple weak models into one strong classifier. In AdaBoost, equally-weighted data are applied first on the classifier. Subsequently, the weights of misclassified data are increased and are passed to a second classifier. We continue emphasizing and changing weights of misclassified data until all training data are appropriately classified. (Alvaro, 2021)

2.6.3.2 Gradient Boosting (GB)

GB adds predictors gradually and sequentially to the ensemble, where preceding predictors correct their successors, thereby increasing the accuracy of the model. The new predictors are fit to eliminate the errors in the previous predictors. The gradient of descent helps the gradient booster in identifying problems in learners' predictions and countering them accordingly. Gradient boosting consists of three components: an optimized loss function, a weak learner that makes predictions, and an additive model that adds weak learners to reduce the loss function.

2.6.4 Stacking

Stacking involves training of different classifiers of and combining the predictions of the several learning algorithms used. This ensemble method is often referred to as stacked generalization. This method works by allowing a training algorithm to combine the predictions of numerous other similar learning algorithms. Regression, density estimations, distance learning, and classifications can all benefit from stacking. In addition to selecting multiple sub-models, stacking allows the addition of an extra model known as the meta-classifier to allow the combination of the feature vectors to be trained again. The model of a stacking classifier is shown in figure 2.9.

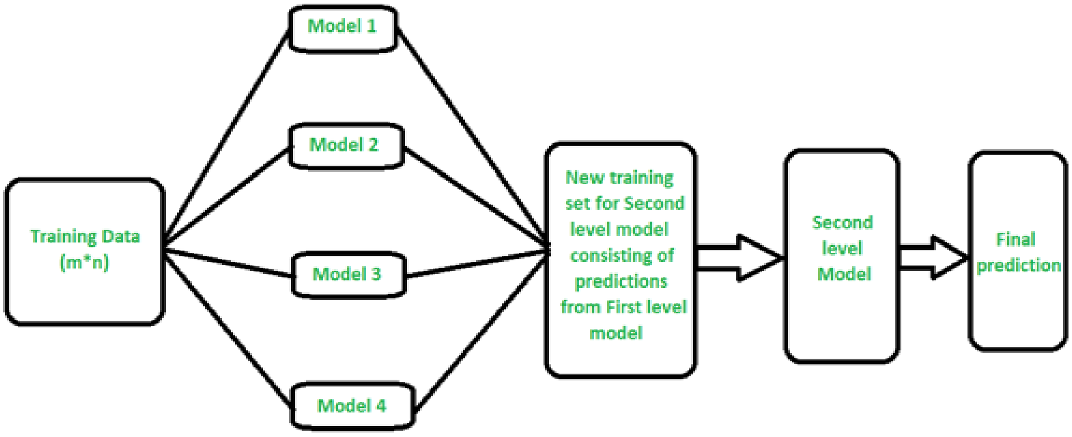

Training Data (m*n)
Model 1
Model 2
Model 3
Model 4
New training set for Second level model consisting of predictions from First level model
Second level Model
Final prediction

Figure 2.9: Diagram of the stacking algorithm

2.7 Basic Concepts and Terminologies

In this section, we will discuss the common terminologies associated with machine learning. It is essential to deal with drawbacks, errors, and enhance predictions in any machine learning model, we must first comprehend the impact of bias, variance and noise on ensemble and machine learners.

2.7.1 Bias

In data science, bias is referred to as an error in the data. It can also be seen where an algorithm cannot learn the target. This type of error is often derived from the expected data and can be overlooked. An appropriate statistical method can reduce the effect of bias on a predicted model.

2.7.2 Bootstrapping

Bootstrapping is a general approach to statistical inference based on resampling from the available data to create samples of the main data. There are several forms of the bootstrap, and other resampling methods such as cross-validation, randomization tests, and permutation tests.

2.7.3 Variance

Variance refers to the changes in the model when different subsets of the dataset are trained. The variance affects the variability in the model predictions and its learning process.

2.7.4 Noise

Noise is an error by the target function.

CHAPTER THREE

METHODOLOGY

Phishing is a kind of fraud and the Uniform Resource Locator (URL) is the main source of spreading attacks. In this research, we are proposing the use of ensemble learning systems to improve accuracy when classifying URLs. The problem of determining whether a given website URL is phishing or not is a binary classification problem that can be solved using labelled data in a supervised learning environment.

This chapter discusses the collection of datasets containing recent website URLs from both malicious and benign sources. The dataset is prepared by extracting important features that aid in the differentiation of phishing websites from authentic ones. To provide input to the machine learning algorithm, the features are preprocessed. The model is then trained on the training set before being tested on the testing set to verify its accuracy.

3.1 Proposed Method

In this section, we described our proposed ensemble approach to improve the detection of phishing attempts on websites. The steps taken to achieve the proposed model is shown in Figure 3.1 below. The raw dataset is preprocessed and prepared for classification algorithm to evaluate its' performance.

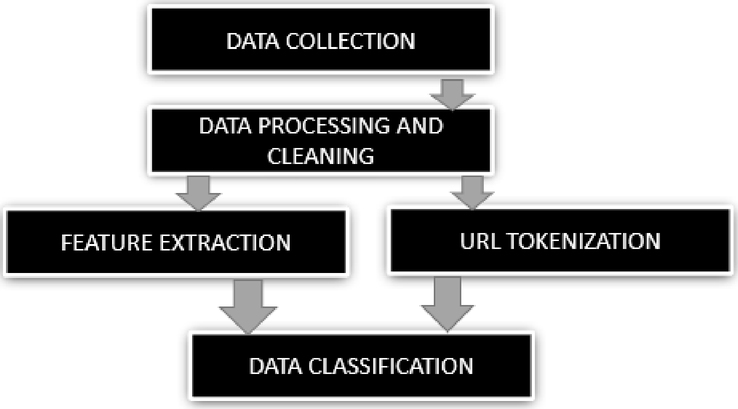


Figure 3.1: Steps to achieve URL classification

3.2 Architecture of Models

The figure 3.2 and 3.3 represents the architecture of the stacking and bagging classifier used in this research.

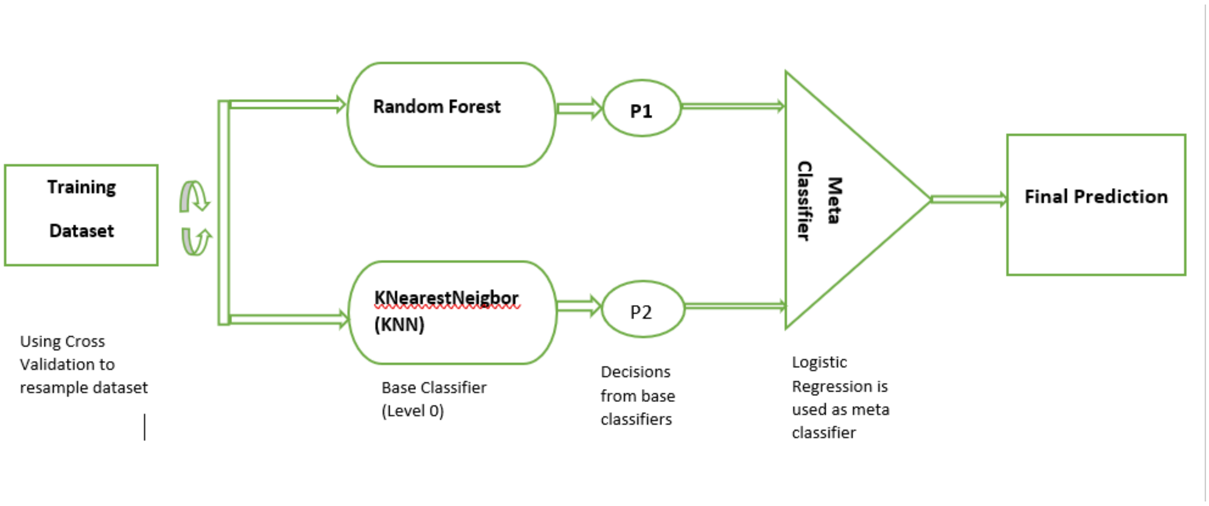

Training
Dataset
Using Cross Validation to resample dataset
Random Forest
KNearestNeigbor (KNN)
Base Classifier (Level 0)
P1
P2
Decisions from base classifiers
Meta Classifier
Logistic Regression is used as meta classifier
Final Prediction

Figure 3.2: Architecture of Stacking classifier model

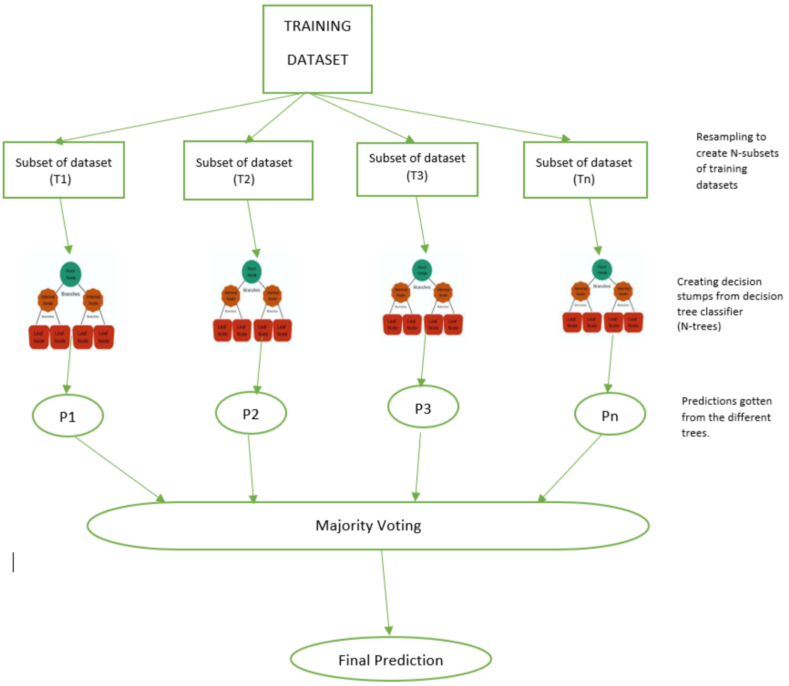

TRAINING
DATASET
Subset of dataset (T1)
Subset of dataset (T2)
Subset of dataset (T3)
Subset of dataset (Tn)
Resampling to create N-subsets of training datasets
Creating decision stumps from decision tree classifier (N-trees)
P1
P2
P3
Pn
Predictions gotten from the different trees.
Majority Voting
Final Prediction

Figure 3.3: Architecture of Bagging classifier model

3.3 Data Collection and Dataset Description

The URLs are gotten from PhishTank. Each URL is described by features and all features are grouped under Lexical Based Features (Address Bar Based features), HTML, and JavaScript-based features. These features are either categorical or binary. The frequency count of each feature we have been extracted to assist in carrying out predictive analysis. These extracted features are passed as input to the machine learning algorithm. The dataset is split into training and testing datasets where the training set is used for training and the testing set is used to ascertain the accuracy of the model. A set of 5000 phishing URLs are gotten from PhishTank and 5000 legitimate URLs are obtained from the open datasets of the University of New Brunswick. In the second dataset, we used a standard publicly available dataset from the Kaggle repository which has been used in several state-of-the-art studies to detect phishing schemes in past research. In the first and second datasets used for this research scope contains 3000 phishing URLs and 3000 legitimate URLs each.

Dataset 1: 3000 phishing URLs from PhishTank and 3000 legitimate URLs from open datasets of the University of New Brunswick.

Dataset 2: 3000 phishing and 3000 legitimate URLs both gotten from Kaggle repository.

3.4 Data Pre-processing

Data cleaning is done to remove all duplicate entries, fill in missing attributes or class values and remove the row of all missing class labels. An attribute mean can be used to fill the values of a missing class. This step involves classifying the URL datasets into two groups; legitimate and phishing URLs. In this work, we used an equal number of legitimate and phishing URL datasets.

3.5 Model Training

Our models are compared to show accuracy rate and precision between using an ensemble system and a classification algorithm. We used 70% of training data and 30% of testing data are used to compare accuracy scores of the classification algorithms.

3.6 Evaluation of model

An ensemble model is used to improve accuracy and efficiency. The following model is included in the combination of models: KNearestNeighbor, Logistic Regression, Random Forest Classifier. The individual prediction of each model is recorded and compared with the prediction value of the combined model.

3.7 Feature Extraction

Feature extraction was performed on the dataset to extract important features of the URL. Resampling and subsampling methods were performed on the dataset to create different subsamples of the original datasets. These subsamples allow the classification algorithm to diversify its' training and testing data. Resampling is essential for decision trees and neural networks because they aren't easily affected by small changes in the dataset.(Matteo Re & Giorgio Valentini, 2012). We have Lexical Features, Domain-Based Features, HTML and JavaScript-based Features extracted from the dataset. The target values of the datasets are in zeros(0) and ones(1). Where 0 represents a phishing URL and 1 represents a legitimate URL. The generated dataset is passed through a traditional classification algorithm which is then passed to an ensemble learning model. The feature selection model processes the initial data and stores the extracted features in an array. In malicious URL Detection Feature Representation a raw URL is often transformed to an appropriate feature vector u → x, before training a prediction model, so that it can be trained by traditional machine learning methods. Because classification models depend on the assumption that feature distributions for malicious and benign URLs are different, this feature representation must be carefully chosen.

The existence of a URL in a blacklist is used as a feature by the blacklist database, as it might be a powerful indicator that a URL is phishing. The length of the URL, the number of special characters, the types of words that exist in the URL string, the alphanumeric distribution of characters, and other lexical features are all considered. Host-based features are those generated from the URL's host-name attributes, such as IP address, WHOIS information, geographical location, and so on. Content features, such as HTML and JavaScript, necessitate directly visiting and downloading the content hosted by the URL to get information. Content features, such as HTML and JavaScript, necessitate directly visiting and downloading the content hosted by the URL to get information. The context and popularity characteristics refer to where the URLs have been posted on social media, as well as their ranking and popularity scores. Many studies have applied a combination of some of these characteristics, which was often chosen by expert domain knowledge.

3.8 URL Based Features:

3.8.1. Lexical Features

1. Internet Protocol (IP) Address: In a study carried out by Aburrous, the author explains that the presence of IP address or hexadecimal characters in the domain of the URL instead of using the domain name is associated with 46.66% of phishing URLs (Aburrous, et. Al,2010).

When an IP address is present in a URL it is regarded as a potential phishing site. The value assigned to this characteristic is 1 (phishing) if the domain section of the URL contains an IP address, or 0 otherwise (legitimate).

2. Presence of @ symbol in URL: When a URL contains '@', it causes the browser to disregard all previous characters before the symbol and focuses on the real address after the '@'. In Aburrous, he explains that most victims usually do not read the complete URL, therefore attackers are often utilizing this as a means to classify the website by the occurrence of '@' has been found in 20% of the malicious website (Aburrous, et. Al, 2010).

3. Presence of HTTPS in the middle of URL: The attackers add "HTTPS" to the domain part of the URL to trick users. We check if the domain part of a URL contains "HTTP/HTTPS", it is then labeled as 1 (phishing) or else 0 (legitimate).

4. Length of URL: The previous research shows that Phishers can disguise the suspicious element of a URL in the address bar by making use of long URLs. The malicious link is concealed within a lengthy URL. Any URL with a character length of more than 54 characters is labeled as 1 (phishing) or 0 (legitimate). In the study done by Aburous, the author has found that 73.33% of the malicious URLs have an abnormally long URL.

5. Redirect Request: A redirect request is commonly indicated by the presence of '//' in the URL website. "http://www.legitimate.com//http://www.phishing.com" is an example of this. The redirect request is placed after the double slash in this case. This function is useful for detecting phishing websites. After the position of the "//" in the URL is calculated, We discovered that if the URL begins with "HTTP," the "//" should be placed in the sixth position. If the URL uses "HTTPS," however, the "//" should occur in the seventh place. The value assigned to this characteristic is 1 (phishing) if the "//" appears anywhere in the URL other than after the protocol, or 0 if it does not (legitimate).

6. Prefix or Suffix "-" in Domain: Most legitimate websites do not use the "-" symbol it is usually associated with phishing websites. The attackers use this symbol to make their victims feel as do they are dealing with a legitimate website. If any URL in our dataset contains '-' this symbol in the domain part of the URL, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

7. Using URL Shortening Services "TinyURL": In Short URLs have become ubiquitous and popular in social networking. The author of "The risks associated with URL shortening services" highlights that URL shortening allows leaking of information and exposes the website to high-security risk by potentially malware attacks. (Thomas Hendrickson, 2016). HTTP Redirect is used on the Domain name as a URL Shortening service. Which helps to link it to the long URL. Most URLs shortening services contain malware. If the URL is using Shortening Services, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).

8. Delimiter Characters: The URLs International Journal of Advanced Science and Technology marks the presence of multiple delimiter characters such as "#, ~, _, %, &" is regarded as one of the characteristics associated with suspicious URL.

9. Dot count: The number of dot symbols in the URL's domain can be used as a flag to detect suspected phishing sites. A phishing website's domain section usually has more dots than a reputable website's domain section. The TLD and protocol parts of the URL are removed when calculating the number of dots present.

10. Checking for Prefix or Suffix: The dash '-' isn't normally included in a URL. It is added by phishers to give the illusion that the website is legitimate. When the '-' symbol is missing, a zero is assigned, and when the symbol is found in the URL, a one is assigned.

11. Having IP Address: Some URLs have IP addresses instead of the domain name. We check for the presence of IP which would mean someone is trying to steal information. If the domain part is an IP address instead of a domain, it is given the value 0(phishing).

3.8.2 Domain-Based Features:

1. Domain Age: In comparison to legal websites, which are normally online for a long period, phishing websites have a relatively brief lifespan. This information obtained from the WHOIS database is used be used to ascertain the domain's age, which is an important indicator for distinguishing between phishing and non-phishing websites. The majority of phishing websites are only active for a short time. For this project, the minimum age of a legal domain is deemed to be 12 months. Age is simply the difference between the time of creation and the time of expiry. If the domain is older than 12 months, the value of this feature is 0 (phishing), otherwise, it is 1(legitimate).

2. Website Validity: The Google Whois API can be used to determine whether or not a website is still functioning. The majority of phishing sites have a brief lifespan and are taken down once a suspicious activity is recognized. Validity is a key distinguishing factor between real and fraudulent websites.

3. Name Server Record: The WHOIS database, which is used to detect phishing sites, either does not recognize the claimed identity or has no records for the hostname. The value assigned to this feature is 0 (phishing) if the DNS record is empty or not discovered, or 1 otherwise (legitimate).

4. Web Traffic: This function determines how many visitors are accessing the website and how many pages they visit. Because phishing websites have such a short existence, the database is unlikely to recognize them. Legitimate websites can be found in the top 100,000 in the worst-case scenario. If the domain has no traffic, it is classified as phishing. Then with a ranking of less than 100,000, it is classified as legitimate.

3.8.3 HTML and JavaScript-based Features

This category contains a lot of features that can be extracted. The following were considered for this project out of all of them: IFrame Redirection and Website Forwarding.

1. IFrame Redirection:

The HTML tag IFrame allows you to insert another webpage into the one you're currently viewing. Phishers can utilize the "iframe" tag to make the frame invisible, i.e. without frame borders. In this situation, phishers use the "frame Border" attribute, which causes the browser to draw a visible barrier. An Iframe attribute is assigned as zero if the Iframe is empty or there is no response from the database.

2. Website Forwarding

There are a few distinctions between phishing and benign websites, one of which is that phishing websites frequently redirect to several destinations. A phishing URL redirects at least four times, but a trustworthy website just redirects once.

3.9 Tokenization of URL

In tokenization, a text is broken into units referred to as tokens. These tokens are words, characters, or sub-words derived from the URL string often referred to as n-gram characters tokenization. As tokens are the building blocks of Natural Language, the most common way of processing the raw text happens at the token level. The tokens derived are then used to prepare a set of unique tokens in the corpus called vocabulary. Initially, URLs are tokenized to form components using provided generic delimiters. The components of the URL are tokenized by using website delimiters. Each vocabulary is tested as a feature using Count Vectorizer and TD-IDF approaches used in Natural Language Processing (NLP). (Lovčí & Švec, n.d.)

CHAPTER FOUR

RESULTS AND DISCUSSION

The traditional ML classifiers and the ensemble learners used in this research were trained on the two datasets collected. The performance of the models resulting from the training were analyzed. As described in the methodology, four different approaches were used for feature selection. The results gotten from using tokenization, lexical features, domain name system features, and HTML features were compared to identify the best approach in phishing URL detection using performance measures such as Accuracy and Precision. The dataset in our study was split into training and testing set in the ratio 70:30. A balanced dataset with equal instances of malicious and legitimate URLs was used. The training set with the extracted features was given as input to different algorithms. It was observed that the domain-based features performed badly and produced a low accuracy score. This shows that this feature isn't very efficient in phishing detection.

4.1. Evaluation Criteria

To compare the performance of the different feature selection techniques and the models used, performance measures such as accuracy, precision, F1-score, recall values gotten from the confusion matrix were employed. An N x N matrix (confusion matrix) is used to evaluate the performance of a classification model, where N is the number of target classes. The matrix compares the actual goal values to the machine learning model's predictions. This provides us with a comprehensive picture of how well our classification model is working and the types of errors it makes. The confusion matrix consists of the parts listed below:

- **True Negatives (TN) -** These are the true negative forecasted values.
- **True Positive (TP) -** These are the classification's actual anticipated true values. The URLs that predicted benign were found to be benign.
- **False Positive (FP)–** The expected value was incorrectly predicted. Although the actual number was negative, the model projected that it would be positive.
- **False Negative (FN)-**The expected value was incorrectly predicted. Although the actual number was positive, the model predicted that it would be negative.

4.1.1 Accuracy

Accuracy sums up how well a model performs across all classes. The ratio between the number of right predictions and the total number of predictions is used to compute it.

Equation 1: Accuracy = (TP + TN) / (TP + TN + FP +FN)

4.1.2 Precision

Precision deals with the positive values of the matrix. The ratio of the number of correctly identified positive samples to the total number of predicted positive (either correctly or incorrectly predicted).

Equation 2: Precision = TP / (TP + FP)

4.1.3 Recall

Sensitivity is another term for recall. The recall is determined by dividing the total number of positive samples by the number of positive samples accurately categorized as positive. It converts the actual labels' result to a true positive value. A good classifier ideally has recall value of 1, which happens when the numerator and denominator are equal. Recall values decreases when FN increases (also increasing value of denominator) and becomes greater than the numerator.

Equation 3: Recall = TP / (TP + FN)

4.1.4 F1 score

The F1-score is a harmonic mean of Precision and Recall, and it reaches its maximum value when Precision and Recall are equal. Both false positive and false negative results are considered in the F1 score. F1 is at its best when it is 1, and at its worst when it is 0. It indicates how accurate the classifier is.

Equation 4: F1 score = 2 * (Recall * Precision) / (Recall + Precision)

4.2. Results

The experiments carried out showed improved accuracy score when ensemble learners were used as compared to the traditional classification methods. The dataset in our study was split into training and testing set in the ratio 70:30. Balanced dataset with equal instances of malicious and legitimate URLs were used. The training set with the extracted features was given as input to different algorithms. We observed that the domain-based features performed badly and produced a low accuracy score. This shows that this feature isn't very efficient in phishing detection.

Table 4.1**:** Result of traditional classifiers using with lexical features for dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.820	0.74	0.84	0.96
Logistic Regression	0.819	0.73	0.83	0.96
KNN	0.79	0.75	0.80	0.85

Table 4.2**:** Result of ensemble learners using lexical features for dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.814	0.74	0.84	0.96
Bagging	0.822	0.74	0.84	0.96
Gradient Boosting	0.805	0.74	0.83	0.95
Random Forest	0.824	0.75	0.84	0.96
Stacking	0.840	0.75	0.84	0.96

Table 4.3***:*** Result of traditional classifiers using lexical features for dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.849	0.85	0.84	0.83
Logistic Regression	0.691	0.63	0.72	0.85
KNN	0.838	0.78	0.84	0.91

Table 4.4**:** Result of ensemble learners using lexical features for dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.851	0.86	0.84	0.83
Bagging	0.848	0.82	0.85	0.88
Gradient Boosting	0.851	0.86	0.84	0.83
Random Forest	0.851	0.86	0.84	0.83
Stacking	0.853	0.85	0.84	0.84

4.2.1 Discussion of Results

Table 4.1 shows the accuracy score of test data on traditional classification algorithm. The decision tree classifier gives the highest accuracy score amongst the other three traditional classifiers with an accuracy rate of 82%. Our experiment shows 81.9% and 79% respectively for both logistic regression and KNN when testing the accuracy of the dataset with lexical features extracted from the URL.

Table 4.2 shows that stacking classifier has the highest accuracy score amongst other ensemble learners with an accuracy score of 84%. From the result displayed above, the gradient boosting classifier has the least accuracy score amongst the ensemble learners used with an accuracy score of 80.5%.

Table 4.3 and 4.4 shows the results across the different classifiers when testing was done on dataset 2. The traditional classifier with the highest accuracy score was the decision tree with a score of 84.9% while the Logistic Regression was the lowest with 69.1%. The stacking classifier gave the best results amongst all classifiers with an accuracy score of 85.3%.

Our experiment also checks the accuracy score of ensemble learning systems. These techniques show increased accuracy scores amongst some ensemble learners. Although some ensemble learners had improved accuracy scores, the result shows that stacking has better accuracy. The stacking model uses Random Forest and KNN as level one classifiers and Logistic Regression is used as the meta-classifier. In Table 4.5 to 4.8 below we display the results gotten from the both datasets when lexical, DNS and HTML features extracted from the URL are combined.

Table 4.5**:** Result of traditional classifiers using with all three features for dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.883	0.85	0.87	0.89
Logistic Regression	0.883	0.82	0.89	0.98
KNN	0.90	0.86	0.91	0.97

Table 4.6***:*** Result of ensemble learners using all three features for dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.883	0.85	0.89	0.94
Bagging	0.867	0.79	0.87	0.98
Gradient Boosting	0.883	0.85	0.89	0.94
Random Forest	0.89	0.84	0.91	1.00
Stacking	0.900	0.86	0.91	0.97

Table 4.7***:*** Result of traditional classifiers using all three features for dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.916	0.93	0.92	0.91
Logistic Regression	0.916	0.93	0.92	0.91
KNN	0.92	0.93	0.92	0.91

Table 4.8**:** Result of ensemble learners using all three features for dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.915	0.93	0.92	0.91
Bagging	0.915	0.93	0.91	0.90
Gradient Boosting	0.90	0.92	0.91	0.90
Random Forest	0.916	0.93	0.92	0.91
Stacking	0.925	0.93	0.92	0.91

4.2.2 Discussion of Results

Tables 4.9 to 4.12 show the accuracy of different classifiers after performing URL tokenization using both traditional and ensemble learning classifiers. The tokenization feature gave better results across the different classifiers used in this project. For dataset 1, the stacking classifier has highest accuracy score with 96% and lowest accuracy score of 86.7% was gotten from the decision tree classifier. In table 4.11 and 4.12, the result from dataset 2 is presented. The result shows the stacking classifier with 99.3% and decision tree with 93% as the highest and lowest accuracy score respectively. In developing an anti-phishing tool, the tokenization feature and stacking classifier seems to be the best fit for this model.

Table 4.9***:*** Result for traditional classification using tokenization on dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.867	0.93	0.87	0.82
Logistic Regression	0.935	0.93	0.94	0.96
KNN	0.955	0.95	0.95	0.96

Table 4.10**:** Result of URL tokenization for ensemble learners on dataset 1

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.95	0.95	0.95	0.95
Bagging	0.955	0.95	0.95	0.96
Gradient Boosting	0.938	0.95	0.95	0.95
Random Forest	0.958	0.96	0.96	0.96
Stacking	0.96	0.97	0.96	0.96

Table 4.11***:*** Result for traditional classification using tokenization on dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
Decision Tree	0.930	0.98	0.93	0.88
Logistic Regression	0.989	0.99	0.99	0.99
KNN	0.974	0.98	0.97	0.97

Table 4.12**:** Result of URL tokenization for ensemble learners on dataset 2

Model	**Accuracy (%)**	**Precision (%)**	**F1 score (%)**	**Recall (%)**
AdaBoost	0.986	0.99	0.99	0.99
Bagging	0.990	1.00	0.99	0.99
Gradient Boosting	0.986	0.99	0.98	0.98
Random Forest	0.970	0.99	0.97	0.95
Stacking	0.993	0.99	0.99	0.99

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This research discussed a way for detecting phishing attacks using feature extraction and URL tokenization. In this research, we used ensemble learning classifiers to improve attack detection accuracy. We combined feature extraction with stacking, bagging, and boosting. The same was done with the URL tokenization feature. After ensemble learners with tokenization performed better. This project contributed in the following way:

- Analyzed past works of literature relating to phishing detection and gave detailed knowledge of different phishing methods used by attackers.
- Proposed the use of tokenization using CountVectorizer and TFIDF with ensemble method to detect phishing attacks with ensemble learning techniques.
- Evaluated the accuracy and precision of the model used in this study.

There is a need to understand the origin of these attacks to enable researchers to provide a stronger and advanced anti-phishing solution. In conclusion, this would help in establishing anti-phishing measures that would thwart phishing attacks and help reduce the rate of these attacks.

5. 2 Future Work

In future works, we would suggest that ensemble learners can be combined with Deep Learning techniques to create hybrid models used for phishing detection. The mediums utilized for phishing attempts have shifted from traditional emails to social media-based phishing, as seen in new researches. There is a significant lag in phishing mechanisms and countermeasures in that sector. With phishing mediums shifting towards social media-based phishing, future researches should include this.

REFERNCES

A, N., H, N. D., Deepa Shenoy, P., & R, V. K. (2018). *ERCRTV: Ensemble of Random Committee and Random Tree for Efficient Anomaly Classification using Voting*.

Abdelhamid, N., Thabtah, F., & Abdel-Jaber, H. (2017). Phishing detection: A recent intelligent machine learning comparison based on models content and features. *2017 IEEE International Conference on Intelligence and Security Informatics: Security and Big Data, ISI 2017*, 72–77. https://doi.org/10.1109/ISI.2017.8004877

Adeyemi Adepetun. (2019, June 27). *Nigeria's exposure to phishing attacks rises as cybercrime cost hits $6 trillion | The Guardian Nigeria News - Nigeria and World News — Business — The Guardian Nigeria News – Nigeria and World News*. https://guardian.ng/business-services/nigerias-exposure-to-phishing-attacks-rises-as-cybercrime-cost-hits-6-trillion/

Alazab, M., & Fellow, S. (2020). *Malicious URL Detection using Deep Learning*. 1–9.

Alkhalil, Z., Hewage, C., Nawaf, L., & Khan, I. (2021). Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Frontiers in Computer Science*, *0*, 6. https://doi.org/10.3389/FCOMP.2021.563060

Almomani, A., Wan, T., Altaher, A., Manasrah, A., Almomani, E., Anbar, M., Alomari, E., & Ramadass, S. (2012). *Evolving Fuzzy Neural Network for Phishing Emails Detection National Advanced IPv6 Centre (NAV6), School of Computer Sciences , Faculty of Information Technology and Computer Sciences ,*. *8*(7), 1099–1107.

Alvaro, C. C. (2021, April 7). *AdaBoost from Scratch. Build your Python implementation of... | by Alvaro Corrales Cano | Towards Data Science*. https://towardsdatascience.com/adaboost-from-scratch-37a936da3d50

Anti-Phishing Working Group. (2020). Phishing Activity Trends Report 3rd Quarter 2020. *Apwg*, *November*, 1–12. https://docs.apwg.org/reports/apwg_trends_report_q3_2020.pdf

APWG. (2020). APWG Trends Report Q2 2020. *Phishing Activities Trends Report, Q2 2020*(August), 1–13. www.apwg.org

Bahnsen, A. C., Bohorquez, E. C., Villegas, S., Vargas, J., & Gonzalez, F. A. (2017). Classifying phishing URLs using recurrent neural networks. *ECrime Researchers Summit, ECrime*, 1–8. https://doi.org/10.1109/ECRIME.2017.7945048

Basit, A., Zafar, M., Javed, A. R., & Jalil, Z. (2020). A Novel Ensemble Machine Learning Method to Detect Phishing Attack. *Proceedings - 2020 23rd IEEE International Multi-Topic Conference, INMIC 2020*, *November*. https://doi.org/10.1109/INMIC50486.2020.9318210

Baykara, M., & Gürel, Z. Z. (2018). Detection of phishing attacks. *6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding*. https://doi.org/10.1109/ISDFS.2018.8355389

Blum, A., Wardman, B., Solorio, T., & Warner, G. (2010). Lexical feature based phishing URL detection using online learning. *Proceedings of the ACM Conference on Computer and Communications Security*, 54–60. https://doi.org/10.1145/1866423.1866434

Brown, G. (2011). Ensemble Learning. *Encyclopedia of Machine Learning*, 312–320. https://doi.org/10.1007/978-0-387-30164-8_252

Canali, D., Cova, M., Vigna, G., & Kruegel, C. (2011). Prophiler : A Fast Filter for the Large-Scale Detection of Malicious Web Pages Categories and Subject Descriptors. *Proc. of the International World Wide Web Conference (WWW)*.

Cao, J., Li, Q., Ji, Y., He, Y., & Guo, D. (2016). Detection of Forwarding-Based Malicious URLs in Online Social Networks. *International Journal of Parallel Programming*. https://doi.org/10.1007/s10766-014-0330-9

Chandrasekaran, M., Narayanan, K., & Upadhyaya, S. (2006). Phishing E-mail Detection Based on Structural Properties. *NYS Cyber Security Conference*.

Charbuty, B., & Abdulazeez, A. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends*, *2*(01), 20–28. https://doi.org/10.38094/jastt20165

Charu Makhijani. (2020, October 5). *Advanced Ensemble Learning Techniques | by Charu Makhijani | Towards Data Science*. https://towardsdatascience.com/advanced-ensemble-learning-techniques-bf755e38cbfb

Christou, O., Pitropakis, N., Papadopoulos, P., McKeown, S., & Buchanan, W. J. (2020). Phishing URL detection through top-level domain analysis: A descriptive approach. *ICISSP 2020 - Proceedings of the 6th International Conference on Information Systems Security and Privacy*, *March*, 289–298. https://doi.org/10.5220/0008902202890298

Curtis, S. R., Rajivan, P., Jones, D. N., & Gonzalez, C. (2018). Phishing attempts among the dark triad: Patterns of attack and vulnerability. *Computers in Human Behavior*, *87*, 174–182. https://doi.org/10.1016/J.CHB.2018.05.037

Evan Lutins. (2017, August 2). *Ensemble Methods in Machine Learning: What are They and Why Use Them? | by Evan Lutins | Towards Data Science*. https://towardsdatascience.com/ensemble-methods-in-machine-learning-what-are-they-and-why-use-them-68ec3f9fef5f

Fahmida Y. Rashid. (2020, November 24). *8 types of phishing attacks and how to identify them | CSO Online*. https://www.csoonline.com/article/3234716/8-types-of-phishing-attacks-and-how-to-identify-them.html

Felegyhazi, M., Kreibich, C., & Paxson, V. (2010). On the potential of proactive domain blacklisting. *LEET 2010 - 3rd USENIX Workshop on Large-Scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More*.

FireEye. (2020). *Cyber Trendscape 2020*. https://www.fireeye.com/offers/rpt-cyber-trendscape.html

Han, W., Cao, Y., Bertino, E., & Yong, J. (2012). Using automated individual white-list to protect web digital identities. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2012.02.020

Hota, H. S., Shrivas, A. K., & Hota, R. (2018). An Ensemble Model for Detecting Phishing Attack with Proposed Remove-Replace Feature Selection Technique. *Procedia Computer Science*, *132*, 900–907. https://doi.org/10.1016/j.procs.2018.05.103

Hutchings, A., Clayton, R., & Anderson, R. (2016). Taking down websites to prevent crime. *ECrime Researchers Summit, ECrime*. https://doi.org/10.1109/ECRIME.2016.7487947

Jain, A. K., & Gupta, B. B. (2018). Towards detection of phishing websites on client-side using machine learning based approach. *Telecommunication Systems*, *68*(4), 687–700. https://doi.org/10.1007/S11235-017-0414-0

Jang-Jaccard, J., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences*. https://doi.org/10.1016/j.jcss.2014.02.005

Jason Browniee. (2021, April 19). *A Gentle Introduction to Ensemble Learning Algorithms*. https://machinelearningmastery.com/tour-of-ensemble-learning-algorithms/

Johnson, J. (2021). *• Internet users in the world 2021 | Statista*. https://www.statista.com/statistics/617136/digital-population-worldwide/

Joshi, A., & Pattanshetti, P. T. R. (2019). Phishing Attack Detection using Feature Selection Techniques. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3418542

Kang, S. (2021). K-nearest neighbor learning with graph neural networks. *Mathematics*, *9*(8). https://doi.org/10.3390/math9080830

Kolari, P., Java, A., Finin, T., Oates, T., & Joshi, A. (n.d.). *Detecting Spam Blogs: A Machine Learning Approach **. Retrieved July 20, 2021, from www.aaai.org

KUALA LUMPUR. (2020, January 9). *91% of all cyber attacks begin with a phishing email to an unexpected victim | Deloitte Malaysia | Risk Advisory | Press releases*. https://www2.deloitte.com/my/en/pages/risk/articles/91-percent-of-all-cyber-attacks-begin-with-a-phishing-email-to-an-unexpected-victim.html

Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). *URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection*. *i*. http://arxiv.org/abs/1802.03162

Lim, W. H., Liew, W. F., Lum, C. Y., & Lee, S. F. (2020). *Phishing Security : Attack , Detection , and Prevention Mechanisms*.

Lovčí, M., & Švec, I. J. (n.d.). *Automatic classification of malicious URLs*.

Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009a). Beyond blacklists: Learning to detect malicious web sites from suspicious URLs. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. https://doi.org/10.1145/1557019.1557153

Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009b). Identifying suspicious URLs: An application of large-scale online learning. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/1553374.1553462

MADHURI, M., YESESWINI, K., & SAGAR, U. V. (2013). INTELLIGENT PHISHING WEBSITE DETECTION AND PREVENTION SYSTEM BY USING LINK GUARD ALGORITHM. *International Journal of Communication Networks and Security*. https://doi.org/10.47893/ijcns.2013.1083

Mao, J., Bian, J., Tian, W., Zhu, S., Wei, T., Li, A., & Liang, Z. (2019). Phishing page detection via learning classifiers from page layout feature. *EURASIP Journal on Wireless Communications and Networking*, *2019*, 43. https://doi.org/10.1186/s13638-019-1361-0

Matteo Re, & Giorgio Valentini. (2012). *(PDF) Ensemble methods: A review*. https://www.researchgate.net/publication/230867318_Ensemble_methods_A_review

MAURICE OGBONNAYA. (2020, October 19). *Cybercrime in Nigeria demands public-private action - ISS Africa*. https://issafrica.org/iss-today/cybercrime-in-nigeria-demands-public-private-action

Naresh, U. (2013). Intelligent Phishing Website Detection and Prevention System by Using Link Guard Algorithm. *IOSR Journal of Computer Engineering*, *14*(3), 28–36. https://doi.org/10.9790/0661-1432836

Ollmann, G. (2008). The Phishing Guide: Understanding and Prevent Phishing Attacks. *Security*.

Pandey, A., Gill, N., Sai Prasad Nadendla, K., & Thaseen, I. S. (2020). Identification of Phishing Attack in Websites Using Random Forest-SVM Hybrid Model. *Advances in Intelligent Systems and Computing*, *941*, 120–128. https://doi.org/10.1007/978-3-030-16660-1_12

Park, H. A. (2013). An introduction to logistic regression: From basic concepts to interpretation with particular attention to nursing domain. *Journal of Korean Academy of Nursing*, *43*(2), 154–164. https://doi.org/10.4040/jkan.2013.43.2.154

Parsons, K., Butavicius, M., Delfabbro, P., & Lillie, M. (2019). Predicting susceptibility to social influence in phishing emails. *International Journal of Human-Computer Studies*, *128*, 17–26. https://doi.org/10.1016/J.IJHCS.2019.02.007

Patil, V., Thakkar, P., Shah, C., Bhat, T., & Godse, S. P. (2018). Detection and Prevention of Phishing Websites Using Machine Learning Approach. *Proceedings - 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018*. https://doi.org/10.1109/ICCUBEA.2018.8697412

Phishing Activity Trends Report 1st Quarter 2021. (2021, June 8). https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf

Phishing Activity Trends Report 3rd quarter 2019. (2019, November 4). https://docs.apwg.org/reports/apwg_trends_report_q3_2019.pdf

Polikar, R. (2009). Ensemble learning. *Scholarpedia*, *4*(1), 2776. https://doi.org/10.4249/SCHOLARPEDIA.2776

Preethi, V., & Velmayil, G. (2016). Automated Phishing Website Detection Using URL Features and Machine Learning Technique. *International Journal of Engineering and Techniques*, *2*(5), 107–115. http://www.ijetjournal.org

Rasool, A., & Jalil, Z. (2020). A review of web browser forensic analysis tools and techniques. *Researchpedia Journal of Computing*, *1*(1), 15–21.

Rokach, L., & Maimon, O. (2006). Decision Trees. *Data Mining and Knowledge Discovery Handbook*, 165–192. https://doi.org/10.1007/0-387-25465-X_9

Sahingoz, O., Buber, E., Demir, Ö., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, *117*, 345–357.

https://doi.org/10.1016/J.ESWA.2018.09.029

Sahoo, D., Liu, C., & Hoi, S. C. H. (2019). *Malicious URL Detection using Machine Learning: A Survey*. https://doi.org/10.1145/nnnnnnn.nnnnnnn

SERIANU. (2017). *Nigeria Cyber Security Report 2017 Demystifying Africa`s Cyber Security Poverty Line*. 1–80. https://www.serianu.com/downloads/NigeriaCyberSecurityReport2017.pdf

Srinivasan, S., Vinayakumar, R., Arunachalam, A., Alazab, M., & Soman, K. (2021). DURLD: Malicious URL Detection Using Deep Learning-Based Character Level Representations. *Malware Analysis Using Artificial Intelligence and Deep Learning*, 535–554. https://doi.org/10.1007/978-3-030-62582-5_21

Steinki, O., & Mohammad, Z. (2015). Introduction to Ensemble Learning. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.2634092

Subasi, A., Molah, E., Almkallawi, F., & Chaudhery, T. J. (2017). Intelligent phishing website detection using random forest classifier. *2017 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2017*, *2018-January*, 1–5. https://doi.org/10.1109/ICECTA.2017.8252051

Thabtah, F., & Abdelhamid, N. (2016). Deriving correlated sets of website features for phishing detection: A computational intelligence approach. *Journal of Information and Knowledge Management*, *15*(4). https://doi.org/10.1142/S0219649216500428

Tyagi, I., Shad, J., Sharma, S., Gaur, S., & Kaur, G. (2018). A Novel Machine Learning Approach to Detect Phishing Websites. *2018 5th International Conference on Signal Processing and Integrated Networks, SPIN 2018*, 425–430. https://doi.org/10.1109/SPIN.2018.8474040

Ubing, A. A., Kamilia, S., Jasmi, B., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Phishing Website Detection: An Improved Accuracy through Feature Selection and Ensemble Learning. *IJACSA) International Journal of Advanced Computer Science and Applications*, *10*(1). www.ijacsa.thesai.org

Verizon: 2019 Data Breach Investigations Report. (2019). *Computer Fraud & Security*, *2019*(6), 4. https://doi.org/10.1016/s1361-3723(19)30060-0

Vinayakumar, R., Soman, K. P., Prabaharan Poornachandran, Akarsh, S., & Elhoseny, M. (2019). Deep learning framework for cyber threat situational awareness based on email and URL data analysis. *Advanced Sciences and Technologies for Security Applications*, 87–124. https://doi.org/10.1007/978-3-030-16837-7_6

Wang, L. (2019). Research and Implementation of Machine Learning Classifier Based on KNN. *IOP Conference Series: Materials Science and Engineering*, *677*(5). https://doi.org/10.1088/1757-899X/677/5/052038

Warburton, D., & F5 Labs. (2020). *2020 Phishing and Fraud Report Phishing During A Pandemic*. 46.

Wenyin, L., Fang, N., Quan, X., Qiu, B., & Liu, G. (2010). Discovering phishing target based on semantic link network. *Future Generation Computer Systems*. https://doi.org/10.1016/j.future.2009.07.012

Widup, S. (2018). 2018 Data Breach Investigations Report: the year of ransomware. *Computer Fraud & Security*, *2018*(5), 4. https://doi.org/10.1016/s1361-3723(18)30040-x

Williams, E. J., Hinds, J., & Joinson, A. N. (2018). Exploring susceptibility to phishing in the workplace. *International Journal of Human Computer Studies*, *120*, 1–13. https://doi.org/10.1016/J.IJHCS.2018.06.004

Xu Ying. (2014, May). *(PDF) Ensemble Learning*. https://www.researchgate.net/publication/262369664_Ensemble_Learning

Yao, W., Ding, Y., & Li, X. (2019). LogoPhish: A new two-dimensional code phishing attack detection method. *Proceedings - 16th IEEE International Symposium on Parallel and Distributed Processing with Applications, 17th IEEE International Conference on Ubiquitous Computing and Communications, 8th IEEE International Conference on Big Data and Cloud Computing, 11t*. https://doi.org/10.1109/BDCloud.2018.00045