PHISHING EMAIL DETECTION BY USING MACHINE LEARNING TECHNIQUES

A Thesis Presented

by

Tariku Yabshe Chiksa

to

The Faculty of Informatics

of

St. Mary's University

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

in

Computer Science

January, 2022

ACCEPTANCE

Phishing Email Detection by Using Machine Learning Techniques

By

Tariku Yabshe Chiksa

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

Internal Examiner
Henock Mulugeta (PhD)

External Examiner
Minale Ashagrie (PhD)

Dean, Faculty of Informatics

January 2022

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

_____

Tariku Yabshe Chiksa

_____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

_____

Alemebante Mulu Kumlign (PhD)

_____

Signature

Addis Ababa

Ethiopia

January 2022

ABSTRACT

Electronic mail (e-mail) is one of the most popular methods for online communication and data transmission over the web because of its rapid and simple dissemination of data, cheap distribution cost, and permanence. Despite its advantages, e-mail has several drawbacks. The most common of these are phishing and spam emails. While both phishing emails and spam can jam your inbox, only phishing is specifically designed to steal login passwords and other important information. Spam is a marketing strategy that involves sending unsolicited emails to large groups of people in order to promote products and services. A phishing email is a genuine-looking email that is intended to fool users into thinking it is a legitimate email and then either expose sensitive information or download malicious software by clicking on malicious links contained in the email's body. Phishing is more harmful in this aspect because it has caused tremendous financial loss to domain users. Therefore, there is an urgent need for phishing email detection with high accuracy. Banking information, credit reports, login data, and other sensitive and personal information are frequently transmitted over email. This makes them valuable to cyber criminals, who can exploit the knowledge for their own gain. In this paper, we proposed a phishing email detection algorithm based on Naïve Base algorithms and a Support Vector Machine classifier. We extracted email features by analyzing the email header structure, email body, email Uniform Resource Locator information, and email script function features. The aim of this paper: (i) Investigate the challenge of the existing email filtration method for the purpose of minimizing the gap caused by junk mail filtration; (ii) Provide an effective and improved way of phishing email classification method by using machine learning approaches; (iii) Prevent users from opening the malicious link and responding to the attacker; and (iv) Prevent phishing emails from being sent to the intended recipient. Experiments are performed on a dataset consisting of a total of 5229, which includes 4115 legitimate emails and 1114 phishing emails. The proposed technique performed well in detecting phishing emails. According to our findings, Support Vector Machines outperformed the Naive Base in detecting phishing emails, with accuracy rates of 98.76% and 97.51%, respectively.

Keywords: Phishing, Classifier, Bit squat, Malware.

ACKNOWLEDGMENTS

First off, I would like to thank God and many others for assisting me in completing the research work on time and successfully. Secondly, I would like to express my gratitude to my adviser, Dr. Alemebante Mulu, for his readiness to assist me and provide constructive suggestions and remarks from the beginning to the end of this research paper. Thirdly, I'd like to thank Dr. Getahun Semeon for his willingness to help me and provide constructive advice from scratch.

In the pursuit of my undertaking, no one has been more essential to me than my family members. I would want to express my gratitude to my family for their unwavering support and encouragement during my studies and research.

Finally, my thanks go to all the people who have supported me in completing the research work, directly or indirectly. I am grateful to everyone who helped me finish my research paper.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

Abbreviation	Description
APT	Advanced Persistent Threats
ASCII	American Standard Code for Information Interchange
CSV	Comma-Separated Values
ECML	Electronic Commerce Modeling Language
HTTPS	Hypertext Transfer Protocol Secure
IMAP	Internet Message Access Protocol
KNN	K-Nearest Neighbor
NB	Naïve Bayes
NLP	Natural Language Processing
POP	Post Office Protocol
SMTP	Simple Mail Transfer Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
SVM/SVC	Support Vector Machine/Support Vector Classifier
URLs	Uniform Resource Locator

Definition of Terms

Phishing – The practice of sending emails purporting to be from legitimate source in order to lure individuals to reveal their personal information such as IDs, passwords, and credit card numbers.

Algorithm – An algorithm is a process, or a step-by-step procedure aimed at solving a particular problem.

Classifier – A set of rules, methods or statistical procedure that identifies to which category an observation belongs based on already trained set of data whose category is known.

Bit squat – A registered domain name with one bit difference on its IP address with reference to another domain.

Malware – An umbrella term used to refer to a variety of malicious software.

Spam – Unsolicited or undesired emails.

Tokenize – Splitting a string into desired constituent part.

Chapter One

1.1. Introduction

Electronic mail, in short Email is one of the most widely used features of the Internet, along with the web. It allows you to send and receive messages to and from anyone with an email address, anywhere in the world. Email uses multiple protocols within the TCP/IP suite. Email has been an extremely important medium of communication for quite some time now, allowing almost instant reachability to any part of the world with internet connectivity [1].

A phishing email on the other hand is a kind of spam email that is sent out specifically, to trick people into sharing their personal details like, password, debit/credit card details, bank account details, etc. Sharing their details through such phishing emails can lead to cases of financial fraud through identity theft. At times, these emails are specifically targeted to dig out personal information about their colleague or company. Such emails are referred to as spear-phishing targeted emails [2].

In the context of email filtering, various unsolicited mail filtering techniques are implemented, such as knowledge-based techniques, clustering techniques, learning-based techniques, heuristic processes, and so on, but the problem is that they are unable to control bypass attacks [3]. This thesis overview current spam detection mechanisms and identifies the challenges of existing email filtering gaps and our study identifies phishing email link and check for any malicious attachments to provide an effective machine learning mechanism to detect, filter and classified phishing email within organization by using Support Vector Machine and Naïve Bayes.

To increase the accuracy rate of phishing email detection system and to control bypass to study issues of current spam detection method is important because of the attacker swift adoption of new techniques content of phishing link and malicious attachments are different from time to time. The goal of our research was to develop an effective phishing detection algorithm that would detect, prevent, and protect users from responding to phishing emails that contained malicious links and attachments, thereby aiding targeted users in reducing the number of phishing email attacks. Additionally, this research identified numerous phishing types and

validated the algorithm's accuracy. The end product is valuable for company side and staff side who seek to be secured from malicious email link and malicious email attacks.

1.2. Motivation

The development of spam filters to continue to be an active research field for academician and industry practitioners researching machine learning techniques for effective spam filtering [4]. The motivation behind this research initiative is to address a gap (unable to control bypass) that has risen over time in the field of spam email detection. The current solutions features are mostly lagging behind the innovativeness the spammers are constantly bringing in, which heavily justifies the emergence of machine learning based anti-spam propositions. In our research, we looked at current mail filtering issues and developed a filtration algorithm to close the gap. This research work critically evaluates number of such reasonably recent solutions and provides insights into ways upon which further improvement can be obtained.

1.3. Statement of Problem

Implementing email filtering (i.e., phishing vs ham) is extremely important for any organization. Email filtering not only keeps spam out of inboxes, but it also improves the quality of life of business emails by ensuring that they function efficiently and are only used for their intended purpose [4]. Phishing filtering is fundamentally an anti-malware tool, as many email attacks try to deceive users into clicking on dangerous attachments or URLs, requesting sensitive information, and so on. Phishing causes several problems either directly or indirectly to the email system [5] . Among them Network conjunction, misuse of storage space and computational resources, loss of work productivity and annoyance to users, legal issues as a result of pornographic advertisements and other objectionable material, financial losses through phishing and other related attacks, spread of viruses, worms and Trojan Horses, Denial of Services and Directory Harvesting attacks.

We have reviewed almost 24 works done by previous researchers for detail section 2.10. As result most of them are effective method of detecting spam but nowadays spammers can easily bypass all these spam filtering applications easily [4] and also, most of them do not specifically for phishing email instead they attempt to make a distinction between spam emails and ham

emails the latter also going by the name of ham emails [6]. Phishing attacks are potentially more harmful in comparison to spam mails. Because they are designed to look legitimate but have the intention of hurting, manipulating, or tricking people into doing something they normally would not or should not. Due to this we focus on improving the effectiveness of detecting phishing emails in specific.

The rapid growths up of spammer phishing techniques unsolicited mail filtering is difficulty for researcher. Due to this phishing detection and spam email filtration mechanism is critical area of researcher. so, implementing, modeling, and designing spam email detection mechanisms time to time and change the mechanisms according to phishing feature. The other thing is still now spammers can easily bypass all the spam filtering applications easily [7] for detail section 2.8.

During the fourth quarter of 2021, 22.5 percent of phishing attacks worldwide were directed towards financial institutions [8] one of among financial institutions are Internet Service providers. Internet is widely used nowadays and the increase in the phishing spam emails causes time and money loss with disrupted users. Phishing emails cause a waste of time and money for the individuals having approximately a hundred phishing email each day. Phishing detection is an important topic for saving people from unsolicited commercial emails. Therefore, there exists a need for research on better ways of detecting phishing emails and alert the user there of or prevent the emails from reaching the users. The main aim is to provide an effective, improved way of phishing email classification, filtering, and preventing the user from opening and responding to the phishing emails based on the challenge of current detection methods.

1.4. Research Questions

1. Why existing mail filtration methods easily bypass by attacker?
2. What are the various types of phishing email feature?
3. How to implement effective detection algorithms for our problem?
4. How will the accuracy of the algorithm be validated?

1.5. Objective

1.5.1. General Objective

The general objective of this study is phishing email detection by using machine learning techniques.

1.5.2. Specific Objectives

To achieve the general objectives of the study, the following specific objectives are formulated:

- ✓ To acquire knowledge from existing mail filtration techniques and literature review.
- ✓ To investigate challenges of existing mail filtration methods to prevent bypass and to recommend solutions for existing challenges.
- ✓ To identify the various types of email phishing methods and to create corpus.
- ✓ To identify malicious links and malicious attachments that attackers used to bypass.
- ✓ To develop effective email filtering machine learning algorithm that captures filters malicious links and malicious content used by attackers in emails.
- ✓ To validate the accuracy of the algorithm.

1.6. Significant of The Study

The study will be creating an effective phishing mail detection algorithm that would detect, prevent, and protect staff from responding to phishing emails i.e., malicious links and malicious content, thus helping targeted staffs to reduce the number of phishing email attacks and this study will be identified numerous phishing types and validate the accuracy of the algorithm. The end product is valuable for company side and staff side who seek to be secured from malicious email link and malicious email content.

Our contribution is investigating challenges of existing mail filtration methods using machine learning and open research questions. While methods for malicious email detection from legitimate emails exist and achieves high accuracy, there are no solutions to classify spam and phishing emails within the malicious email flow. Therefore, in this paper we propose a solution, dedicated to classifying unwanted emails to spam and phishing email categories. The challenges of the machine learning algorithms in efficiently handling the menace of spam email were pointed out the other thing is relative studies of the machine learning techniques available in

literature was done. To address the problem technically we used hybrid method which is Support Vector Machine and Naïve Bayes algorithm.

1.7. Scope

The paper review existing machine learning mail filtration methods rather existing non-machine learning based approaches of email filtration and not include all existing mail filtration methods.

1.8. Limitation

The paper didn't include all existing mail filtration because of the time limitation. In addition to we investigated various challenges of existing mail filtration machine learning approaches that did not control bypass. however, we address few of them we will put section 5.2 for detail. Other challenges recommend for other researcher of academician to address the challenges.

1.9. Thesis Organization

The thesis is containing of five chapters coordinated as the follows:

- **Chapter Two**: Literature Review: this chapter provides literature review and background: It includes concept of email, component and architecture of email, email spam filtering architecture, spam email filtering process, overview of existed email filtration methods, Challenges of existing Email filtration methods, phishing mail techniques, various types of features used in spam email classification.
- **Chapter Three**: Methodology: this chapter provides an outline of the research methodology which used in this thesis. The architecture of Proposed Method, Overview of the software that used for the evaluation of the proposed method and the dataset were used in this research. Various phishing email feature details are described.
- **Chapter Four**: Implementation and Experiment: The implementation details of experiment and the results that were obtained for all the proposed scenarios and comparison of the results.
- **Chapter Five**: Conclusion, recommendation, and future work.

Chapter Two

2. Literature Review

This section provides an overview on some of the main studies conducted on data mining techniques and algorithms to detect phishing emails. There are only a few research efforts that focus entirely on tackling the problem of phishing attacks. Phishing e-mails are often related to spam and most of these techniques target spam control as a mechanism to prevent such identity theft scams. The primary difference is that the spam messages lack proper feature selection that appropriately demarcates spam from phishing messages.

Zhan et, al. [9] proposed a method to detect and filter phishing emails by employing Stochastic Learning-Based Weak Estimators (SLWE) in real life environment. SLWE approach was studied and implemented based on Naive Bayes classification for filtering phishing emails that are unpredictable in nature. They used two different datasets: 1,200 real benign emails and 600 real phishing emails. To evaluate the effectiveness of their proposal, they compared their captured results from SLWE approach with Maximum Likelihood Estimator (MLE). MLE is a popular and widely used estimation scheme. Their results revealed that SLWE-based Naive Bayes approach outperforms the MLE scheme regarding accuracy. However, their proposed method suffers from an enormous number of features, which can affect system performance, and unlimited training, which can consume large amounts of storage [9].

Chandrasekaran depended on the distinctive structural features of the email to detect phishing emails. These features work in cooperation with the SVM to predict phishing emails and prevent them from originally reaching the user [10].

Lueg presented a brief survey to explore the gaps in whether information filtering and information retrieval technology can be applied to postulate Email spam detection in a logical, theoretically grounded manner, in order to facilitate the introduction of spam filtering technique that could be operational in an efficient way. However, the survey did not present the details of the Machine learning algorithms, the simulation tools, the publicly available datasets, and the architecture of the email spam environment. It also fails short of presenting the parameters used by previous research in evaluating other proposed techniques [11].

Wang reviewed the different techniques used to filter out unsolicited spam emails. The paper also to categorized email spams into different hierarchical folders, and automatically regulate the tasks needed to response to an email message. However, some of the limitations of the review article are that; machine learning techniques, email spam architecture, comparative analysis of previous algorithms and the simulation environment were all not covered [12].

The Random Tree classifier proved a 99.72% accuracy which means it works best to detect spam emails. In conclusion, the accuracy of email filters was enhanced incredibly when the algorithm with feature selection was applied into the entire process and that classifiers of tree shape are more efficient in detecting spam emails [13].

Marsono, M.N, et al. [14] They introduced equipment engineering of Naive Bayes inference motor for spam control utilizing two class email classification. That can order more 117 million features for every second given a stream of probabilities as information sources. This work can be reached out to examine proactive spam taking care of plans on accepting email servers and spam throttling on network gateways.

Y. Tang, S. Krasser, et al. [15] a framework that utilized the SVM for classification reason, such framework removes email sender conduct information in light of worldwide sending dispersion, investigate them and allot an estimation of trust to every IP address sending email message, the Experimental outcomes demonstrate that the SVM classifier is viable, precise and substantially speedier than the Random Forests (RF) Classifier.

Rathi et al. [16] suggested a method for determining the best classifier for email classification using data mining techniques. They compared the performance of numerous classifiers using "with feature selection algorithm" and "without feature selection algorithm" data mining techniques. They considered the specified algorithm for feature selection after picking the best feature selection method. They use a variety of algorithms to experiment with their data, including Nave Bayes, Bayes Net, Support vector machine, Function tree, J48, Random Forest, and Random Tree. There are 4601 occurrences and 58 attributes in the entire dataset. The maximum accuracy for the Random Tree method was 99.72 percent, while the lowest accuracy for the Nave Bayes algorithm was 78.94 percent.

DeBarr et al. [17] employ Random Forest algorithms to classify spam email, then apply active learning to refine the classification model. They took data from RFC 822 (Internet) email messages, separated each into two sections, then transformed each message to TF/IDF features. Select an initial collection of email messages to label as training examples using the Partitioning Around Medoids (PAM) algorithm and a clustering technique. They explore with Random Forest, Naive Bayes, SVM, and KNN after considering the cluster prototype messages for training. With 95.2 percent accuracy, the Random Forest algorithm is the best classifier. With 95.2 % accuracy, the best classifier.

Sahami et al. [18] proposed the use of features for junk email filtering and built the Bayesian classifier. Explicit highlights were phrases like "Free Money" and "!!!!" over ornamented accentuation marks. The accuracy of filters was improved by placing these additional highlights next to the trademark Email message material.

2.1. General Concept of Email

This section will present the concepts applicable to the work done in this thesis.it includes concept of email, component and architecture of email, email filtering, email treats, email spam filtering architecture, spam email filtering process, overview of existed email filtration methods, challenges of existing Email filtration methods, proposed solution, phishing mail techniques and various types of features used in spam email classification. Present how spam and phishing filter work and presentation of email security measures for detecting and preventing phishing emails, overview machine learning approaches to address the phishing email problem.

Electronic mail is means of communication a way to send and receive messages across the Internet. Email uses multiple protocols within the TCP/IP suite. Email has been an extremely important medium of communication for quite some time now, allowing almost instant reachability to any part of the world with internet connectivity [19]. Email can help you become more efficient, productive, and prepared for business. When it comes to business email, there are a few things to keep in mind: **Cheap** - regardless of distance or number of recipients, sending an email cost the same. **Quickly** - an email should reach its intended destination within minutes, if not hours. **Collaboration** entails speaking to a group of people at once. Many businesses and

organizations communicate and manage their correspondence using email applications such as Microsoft Outlook.

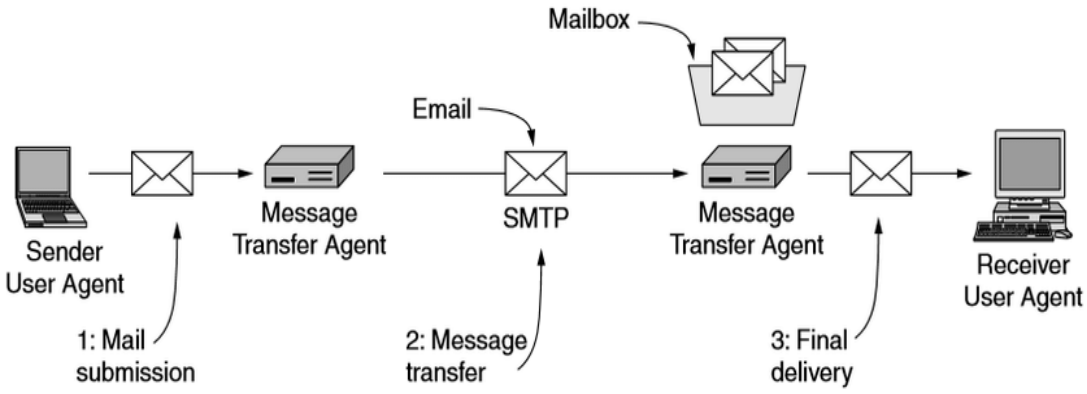

Figure 1 Email architecture [1]

The email system architecture is illustrated in Figure 1. It contains two sub systems: (i) the user agents are used to read, send, compose, replies to messages, display incoming messages, and arrange messages by filing, searching, and deleting them. Examples to most common user agents are Google Gmail, Microsoft Outlook, Mozilla, and Apple Mail. (ii) The message transfer agents are used to send messages from the source to the destination with the help of Simple Mail Transfer Protocol (SMTP). They are also known as mail servers [20].

2.2. Email structure and components

In this section we will discuss the main architectural components for an email infrastructure. These components are addresses, protocols, agents and message formats.

2.2.1. Mail Address

An email address is the most fundamental form online identity [21]. It lets you send and receive emails with anyone, create an account on various websites or apps, receive email newsletters from interesting sources, accept critical notifications, apply for jobs, etc. An email address identifies an email box to which messages are delivered. An email address, such as tariku.yabshe@gmail.com, is made up from a local-part, the symbol @, and a domain, which may be a domain name or an IP address enclosed in brackets [21].

2.2.2. Mail Protocol

There are three common protocols used to deliver email over the Internet: The Simple Mail Transfer Protocol (SMTP), the Post Office Protocol (POP), and the Internet Message Access Protocol (IMAP). All three use TCP, and the last two are used for accessing electronic mailboxes.

2.2.3. POP and IMAP

Traditionally, users accessed their mailboxes with a mail reader, which opened the mailbox as a local file. Later mechanisms were introduced to be able to access a mailbox stored on a server using a mail reader that accesses it over the network. Several versions of the Post Office Protocol [22]. It is mostly used to transfer complete messages from a server to a client to be read and reacted upon online or offline. Most of the time the message is deleted afterwards on the server. This mechanism has an obvious drawback once people started to access their mail from different clients all over the net. This way there was not a single consistent view of a user's mailboxes. This is where IMAP (Internet Message Access Protocol) comes to the rescue. The most recent specification is Internet Message Access Protocol – version4 [22]. The idea here is that the complete mail store resides and stays on the server and that clients access it in order to present it to the reader. Modifications, such as message status updates, can be applied to the messages stored on the server, and clients can use caching strategies for performance optimization and for enabling off-line access.

2.2.4. SMTP

SMTP was indeed a very simple protocol intended for mail transport. It lacked some of the more advanced features of the X.400 message service. In the original specification only a few commands were defined [22].

2.2.5. Email Component

There is a standard structure for emails. Email contents are primarily classified as header and the body. Email head contain: From, To, CC Bcc, subject and attachment.

From: is your email address, or the address sending the email. Usually this is already filled in with your address. **To:** is where you type the email address of the primary recipients. **Subject:** is a concise indication of the subject of your message. It is important to include a subject line because it will benefit the recipient by allowing them to see what your email is about before they

open it. It is especially helpful if you are sending email to someone who might not recognize their address. This will help indicate to them that the email is not spam, or junk email [22].

Add CC / Add BCC: To add secondary addresses to your email, click on these links and additional boxes will appear. **Add CC**: CC stands for "carbon copy." This allows you to "copy" a person on an email that you are sending to someone else. **Add BCC:** This is for "blind carbon copies." If you send a copy of an email to someone by putting their address here, the recipients in the **To:** and **CC:** boxes will not be able to see that person's address. **Attachments**: Some emails could be attached with files such as text, image, audio, video etc. These files are specified here. **Body**: The actual content is stored in this part. This will be in the format of text. This field could also include signatures or text generated automatically by the sender's email system. As we mentioned earlier, the contents of the emails can be varied according to the different email systems used by each user.

2.3. Email threats

The email threat landscape is continuously evolving. Every year, cybercriminals develop new ways of tricking and attacking their victims through email. The context, scenarios, and types of emails differ, but the main threats stay the same. The three main email threats are malicious attachments, malicious URLs, and social engineering. These threats lead to the result in data loss, stolen information, disruption of business, and monetary loss. Most phishing emails involve one or more of these three threats.

2.3.1. Malicious attachments

Malicious email attachments are known to contain malicious software (Malware), which can install viruses, trojan horses, spyware, bots, set up ransomware attacks, infect Office files through macros, or launch Advanced Persistent Threats (APT). Malware is designed to launch when an email attachment is opened. It can be disguised as documents, PDFs, voicemails, e-faxes, images, and other types of files that would seem to be trustworthy or exciting.

Symantec reported in 2019 a malicious email rate of 1 in 412, where 48% of all malicious attachments where Office files such as word and excel files. Verizon's Data Breach Investigation Report 2019, states that 94 % of all malwares was delivered through email [23].

2.3.2. Malicious URLs

A malicious Uniformed Resource Locator (URL) is a clickable link embedded within the body or attachment of an email [24]. It is created with the sole purpose of compromising the recipient of the email. Malicious URLs are often disguised in images, buttons, or text that do not match the intended use. Symantec reported in 2019 that in their collected data from 2018, a malicious URL was found in every 170 email URL [25].

Clicking on a malicious URL can download and execute malicious scripts or install malware. It can also be a web address that takes the target to a fake website. This to persuade them to unintentionally giving away sensitive information, such as usernames and passwords, or expose them to an insecure location capable of installing malware on their computer.

2.3.3. Social engineering

Social engineering used in emails involves a form of psychological manipulation, fooling otherwise unsuspecting email recipients [26]. Such manipulation tries to invoke urgency, fear, or interest in the victim through text in an email. Social engineering may lead the victim to click malicious links, open malicious files, or perform actions such as giving away sensitive information or transferring money to an illegitimate source. It can be tough to prevent such threats as it exploits human errors [26].

2.4. Email filtering

Email filtering is the processing of emails to rearrange it in accordance with some definite standards [4]. Mail filters are generally used to manage incoming mails, filter spam emails, detect and eliminate mails that contain any malicious codes such as virus, trojan or malware. Phishing attacks are typically perpetrated via emails. These emails usually contain social engineering messages (with specific phrases) that demand users to perform specific actions (such as clicking on a URL). Therefore, the content of these emails are useful features for phishing detection. Very few phishing email filters have been developed, in contrast to many existing email filters that have been developed for spam emails [27].

2.4.1. Email Spam Filtering Architecture

The aim of Spam filtering is to decrease the barest minimum the number of spontaneous emails. The workings of email are influence by some basic protocols which include the SMTP. Some of the widely used Mail User Agents (MUAs) are Mutt, Elm, Eudora, Microsoft Outlook, Pine, Mozilla Thunderbird, IBM notes, KMail and Balsa [28]. They are email clients that assists the user to read and compose emails. Spam filters can be deployed at strategic places in both clients and servers. Spam filters are deployed by many Internet Service Providers (ISPs) at every layer of the network, in front of email server or at mail relay where there is the presence of firewall [4]. The firewall is a network security system that monitors and manages the incoming and outgoing network traffic based on predetermined security rules. The email server serves as an incorporated anti-spam and anti-virus solution providing a comprehensive safety measure for email at the network perimeter [4]. Filters can be implemented in clients, where they can be mounted as add-ons in computers to serve as intermediary between some endpoint devices. Filters block unsolicited or suspicious emails that are a threat to the security of network from getting to the computer system. Also, at the email level, the user can have a customized spam filter that will block spam emails in accordance with some set conditions [29].

2.4.1.1. How Spam Email Filters Work

Several spam filtering methods have been utilized by Gmail, Outlook, and Yahoo to deliver only the legitimate emails to their users and filter out the illegitimate messages. Conversely, these filters also sometimes erroneously block authentic messages. It has been reported that about 20 percent of authorization-based emails usually fail to get to the inbox of the expected recipient [30]. The email providers have designed different mechanisms for use in email spam filter to curtail the dangers posed by phishing, email-borne malware, and ransomware to email users. The mechanisms are used to decide the risk level of each incoming email. Satisfactory spam limits, sender policy frameworks, whitelists and blacklists, and recipient verification tools are among such mechanisms [30]. This section discusses the operations of Gmail, Yahoo and Outlook emails anti-spam filters.

2.4.1.2. Gmail Spam Filter

Google data centers use of hundreds of rules to determine email classification whether an email is ham vs spam [4]. Every one of these rules depicts specific features of a spam and certain

statistical value is connected with it, depending on the likelihood that the feature is a spam. The weighted importance of each feature is then used to construct an equation. A test is conducted using the score against a sensitivity threshold decided by each user's spam filter. And consequently, it is classified as a legal or unsolicited email. Google is believed to classify emails using cutting-edge spam detection machine learning methods such as logistic regression and neural networks. Optical character recognition (OCR) is also used by Gmail to protect users from picture spam [31]. Gmail can also link hundreds of parameters to improve spam detection thanks to machine-learning algorithms built to aggregate and rank enormous collections of Google search results. Factors like as domain reputation, links in message headers, and others have changed the character of spam throughout time. Messages may end up in the spam bin as a result of these factors. Spam filtering is based on "filters" that are constantly updated as new tools, algorithms, and spam is discovered, as well as comments from Gmail users concerning potential spammers. Many spam filters use text filters to eliminate spammers' threats based on the senders and their history.

2.4.1.3. Yahoo Mail Filter Spam

Yahoo mail is the first free webmail providers in the world with over 320 million users [32]. The email provider has its own spam algorithms that it uses to detect spam messages. The basic methods used by Yahoo to detect spam messages include URL filtering, email content and spam complaints from users [33]. Unlike Gmail, Yahoo filter emails messages by domains and not IP address. Yahoo mail uses combination of techniques to filter out spam messages. It also provides mechanisms that prevent a valid user from being mistaken for a spammer. Examples are ability of the users to troubleshoot SMTP Errors by referring to their SMTP logs. Another one is the complaint feedback loop service that helps a user to maintain a positive reputation with Yahoo. Yahoo whitelisting (internal whitelisting and Return Path Certification) is also provided [4]. Unlike blacklisting, a whitelist blocks by letting the user specify the list of senders to receive mail from. The addresses of such senders are placed on a trusted-users list. Yahoo mail spam filters allows the user to use a combination of whitelist and other spam fighting feature as a way to reduce the number of valid messages that are erroneously classified as spam. Using a whitelist alone, on the other hand, will make the filter extremely tight, implying that any unauthorized user will be automatically blocked. Automatic whitelisting is used by several anti-spam systems. In this situation, the email address of an anonymous sender is checked against a database; if there

is no history of spamming, the message is delivered to the recipient's inbox, and the sender is placed to the whitelist.

2.4.1.4. Outlook Email Spam Filter

Following Gmail and Yahoo Mail, we reviewed Microsoft Outlook and how it handles spam filtering in this part [4]. Hotmail and Windows Live Mail were renamed Outlook.com by Microsoft in 2013. Outlook.com is based on Microsoft's Metro design language and closely resembles Microsoft Outlook's interface. Microsoft's Outlook.com is a set of applications, one of which being the Outlook webmail service. Users can send and receive emails through their web browser using the Outlook webmail service. It allows the users to connect cloud storage services to their account so that when they want to send an email with file attachments, they can select files from not only their computer and OneDrive account but also from Google Drive, Box, and Dropbox account. Moreover, Outlook webmail service also allows users to encrypt their email messages and disallow the recipient from forwarding the email. Whenever a message is encrypted in Outlook.com, it is only the person with the password that will be able to decrypt the message and read it. This is a security feature that ensures that the message is only read by the designated recipient. The primary distinction between the MS Outlook desktop application and the Outlook.com webmail service is that the MS Outlook desktop application allows you to send and receive emails via an email server, whereas Outlook.com is an email server. On the other hand, the Outlook.com webmail service is designed for businesses and professionals that rely on email. Moreover, MS Outlook desktop application is a commercial software that comes along with the Microsoft Office package. It is a computer software program that provides services like email management, address book, notebook, a web browser, and a calendar which allows users to plan their programmers and arrange upcoming meetings. Outlook.com has almost 400 million users [34]. According to statistics, their site receives approximately eight billion emails every day, with 30–35 percent of those emails being sent to consumers' inboxes. Outlook.com has its own unique methods for filtering email spam [35].

2.5. Email Spam Filtering Process

An email message is made up of two major components which are the header and the body. The header is the area that have broad information about the content of the email. It includes the

subject, sender, and receiver. The body is the heart of the email. It can include information that does not have a pre-defined data. Examples include web page, audio, video, analog data, images, files, and HTML markup. The email header is comprised of fields such as sender's address, the recipient's address, or timestamp which indicate when the message was sent by intermediary servers to the Message Transport Agents (MTAs) that function as an office for organizing mails. The header line usually starts with a “From” and it goes through some modification whenever it moves from one server to another through an in-between server. Headers allow the user to view the route the email passes through, and the time taken by each server to treat the mail. The available information has to pass through some processing before the classifier can make use of it for filtering [36]. Figure. 2 below depicts a mail server architecture and how spam filtering is done.

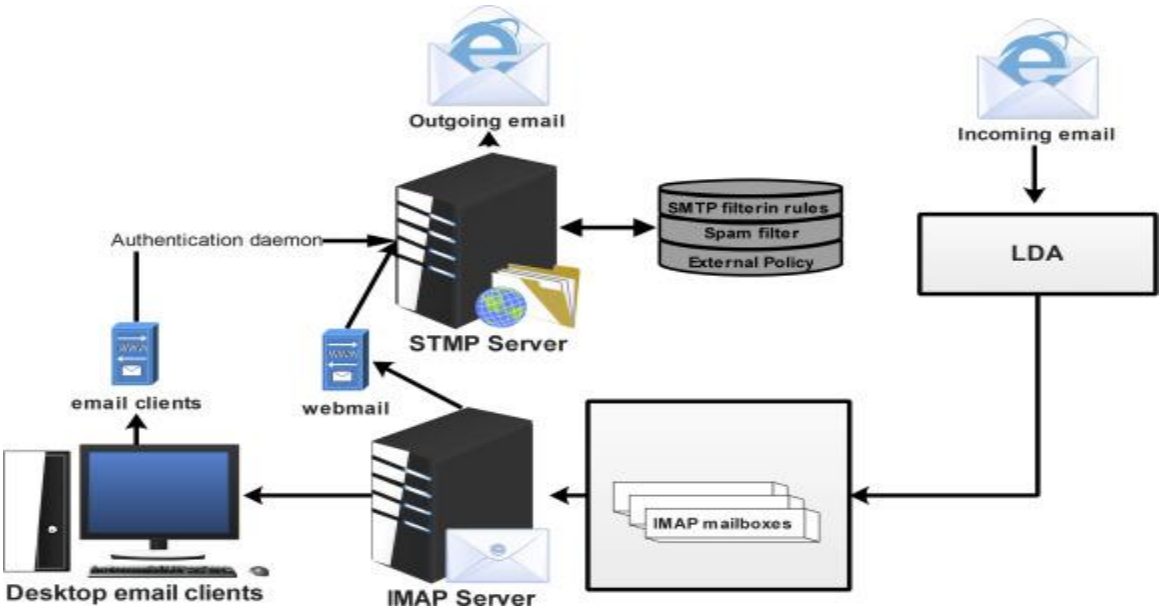

Figure 2 Email server spam filtering architecture [4]

2.6. Phishing Email

Phishing is one of the most popular forms of hacking, attempting to acquire account information and user credential details by posing as a directive coming from a legitimate source and an authority like a trustworthy company or organization. Phishing emails are also one of the easiest and most used methods. Phishing is a major threat to all Internet users and is difficult to trace or defend against since it does not present itself as obviously malicious in nature [6].

Some people, known as cybercriminals, have found ways to exploit shortcomings and faults found within the email, exploiting core email protocols, email functionalities, and weaknesses in human to machine interactions. This is known as email phishing, a cybercriminal attack vector that has increased dramatically in the number of incidents in the last years [37].

Email phishing can be described as a type of social engineering attack, manipulating the victim (email receiver) into doing as the attacker (email sender) wants [38]. It occurs when an attacker masqueraded as a trusted entity, fools the target into performing actions based on the content of an unsolicited email [39]. This can be clicking malicious links or attachments found within the unsolicited email. This can lead to the installation of malware or theft of sensitive, personal, or financial information and data.

Email phishing is most known for trying to steal personal and financial information, but it is also used to compromise computers and IT networks on a personal, business, and national level. It serves as a gateway and early phase of cybercriminal attacks, leading to more complex and dangerous situations [40].

Phishing is an act that attempts to electronically obtain delicate or confidential information from users (usually, for the purpose of fraud) by creating a replica website of a legitimate organization [39]. Phishing is, usually, perpetrated by sending deceitful and well composed emails to users. These emails usually contain links to cloned websites and clicking on this links may redirect users to a phishing website or a malware hosting website. Malware hosting websites are, usually, infected with malicious codes that can gain access to private information of users and also cause damages to users' computers. Due to the vast number of email messages received by various users today, separating legitimate emails from phishing emails is a challenging task therefore the need for a quicker, robust, and effective filtering technique cannot be overstated.

Several approaches have been proposed in the literature, including network-based approach, blacklist, whitelist, and content-based approach. Network-based approaches are costly to implement, difficult to maintain and time consuming [41]. Blacklist (that is, list of reported phishing websites) and whitelist (that is, list of target companies) approaches yield high FP and FN rates; their effectiveness is limited to the information stored in them. This limitation makes blacklist and whitelist incapable of automatically detecting new phishing attacks as they occur

[42]. The Anti-Phishing Working Group (APWG) noted that the average uptime for phishing a website is 44.39 hours (that is, less than 2 days) [43]. Content-based approach aims at capturing the content and structural properties of a data [44]. According to White et al [43], the blacklist approach is the widely used phishing detection approach adopted by many today. Nevertheless, Bergholz et al. [45], pointed out that a content-based technique is the most accurate and secure of all the phishing detection techniques mentioned above. This is because, the content-based technique is capable of discovering new fraudulent patterns in large datasets as they evolve.

Phishing is a classification problem and Martin et al [46], outlined five stages involved in phishing attacks.

1. **Planning**: At this stage, plans on who the target organization should be and how to get the email address of the organization's customers are determined.

2. **Setup**: Here, the method for sending the messages (usually mass mailing) and obtaining the revealed user's information is devised.

3. **Attack**: At this stage, the fraudulent and deceptive message is sent out to users' addresses.

4. **Collection**: Here, the information of the victimized users are captured.

5. **Attack**: At this stage, the actual fraud is committed using the captured information revealed by users at the collection stage.

The detection of phishing has been approached in a variety of ways. Adida et al [47], suggested that phishing can be tackled and eliminated at the email level, since many scammers use email as their tool for committing fraud. Dhamija and Tygar [48] also suggested that email can be eliminated at the website level. They proposed that a security toolbar may be incorporated into web browsers. Another approach proposed by Dynamic Security Skins [49] involves the use of visual hash. In this approach, visual hash was generated randomly and used to customize the web and windows form of a browser. Visual hash is responsible for identifying websites that have been authenticated successfully by the browser. Buntine also proposed a method called Cryptographic identity verification [50]. The author pointed out that this method can only work if the entire web infrastructure (both servers and client) is changed. In addition, increasing the awareness of users can increase mitigation against malicious attack; users should be well trained on various ways of identifying phishing website Khonji [51] summarizes the approaches that can

be applied to tackle phishing attacks into four categories, namely: offensive defense, correction, prevention, and detection approach, respectively.

A) Offensive Defense Approaches

The objective of approaches that fall in this category, is to neutralize the effect of the phishing attack. This method is, mostly, applicable to users that have already fallen victim to the attack (that is, users that have already filled out and submitted their private information into the HTML forms of the phishing website). In this approach, whenever a user is misled to a phishing website, a software installed on the user's browser will also submit several fake samples of information to the phishing website, so that it will be difficult for the attackers to locate the actual information submitted by the user.

B) Correction Approaches

Approaches in this category aim at, either, removing the phishing files from the website or making the phishing website inaccessible. Both can be achieved by alarming the internet service provider that hosted the website, in order that they will carry out the appropriate or required action.

C) Prevention Approaches

These approaches aim to both prevent users from falling victims and to stop phishers from defrauding users in the future. The latter can be achieved by involving law enforcement agencies. These agencies can carry out their investigation and penalize these attackers by making them pay dearly for their crimes. This serves as a deterrent and, in turn, minimizes subsequent attacks.

D) Detection Methodologies

The primary goal of this category's methodologies is to recognize phishing assaults and classify them as legitimate or illegitimate. This is normally accomplished by scanning each email for hundreds of suspected phishing features and automatically filtering them out. The analysis of phishing traits enables the detection system to react to new phishing assaults as they emerge. The blacklist detection technique, whitelist detection approach, network-based detection approach, and content-based detection approach are the four types of detection approaches. These four approaches are briefly explored below.

i. Blacklist Detection Approach

The term "blacklist" refers to a list of phishing websites that have been reported. In order to identify blacklisted phishing addresses, certain internet service providers (ISPs), online browser

providers, and email providers (such as Gmail, Yahoo, Microsoft, and others) use the blacklist strategy. These companies utilize the information on the blacklist to secure their systems and, as a result, protect their customers against phishing assaults. If an email is sent from an IP address that has already been blacklisted, the email provider can either refuse to deliver the email or send it to the recipient's spam folder. A blacklist usually contains domain names and IP addresses of previously detected phishing websites. Some blacklists also contain keywords, IP addresses of open proxies and relays, IP addresses of ISPs that host phishing websites and RFC violators (IP addresses that violate the internet and network engineering standards). Almomani et al. [52] reported that there are more than 20 spam blacklists commonly used today and these blacklists are, usually, updated at regular intervals; for example, the blacklist of Firefox browser (stored in the user's profile) is, usually, updated every 30 minutes [53].

ii. Whitelist Detection Approach

A whitelist is a list of companies that have been targeted (such as eBay, Paypal, Visa, etc.). The whitelist and blacklist approaches are fairly similar in that they both protect users against fraudulent assaults. The information contained in both a whitelist and a blacklist is the main distinction between them. A whitelist is a collection of spam-free email addresses, IP addresses, and domain names. Various providers, in general, employ a whitelist to influence their filtering decisions. For example, an organization's network administrator may elect to create a whitelist of Media Access Regulate (MAC) addresses and utilize it to control network access. Additionally, some spam filters maintain a whitelist of email addresses, IP addresses, and domain names that they utilize to determine whether or not an email is authentic.

iii. Network-based Detection Approach

This approach is used by various network administrators to secure their network from intrusion. Generally, when a user sends a message over the network, it is formatted into a smaller unit called packet which contains the message sent by the user and the IP address of the sending network. However, the IP address can be faked in such a way that it will be hidden. Generally, network-based approach aims at blocking any network packet that is deemed to be illegitimate (that is, packets that contain disguised IP addresses).

iv. Content-based Detection Approach

The content-based approach is another method that can be used to detect fraudulent attacks. This approach involves analyzing the content and structural properties of the data. For example, Microsoft

Internet Explorer (Version 7) has an inbuilt classifier, that analyzes the contents of web pages and filters them, based on some criteria [54]. Bergholz et al. [45], noted that the content-based approach is the most effective and secure of all the filtering approaches, even though [55] also noted that the black-list approach is the most widely used approach.

2.7. Types of phishing emails

Criminals have countless methods and types of phishing emails to trick email users. The differences between the email phishing techniques are the context and scenarios the threats are used in. Hackers send fraudulent emails out to literally millions of people, hoping a few will click on the attached links, documents, or pictures, with the goal of getting recipients to willingly provide valuable private information such as social security numbers, passwords, banking numbers, PINs, credit card numbers and so on. This can be achieved through a few different methods. Phishing email can be categorized into 10 [56], namely:

The Government Maneuver

This type of email looks like it originated from a federal body, such as the Microsoft, and tries to scare people into providing their information. Common messages include, 'Your insurance has been denied because of incomplete information. Click here to provide the information.' Or 'Because you illegally downloaded files, your Internet access will be revoked until you enter the requested information in the form below.'

The Friend Tactic

If an unknown individual claims to know you in an email, you are probably not suffering from amnesia. More than likely, it is an attempt to get you to wire him/her money. A variation on this theme is that one of your known friends is in a foreign country and needs your help. Before you send your 'friend' money, give them a call to verify. Your true friend's email contact list was probably hijacked.

The Billing Problem

This phishing tactic is tricky because it appears quite legitimate. This email informs you that an item you ordered online will not be delivered to you because your credit card has expired (or

your billing address is incorrect, etc.). If you click on the provided link, you will be directed to a faked website that requests updated payment/shipping information, among other things.

The Expiration Date

This type of email falsely explains that your account with [company name] is about to expire, and you must sign in as soon as possible to avoid losing all your data. Conveniently enough, there is a link in the email, which again takes you to a spoofed login page.

The Virus or Compromised Account Scare

These types of email state that your computer has been infected or that one of your accounts has been breached. In order to avoid losing your money or data or infecting your computer the email instructs you to follow a link to download the attachment.

The Contest Winner

Don't get too excited when you receive emails that claim you've won something or received an inheritance from a relative you've never heard of. 99.9% of the time, these are absolutely bogus. To claim your prize, the email requires you click a link and enter your info for prize shipment.

The Friendly Bank

Your bank may offer account notifications when certain amounts are withdrawn from your accounts. This ploy tricks you with a fake account notification stating that an amount has been withdrawn from your account that exceeds your notification limit. If you have any questions about this withdrawal (which you probably would), it gives you a convenient link that leads to a web form asking for your bank account number "for verification purposes." Instead of clicking on the link, give your bank a call. They may want to take action on the malicious email.

The Victim

Being wrongly accused of something doesn't feel good. This type of phishing email acts as an angry customer who supposedly sent you money in return for a shipped product. The email concludes with the threat that they will inform the authorities if they don't hear from you.

The Tax Communication: Practically everyone has annual taxes to submit. That's why this phishing attempt is so popular. The message states that you are either eligible to receive a tax

refund, or you have been selected to be audited. It then requests that you submit a tax refund request or tax form.

The Checkup

This is one of the more unassuming phishing email attempts. It claims [company name] is conducting a routine security procedure and requests you verify your account by providing information. This scam is especially effective if you happen to be a customer of the named business.

2.8. Machine Learning

Machine learning (ML) is a field of computer science that has existed in theory for decades but have in recent years proven to be very useful in practice. By normalizing emails to a data format that is readable for ML algorithms, they can with enough processing power and available data, process enormous amounts of information, and learn almost any email phishing pattern possible. Not only able to recognize known patterns but also use that knowledge to find new ones [57].

The data used for processing and training is what can be found in the email content. Metadata can be used to detect email spoofing. Linguistic data can be used to spot commonly recurring linguistic patterns. Email routing and network usage data can be used to identify compromised domains and phishing email campaigns [58].

With the constant flow of new data to process and learn from, the ML algorithms evolve as the email phishing attack vector evolves. This makes it able to have an almost 100% email phishing detection rate with low false positives and false negatives.

Google announced in 2017 that ML models were helping prevent 99.9% of spam and phishing messages from reaching Gmail user's inboxes. February 2020, they started using "Deep learning" AI to prevent emails containing malware from reaching their users' inboxes. With a detection rate of 99.9%, the Gmail scanner processes 300 billion Gmail attachments every week. With over 1.5 billion active users, there is almost no limit to data available for google Gmail's ML algorithms to process and self-improve. Keep in mind, 0.1% is 300 million Gmail attachments it could not detect, still making it a substantial threat [59].

2.9. Machine learning Algorithm for Email classification

Machine learning approach has been widely studied as seen section 2.8, and there are lots of algorithms can be used in email filtering. They include random forest, logistic regression, Support Vector Machines, Naïve Bayes, Neural Networks, and the Multilayer Perceptron, Adaptive and Boosting.

2.9.1. Random forest

The Random Forests classifier (RF) is a classifier that makes use of decision trees. It generates a large number of decision trees; each tree using a random number of samples and features from our data set. If classifiable data are input, the classifier returns the label that was decided on by the largest number of decision trees [60].

2.9.2. Logistic Regression

Logistic Regression (LogReg) is a well-established statistical model for classifying data. It binarily classifies data by fitting the data points to a logistic function. The classifier is very powerful for simple, linearly separable data, but its performance starts to decline for data with complex relationships between variables [61].

2.9.3. Neural Networks and the Multilayer Perceptron

Neural Networks are a type of classifier that attempt to mimic the biological brain. The network consists of connected layers of so-called nodes, which resemble neurons as we know them in biology. While neurons cannot do much by themselves, introducing a proper number of connected neurons allows for evaluation of complex functions. Some of the simplest, though very useful structures can emulate logic gates [62]. Slightly more complex networks are capable of classifying linearly separable data, but clever manipulation of input features (i.e., by the use of kernel functions) allows us to circumvent even this constraint [63]. Neural networks are highly configurable by setting hyper-parameters such as the number of hidden layers, number of neurons per layer and the optimization algorithm. Configuring the hyper-parameters of neural networks is a nontrivial task that generally requires a lot of experience and knowledge, though they can often initialize the model by making some educated guesses and improve from there [64]. Though many different types of neural networks exist, their research focuses on the

Multilayer Perceptron (MLP), which is a type of feed-forward neural network. In feed-forward neural networks, nodes do not form a cycle, meaning neurons only output to the next layer of neurons. The Multilayer Perceptron is one of the most basic versions of the neural network, consisting of only an input layer, a configurable number of hidden layers and an output layer [65].

2.9.4. Support Vector Machines

Support Vector Machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. They have High accuracy, good theoretical guarantees regarding overfitting, and with an appropriate kernel they can work well even if data is not linearly separable in the base feature space. Especially popular in text classification problems where very high-dimensional spaces are the norm.

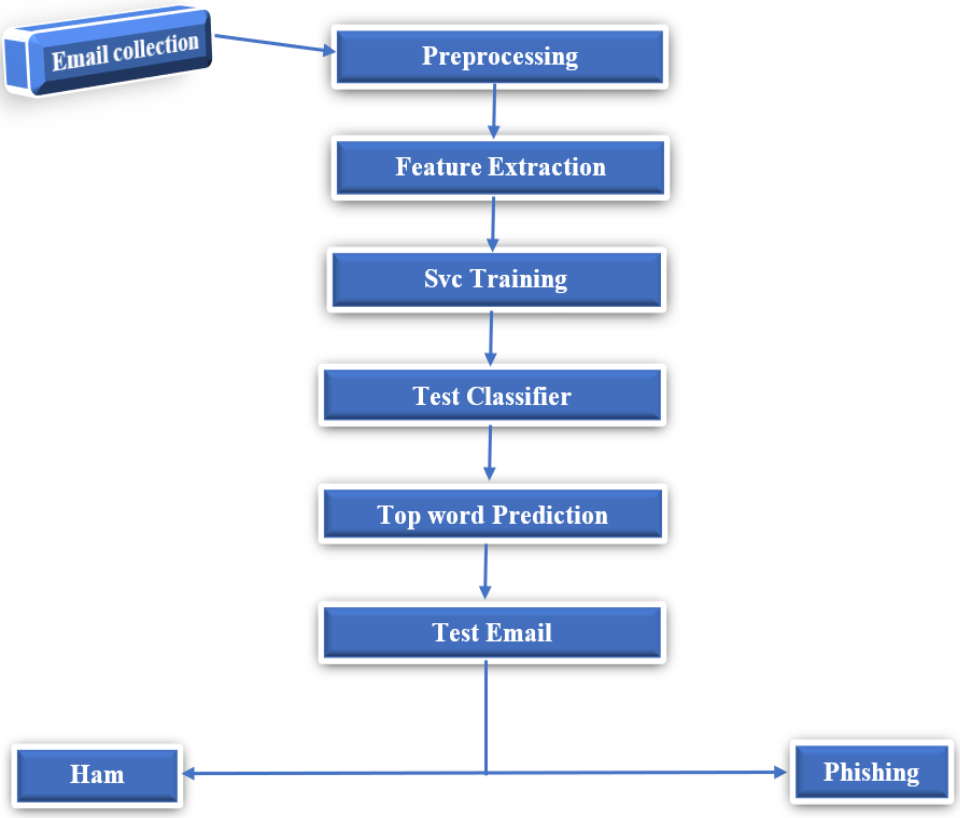

Figure 3 Process of phishing filtering based on SVC Algorithm

1. The preprocessing step was utilized to expel the noises from the email which are irrelevant and require not to be available. The pre-processing step incorporates
    a. Removal of Numbers
    b. Removal of Special Symbol
    c. Removal of URLS
    d. Stripping HTML
    e. Word Stemming.
2. Feature Extraction was utilized to separate the essential and important features from the email body. The feature transforms the email into 2D vector space having features number.
3. In the SVM Training step the email spams were utilized for the training necessity. The training dataset include content of spam and classifier were prepared utilizing it. Subsequent to training, the classifier was prepared to classify the spam emails.
4. The classifier was tested in the fourth step which is Test Classifier step with various training information to test the accuracy of the classifier.
5. In the fifth step which is Test Email step where after the training stage was finished, an example email was given as input to the classifier to characterize the email.

2.9.5. Adaptive Boosting

AdaBoost, or adaptive boosting, is a machine-learning method that combines a set of lesser classifiers to create a stronger one. The classifier chooses a "team" of other, simpler classifiers, such as SVMs (2.9.4) and Random Forests (2.9.1) and gives each one a weight. Individual classifiers will solve the problem on their own and vote on a solution based on their predictions. The AdaBoost algorithm then assesses their performance and re-assigns a weight to each classifier based on the criteria utilized. This cycle is repeated until the stopping criteria are satisfied. The most effective combination of classifiers, along with their weights, will act as our final classifier [66].

2.9.6. Naïve Bayes Algorithm

The Naive Bayes algorithm is a simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combination of values in a given dataset [67]. example, if phishing emails are arrived at due existence of phishing email keywords, then a particular

keyword can be used to more accurately assess the probability that a particular email is indeed a phishing email, compared to the assessment of the probability of phishing emails made without considering that particular keyword. In this study Naïve Bayes classifier use bag of words features to identify phishing email and a text is representing as the bag of its word. The bag of words is always used in methods of document classification, where the frequency of occurrence of each word is used as a feature for training classifier [68].

Naïve Bayes technique used Bayes theorem to determine that probabilities junk email. Some words have particular probabilities of occurring in unsolicited mail or ham email. Example, suppose that we know exactly, that the word Free could never occur in a non-spam e-mail. Then, when we saw a message containing this word, we could tell for sure that were spam e-mail. Bayesian spam filters have learned a very high spam probability for the words such as Free and Viagra, but a very low spam probability for words seen in non-spam e-mail, such as the names of friend and family member. So, to calculate the probability that e-mail is spam or non-spam Naive Bayes technique used Bayes theorem as shown in formula below [69].

$$P(spam|word) = \frac{P(spam).\ P(word|spam)}{P(spam).P(word|spam)+P(non-spam).P(word|non-spam)} \quad (2.10.6)$$

Where:

P(spamword) is probability that an e-mail has particular word given the e-mail is spam.

P(spam) is probability that any given message is spam.

P(wordspam) is probability that the particular word appears in spam message.

P(non−spam) is the probability that any particular word is not spam.

P(wordnon−spam) is the probability that the particular word appears in non-spam message. Below picture show that the step of mail classification based on Naïve Base algorithm.

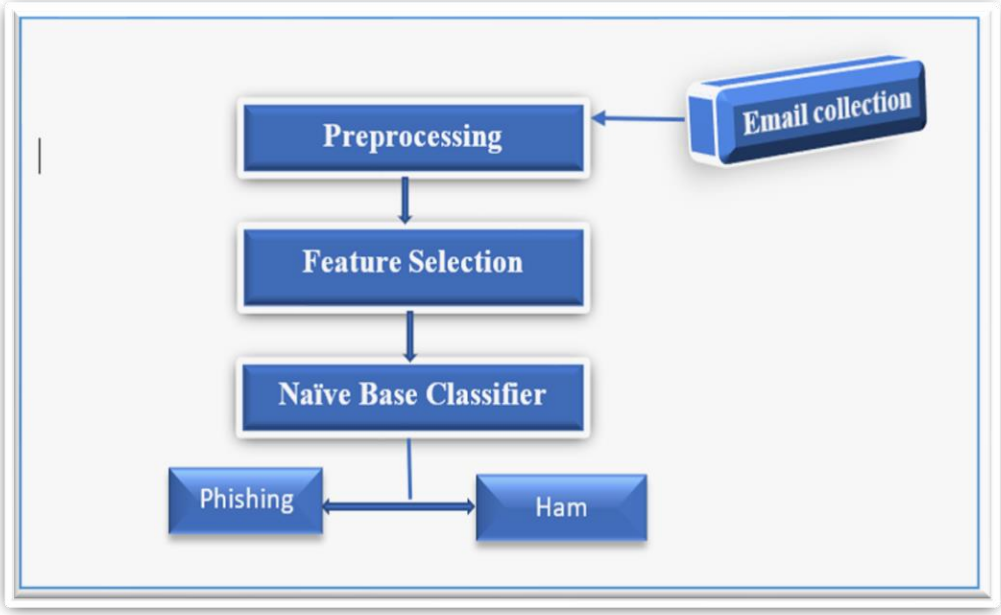

Figure 4 Process of phishing filtering based on Naive Bayes Algorithm

In our work we selected hybrid machine learning algorithm namely Naïve Bayes algorithm and SVM to develop effective phishing detection method because:

i. According to most popular machine learning classifier Naïve Bayes has a very satisfying performance among the other methods to get higher accuracy [32].
ii. It is a probabilistic classifier based on the Bayes' Theorem with strong (Naïve) independence assumptions between the features [68].
iii. It requires only a small number of training data to estimate the parameters for classification.
iv. It plays well in case of categorical input variables compared to numerical variables [61].
v. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, SVM has been shown to outperform even the most advanced classification algorithms.

Applications of Naive Bayes Algorithms

Real time Prediction: Naive Bayes is an excited learning classifier, and it is sure fast. Thus, it could be used for making predictions in real time.

Multi class Prediction: This algorithm is also well known for multi class prediction feature. Here we can predict the probability of multiple classes of target variable.

Text classification/ Spam Filtering/ Sentiment Analysis: Naive Bayes classifiers mostly used in text classification (due to better result in multi class problems and independence rule) have higher success rate as compared to other algorithms. As a result, it is widely used in Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative customer sentiments)

Recommendation System: Naive Bayes Classifier and Collaborative Filtering together builds a recommendation System that uses machine learning and data mining techniques to filter unseen information and predict whether a user would like a given resource or not.

2.10.Related Works

According to Hanif Bhuiyan [3] research various numbers of email spam filtering technique such as knowledge-based technique, clustering techniques, learning-based technique, heuristic processes and so on. The paper illustrates a survey of different existing email spam filtering system regarding machine learning technique such as Naive Bayes, SVM, K-Nearest Neighbor, Bayes Additive Regression, KNN tree, among all the existing methods of email spam filtering, some are effective, and some are trying to implement another process for increasing their accuracy rate. The study elaborates different Existing Spam Filtering system through Machine learning techniques by exploring several methods, concluding the overview of several Spam Filtering techniques and summarizing the accuracy of different proposed approach regarding several parameters. Table 1 overview of current unsolicited mail filtration methods based on author, algorithm they used to classify, dataset and accuracy performance.

S.No.	Author	Algorithms	Corpus or Datasets	Accuracy/Performance
1	Mohammed et al.	Naive Bayes, SVM, KNN, Decision Tree, Rules	Email-1431	85.96% Accuracy Achieved

2	Subramaniam et al.	Naive Bayesian	Collection of spam emails from Google's Gmail Account	96.00% Accuracy Achieved
3	Sharma et al.	Various Machine Learning Algorithms Adaptions	SPAMBASE	94.28% Accuracy Achieved
4	Banday et al.	Naive Bayes, K-Nearest Neighbor, SVM, classification Bayes Additive Regression Tree	Real life data set	96.69% Accuracy Achieved
5	Awad et al.	Naive Bayes, SVM, k Nearest Neighbor, Artificial Neural Networks, Rough Sets	Spam Assassin	99.46% Accuracy Achieved
6	Chhabra et al.	Nonlinear SVM classifier.	Enron dataset	For Dataset 3, spam: real, the ratio is 1:3, for satisfactory Recall and Precision Values
7	Tretyakov et al.	Bayesian classification, k-NN, ANNs, SVMs	PU1 corpus	94.4% Accuracy Achieved
8	Shahi et al.	Naïve Bayes, SVM	Nepali SMS	92.74% Accuracy Achieved
9	Kaul et al	SVM	Sample emails	90% ~ 95%Accuracy Achieved
10	Suganya et al.	Rule Based Method	Online Social Networks (OSNs) user post	Excellence Accuracy for Given Datasets
11	Rathi et al.	Naive Bayes, Bayes Net, SVM, and	Custom Collection	99.72% Accuracy

		Random Forest		Rate
12	Mohammed et al.	Word Filterization by Tokenization, Appling	Nielson Email-1431	Reported Satisfactory Accuracy for Proposed Method
13	Singh et al.	Naive Bayes, k-Nearest Neighbor, SVM, Artificial Neural Network.	Custom Collection	Reported Improvement of precision rate at least 2%
14	Abdulhamid et al.	Various Machine Learning Algorithms	UCI Machine Learning Repository	94.2% Accuracy Achieved
15	Sah et al.	Naïve Bayes, SVM	& Custom Collection	Reported good Accuracy overall
16	Verma et al.	Customized SVM	Apache Public Corpus	98% Accuracy Rate Reported
17	Rusland et al.	Modified Naive Bayes with selective features	Spam Base, Spam Data	Spam Base get 88%Precision Rate and Spam Data get 83%
18	ksel et al.	Microsoft Azure platform defined decision tree and SVM	Custom Collection	SVM Accuracy 97.6% Decision Tree Accuracy 82.6%
19	Choudhary et al.	Feature Engineered Naive Bayes	The SMS Spam Corpus v.0.1	96.5% True Positive Rate Accuracy
20	DeBarr et al.	Random Forest algorithm	Custom Collection	95.2% Accuracy

22	Kumar et al, 2012	Decision Tree	Spam base UCI	99%
23	Woitaszek et al.	Simple SVM with personalized dictionary	Email	95.26
24	Zhao and Zhang	Rough Set Based	Custom Collection	97.37%

Table 1 Survey of existing email filtration methods

From the study most of the approaches adopt different dataset such as ECML data and Spam base UCI archive [70]. Among several papers, Mohammad et al. introduce a classifier for feature selection which regarded as the most novel classifier for feature selection [71] [72]. Rathi et al proposed an approach considering Naïve Bayes, Bayes Net, SVM and Random Forest algorithm and obtain the higher accuracy than others which approximately crossed 99.72% accuracy. Another one is, Awad et al. which proposed an approach considering Naïve Bayes, SVM, K-Nearest Neighbor, Artificial neural Networks, "Rough sets" algorithm and obtain 99.46% accuracy which seems good on their effectiveness [71]. Zhao and Zhang implemented a rough set based model to classify emails into three categories: Spam, non-spam and suspicious and compared it with Naïve Bayesian Classifier and obtained higher accuracy than others which approximately crossed 97.37% accuracy and Woitaszek et al. used simple SVM along with a personalized dictionary for model training and obtained 95.26% accuracy [73] From the assessment it should predict that, Naïve Bayes and SVM algorithm is the highly efficient algorithm in machine learning technique and have the ability to better classification of email spam. Though all are effective but still now spam filtering system have some lacking.

Zamir, Ammara, et al. [74] Proposed a feature-centric spam email detection model (FSEDM) based on content, sentiment, semantic, user and spam-lexicon features set. Exploit the role of sentiment features along with other proposed features to evaluate the classification accuracy of machine learning algorithms for spam email detection. The classification algorithm work only content, sentiment, semantic, user and spam-lexicon features set.

Wadi' Hijawi et.al. [75] Developed and implemented of an open-source tool. To provides a flexible way to extract a large number of features from any email corpus to produce cleansed

data set. The goal of the tool is to ease the task of processing email corpus and extract large number of representative features. Improved spam detection rates based on different popular machine learning algorithms. For future work, more features are planned to be added to the extraction tool. More-over, the influence of spam features can be studied based on different spam corpora.

Pandey [76] ,Examined the absolute most well-known machine learning strategies (Naïve Bayesian Classification, SVMs, Logistic Regression, Random Forest Algorithm) and of their relevance to the issue of spam Email classification. More research should be done to rise the performance of the Naïve bayes either through hybrid system or else by decide the feature dependence issue within the naïve bayes classifier, otherwise hybrid the Immune through harsh sets.

Sah et al. [77] proposed a method for detecting of malicious spam through feature selection and improve the training time and accuracy of malicious spam detection system. They also showed the comparison of difference classifier as Naïve Bayes (NB) and Support Vector Machine (SVM) based on accuracy and computation time. The proposed approach completed by four steps such as preparing the text data, creating word dictionary, Feature extraction process and training the classifier. For preparing text data researchers split the dataset into the training set (702 mails) and a test set (260 mails) and divided into spam and ham mails. Performed feature selection process by generating feature vector matrix. According to the approach, Naïve Bayes selected as good classifiers among others.

Sharma et al. [78] proposed a method for spam detection using Support Vector Machine algorithm and feature extraction. This methodology works through several steps such as Email collections, preprocessing, feature extraction, SVM training, test classifier, top word predictors, test email and result. First, they take a dataset from Apache Public corpus. In preprocessing section, they remove all special symbol, URL and HTML tags and also unnecessary alphabet. Then they mapped all word from the dictionary using Vocab file. SVM classifier applied on the training dataset. The Accuracy of the system was 98%.

2.10.1. Challenges of Existing Email Filtration Methods

- Using of single system method instead of using hybrid systems is one of the challenges. Hybrid methods look to be the most efficient way to generate a successful anti-spam filter nowadays.

- swift adaption of spammers implements spam/phishing to gain personal information about the user for fraudulent proposes and inflexibility of spam filters to adapt the changes.

- Machine learning approaches is well known approaches to provides better techniques that are able to control unsolicited mail but due to the dynamic nature of the Web, there are no 100% secure systems around the world which can handle this problem.
- Most of the existing email filtration methods do not specifically for phishing email instead they attempt to make a distinction between spam emails and ham emails the latter also going by the name of ham emails [79].

- Other researcher discovered that bag of words model are relatively effective features for filtering spam and phishing emails, and email headers are features which are as critical as message body in detecting spam mails.

- Some studies considered using subject line, header, and message body as the most important feature in classifying messages as either spam or ham. However, it is worth mentioning that suspicious subject line, header, and body alone can lead to error in spam mail classification. Users might also need to select features manually.

- Some papers focused on feature-free methods for email spam filtering since it has proven to have higher accuracy than the feature-based technique. It should however be noted that feature-free techniques have a high computational cost since it usually takes much longer time in its email classification task. It also suffers from implementation complexity.

- Some researchers used the behavioral patterns of spammers as an important aspect of spam detection while machine learning algorithms were used for extracting the important

features from the message body. Comprehensive feature engineering might be required for better accuracy.

2.10.2. Summary

We used hybrid methods for classification algorithm (i.e., Naïve Bayes algorithm and SVM). Both are improved performance among the other machine learning methods. Due to the rapid growths up of spammer attacking techniques, researcher implementing, modeling, and designing phishing email detection mechanisms time to time accordingly. The other thing is still now spammers can easily bypass all these spam filtering applications easily. This is not because the filters are not powerful enough, it is due to the swift adaption of new techniques by the spammers and the inflexibility of spam filters to adapt the changes. For extracting features behavioral patterns important for machine learning algorithm but Comprehensive feature engineering might be required for better accuracy.

Chapter Three

3. Methodology

3.1. Introduction

The methodology used in this paper, initially, a comprehensive literature review on the features that were used in phishing emails detection as well as the data mining techniques used as illustrated in this chapter section 3.6. In addition, analysis of the phishing features found in the existing studies in the literature. The Data set can be taken from Ethio Telecom email data. The data set consists of spam emails, phishing emails, and hams emails. After preprocessed the data set (based on NLP text categorization methods) with the combination NLP and Naïve Bayes classifier algorithm are used for learning the classification model thus, filtering fishing turns on a classification problem. The stages used for our study illustrated as follows: Preprocessing, Tokenization, Feature Extraction, Feature Selection, Naïve Base Classifier and Support Vector Classifier, Test classifier finally identify spam/ phishing, and ham.

3.2. Email collection

Ethio Telecom email data and an online data phishing sample can be used to create the data collection. A total of 5229 emails were collected, with 4115 legitimate emails and 1114 phishing emails making up the dataset. The data was gathered between 2019 and 2021.

3.3. Architecture of Proposed Method

In this section, we detail the Naïve Base classifier and SVM used in our paper and our proposed set of features we extracted. The architecture of proposed phishing emails detection is shown in Figure 5. We extracted set of features from four parts including Email header, Email URL, Email body and Email script features. Then we choose Naïve Base classifier and SVM to detect phishing emails.

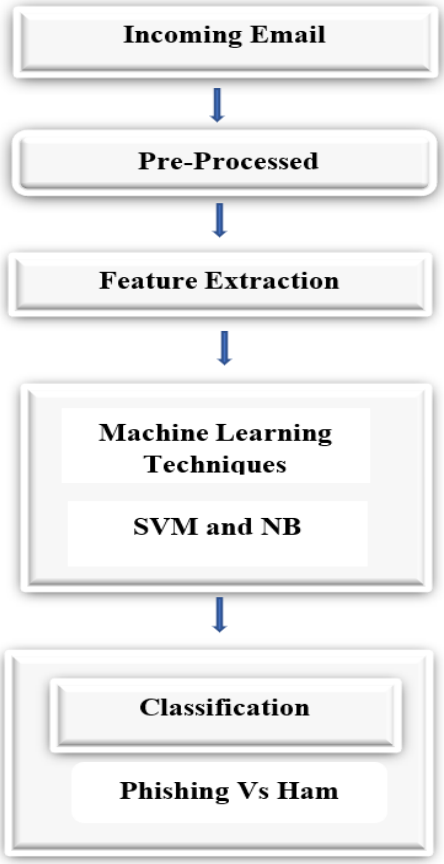

Figure 5 Architecture of Proposed Algorithm

3.3.1. Pre-processed

Data preprocessing is process of preparing the raw data and make it suitable for machine learning model. Data preprocessing is a crucial step to improve the quality of data to promote the extraction of meaningful understandings from the data. Data preprocessing in Machine Learning refers to the skill of organizing the raw data to make it suitable for a building and training Machine Learning models. In other word, data preprocessing in Machine Learning is a data mining technique that transforms raw data into an understandable and readable format. Data Preprocessing is that step in which the data gets transformed, or encoded, to bring it to such a state that now the machine can easily parse it. Data preprocessing increases the accuracy and efficiency of a machine learning model. In other words, the features of the data can now be easily interpreted by the algorithm. We have illustrated 5 preprocessed steps in our work.

3.3.1.1. Removal of Whitespace

Clean text often means tokens or a list of words that our machine learning models can work with. This means converting the raw text into a list of words and saving it again. A very simple way to do this would be to split the document by whitespace, including “”, newlines, tabs and few more. We can achieve this in Python with the split () function on the loaded string.

3.3.1.2. Removal of Punctuation

In the process of removing of punctuation, we first define a string of punctuation. Then we need to iterate over the provided string using a for loop wherein, we check if the character is a punctuation mark or not, using the membership test. We have an empty string to which we concatenate the character if it is not a punctuation. Finally, we display the cleaned-up string.

3.3.1.3 Tokenization

Tokenization refers to splitting bigger text data, essays, or corpus’s into smaller segments. These smaller segments can be in the form of smaller documents or lines of text data in other word it is the means of changing sentence into a series of words so that processing word by word can be easily performed. Given a sequence of character and a defined document, tokenization is the task of dividing it up into items, known as tokens, maybe at same time discarding characters, like punctuation. We tend to use white space character for tokenization.

3.3.1.4 Removal of Stop words

Stop words are words which are not of much significance to be used in Search Queries. Most of the search engines are programmed to ignore the stop words. In simple word stop words are not extremely meaningful inside deciding phishing or else legitimate position, so these words have been detached starting from the emails below shows sample of stop words.

i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further'.

3.3.1.5 Stemming

Stemming makes an attempt to get rid of the variations between inflected forms of a word, so as to scale back every word to its root form. Stemming can be performed using two approaches: the dictionary-based approach and porter stemming algorithm

3.3.1.6. Lemmatization

It is the procedure of compilation together the dissimilar inflected types of a word so they can be analyzing as a particular item. For example, "include", "includes," and "included" would all be represented as "include".

3.4. Feature Extraction

3.5. Features used in email classification.

Email classification features we used for our email classification are described in this section. These features were identified from different literature and Ethio Telecom unsolicited mail data. The combination of these features together forms a feature set that effectively categorized emails into phishing and Legitimate.

3.5.1. URLs Containing IP Address and Hexadecimal

The URL for many genuine websites usually contains the name of the website (e.g., http://www.ethiotelecom.com/, which tells us that this URL can be used to connect to the website of Ethio Telecom). For the purpose of identity hiding, phishers usually mask their website name by using URLs that contain IP address and hexadecimal format.

For example,

a) http://172.22.12.1/signin.ethiotelecom.com

b) http://0xd3:0xe9:0x27:0x91/signin.ethiotelecom.com

Therefore, the existence of IP-based and hexadecimal-based URLs in an email indicates that it could be a phishing email [27].

3.5.2. Differences in the “href” Attribute and the link Text

The HTML <a> tag defines an anchor that may be used to establish a link to another website. Linking to another website can be accomplished by defining a “href” attribute; this attribute

describes the location of the website that is to be linked to. The links are usually rendered to the browser after the “Link text” has been clicked (e.g., Link Text). The link text could be a plain text (e.g., Click Here), a URL (gmail.com), an image, or any other HTML element. If the link text is a URL (and it is a legitimate link), it should tally with the website location pointed to by the “href” attribute (e.g., gmail.com); if there is a disparity between the href attribute and the link text (e.g., ggmail.com), then the link is likely pointing to a phishing website. All the links (containing a URL-based link text) in an email are checked and if there is a disparity between the link text and the href attribute, then a positive Boolean feature is recorded. Similar feature was used in [80].

3.6.3. Presence of Link, Click, and Here in Link Text of a Link

The text of the links present in most phishing emails usually contain words like “Click,” “Here,” “Login,” and “Update.” For this feature, all the text of each link in an email is checked and a Boolean value is recorded based on the presence or absence of the words Click, Here, Login, Update, and Link in the Link text [81]. For example, “We are pleased to announce your COVID-19 Insurance is covered. Click here for the terms & conditions. Do you know you can bid and win business class? Submit your Bid here and elevate your experience” Similar feature was used in [82] and [80].

3.6.4. Number of Dots in Domain Name

The number of dots that should be contained in the domain name of a legitimate organization should not be more than three as proposed by Almomani et al. [52] , A binary value of 1 is recorded if an email contains a URL whose number of dots is above three. Similar feature was used in [80].

3.6.5. HTML Email

The email format for each email is defined by MIME standards. The MIME standard defines the type of content contained in each email. The content type (defined by the content-type attribute) could be plain text (indicated by “text/plain”), HTML (indicated by “text/html”). Fette et al [83], Proposed that an email is a potential phishing email if it contains a content-type with attribute “text/html”; they based their argument on the fact that it is almost impossible for phishing attacks to be launched without the use of HTML links. Similar feature was used in [80].

3.6.6. Presence of JavaScript

The script (script>) element can be used to embed JavaScript in the body of an email, or the anchor (a>) tag can be used to embed JavaScript in a link. To hide information from users, some phishers employ JavaScript. If the "JavaScript" string is found in either the email body or a link, Fette et al. [83], suggested that the email is a potential phishing email.

3.6.7. Number of Links

The total number of links embedded in an email is recorded and used as a feature for classification. Zhang and Yuan [84], explained that phishing emails usually contain multiple numbers of links to illegitimate websites. Similar feature was used in [80] and [84].

3.6.8. Number of Linked to Domain

Fette et al. [83] , refers to all the URLs present in an email that are extracted, and a count is recorded for the number of distinct domain names present in each of the extracted URLs. The recorded value is used as a feature. Take note that each domain name in an email is only counted once; subsequent occurrence (of an already counted domain name) is discarded not counted. Similar feature was used in [80] and [84].

3.6.9. From Body Match Domain Check

To extract this feature, all the domain names in an email are extracted and each of these domain names is matched with the sender's domain (i.e., the domain name referred to by the "From" field of the same email); If there is disparity between any of the comparisons, then Almomani et al. [52], suggest that the email is likely a phishing email.

3.6.10. Impersonated URI

Use of impersonated URI in the anchor text with added letters but very similar to the URL of the legitimate site, for example: Click Here. The above URL seem to be from PayPal, Inc United States but that is not the case.

3.6.11. Malware Based Phishing

This type of phishing usually involves the installing of malicious software on the victim's machine. Thereafter, the malware gathers confidential information from the victim [85]. In this case, the malware does the same job as that of a redirect to masqueraded site, upon clicking on the phishing links. This type of phishing incorporates malwares such as key loggers, Trojans via attachments and hosts file poisoning [86].

3.6.12. Encoding by ASCII or long number with character

Use of encoding schemes, for example, forming links by encoding alphabets corresponding to their ASCII codes or use of special characters such as @ on the anchor text.

3.6.13. Word List Features

According to Andronicus A. [42], study some group of words that frequently appear in phishing emails were used as features. We grouped these words into six different groups and each of these groups is used as a single feature (making a total of six different features). For each group, presence of each word is counted and normalized. The groups of words include the following.

I. Update; Confirm.
II. User; Customer; Client.
III. Suspend; Restrict; Hold.
IV. Verify; Account; Notify.
V. Login; Username; Password; Click; Log.
VI. SSN; Social Security; Secure; Inconvenience.
VII. Bank credit, Access

This feature is similar to the one proposed by Basnet et al. [87] and [88] .

Chapter Four

4. Implementation and Experiment

4.1. Introduction

In this chapter we illustrated the implementation details of experiment and the results that were obtained for all the proposed scenarios and comparison of the results. The data set used for training and testing scenario and the software that used to the system. We demonstrate the effectiveness of our best solution to the phishing detection challenge. The model's accuracy, precision, recall, and F-Score will be evaluated. Compare the performance of the system based on the accuracy, precision, recall, and F-measure of the NB and SVM algorithms.

4.2. Tools

Our study is conducted using Python for implementation, a high-level, general-purpose programming language. It is widely considered to be the preferred language for machine learning purposes [89] and, it is an open-source programming language that offers a wide range of data processing libraries, such as NumPy, Pandas and Scikit-Learn [90]. Our research uses Scikit-Learn as its main library for machine learning. It offers a wide variety of algorithms, performance metrics, and optimization methods [91].

4.3. Data Set Description

The Data set collected from Ethio Telecom and different resources of online phishing email sample type. The dataset contains 5229 emails instances with 5229 rows and 2 columns categorized as “Label” and “Message” respectively. Each Email message classified as ham (Legitimate) or phishing or spam. Source dataset is raw and is not preprocessed.

4.4. Preprocessing

We implemented the concept of TFIDF (term frequency-inverse document frequency) in order to remove words that are common in emails irrespective of them being phishing or ham. We also eliminated all numeric and alpha numeric values from the source dataset. We identified all the stop words (ex: a, an, the) by identifying the words with abnormally high frequencies and removed them, because they do not play a role in determining whether an email is phishing or not.

The term frequency–inverse document frequency (TFIDF) is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus [92]. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents.

TF-IDF for a word in a document is calculated by multiplying two different metrics:

- The **term frequency** of a word in a document. There are several ways of calculating this frequency, with the simplest being a raw count of instances a word appears in a document. Then, there are ways to adjust the frequency, by length of a document, or by the raw frequency of the most frequent word in a document.
- The **inverse document frequency** of the word across a set of documents. This means, how common or rare a word is in the entire document set. The closer it is to 0, the more common a word is. This metric can be calculated by taking the total number of documents, dividing it by the number of documents that contain a word, and calculating the logarithm.
- So, if the word is very common and appears in many documents, this number will approach 0. Otherwise, it will approach 1.

TF-IDF Steps:

Step1: HashingTF: This is a Transformer that turns a set of terms into fixed-length feature vectors.

Step2: IDF: This is an Estimator that fits on a dataset and generates an IDFModel that takes feature vectors from the HashingTF result and scales each feature with larger weights to the

features that appear less frequently in the data set. This model produces a column of features that will be utilized as input by the classifiers.

Step3: Classifier

4.5. Machine Learning Techniques

We used 80% of data for training and 20% of data as test data and for applying Naïve Bayes Classifier and Support Vector Machines. We installed dependencies from SKLearn package to implement them. In Naïve Bayes classifier for text classification, we can either use the concepts of Gaussian Naïve Bayes or Multi-variate Bernoulli Naïve Bayes or Multinomial Naïve Bayes. In our case, we have implemented Multinomial Naïve Bayes because in phishing detection, frequency of a word also plays a role in determining if an email is phishing or not and Multinomial Naïve Bayes does exactly that. In addition to Multinomial Naïve Bayes Classifier, we also implemented Support Vector Machines with the same dataset. For evaluating which of these is a better classifier for this problem we created the confusion matrix and found out the accuracy and F-score of both the cases.

4.6. Proposed Algorithm

Step 1: Select the Email content

Step 2: Extract features with help of tokenization and word count algorithm.

Step 3: Training the dataset with the help of Naive Bayesian Classifier and SVC.

Step 4: Find the probability of phishing and ham mails.

Prob_phish = (sum (train_matrix(Phish_indices,)) + 1) ./ (phish_wc + numtokens)

Prob_ham = (sum (train_matrix(ham_indices,)) + 1) ./ (ham_wc + numtokens)

Step 5: Testing the dataset

log_a = test_matrix * (log (prob_tokens_phish))' + log (prob_phish)

log_b = test_matrix * (log (prob_tokens_ham))' + log (1- prob_phish)

If output = log_a > log_b then documents are phishing else the documents are ham

Step 6: Categorize the Phishing mails and ham mails.

Step 7: Calculate the text data error and the word that is incorrectly categorized.

Num_docs_wrong = sum (xor(output, text_lables))

Step 8: Demonstrate the error rate of text data and compute the fraction of wrongly categorized word

Fraction_of_wrong = num_docs_wrong/num_test_docs

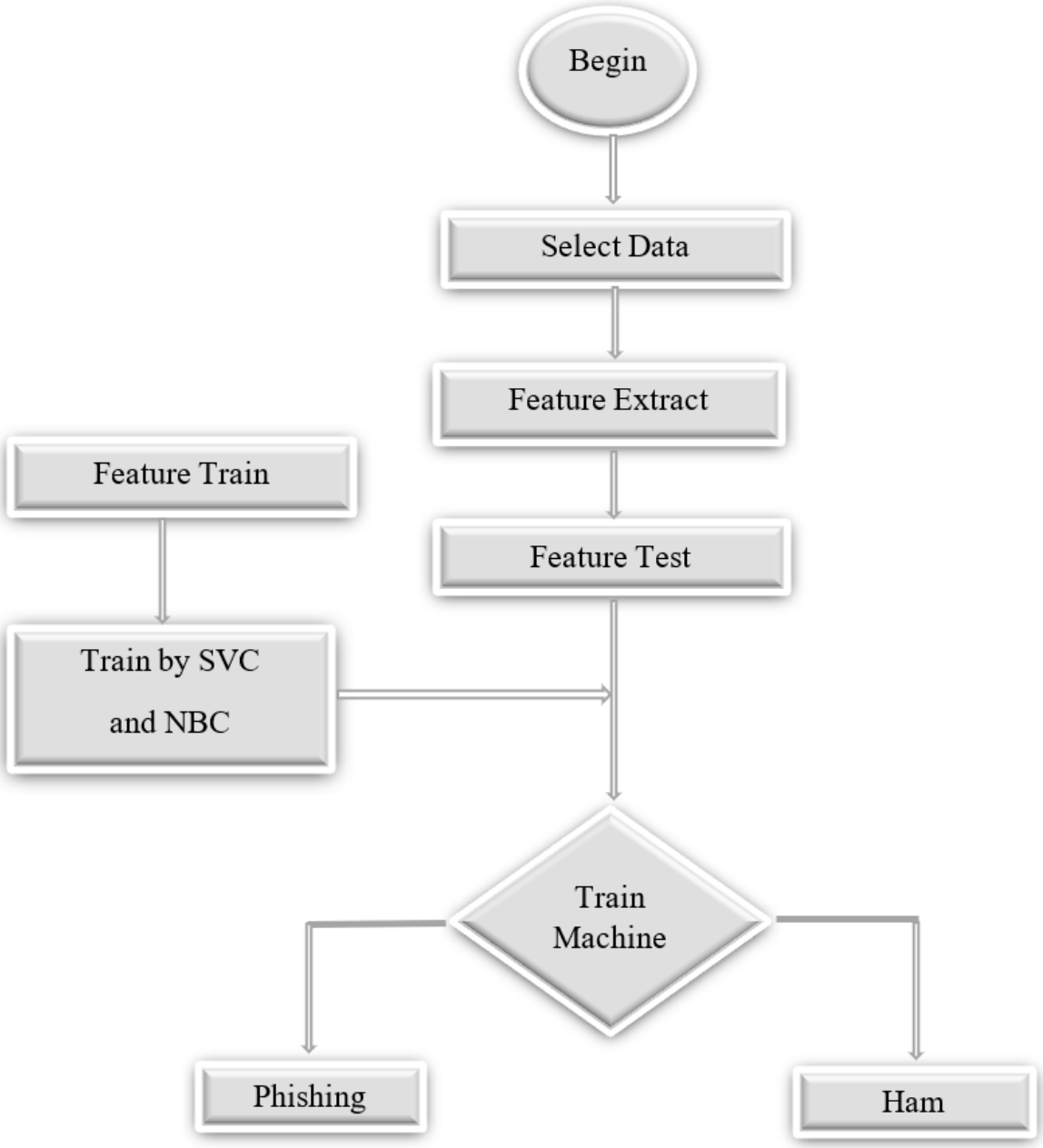

Figure 6 screenshot of proposed method for email classification

4.7. Experimental Results

In this section, we present the performance metrics of our optimal solution to the problem of phishing detection. We evaluate accuracy, precision, recall, and F-measure of the model and compare the NB and SVM algorithm in terms of Accuracy, Precision, Recall and the F-measure**.**

4.7.1. Performance metrics

Performance metrics are variables that we can use to express the performance of a system in a real number. This is done so we can compare different systems and models, which allows us to choose the best option available to use. Two widely used performance metrics are accuracy and F-Score. These two metrics are explained and compared in this appendix. We use the article: “Accuracy vs. F-Score” as a basis [93]. We make a distinction between four different situations, being the True Positive (TP), False Positive (FP), False Negative (FN), and the True Negative (TN). The "True" labels TP and TN denote correct predictions, whereas the "False" labels FP and FN denote incorrect predictions. The “Positive” labels TP and FP denote the presence of the researched phenomenon anticipated by the model, whereas the “Negative” labels denote its absence. For clarity, we give a matrix known as a "Confusion Matrix". This matrix is found in Table 2 below.

	Classified Phishing	**Classified Ham**
Actual Phishing	TP	FN
Actual Ham	FP	TN

Table 2 confusion matrix

	Classified Phishing	Classified Ham
Actual Phishing	824	3
Actual Ham	10	209

Table 3 Confusion Matrix for SVC

Table 3 shows the results achieved by Support Vector Classifier and we can see from the table the correctly classified 824 instances as Phishing email (TP), and 209 instances correctly classified as regular or Ham email (TN), and the 10 instances have been classified as Phishing but actually they are not (FP), and 3 instances has been classified as ham but actually they are phishing email (FN).

	Classified Phishing	Classified Ham
Actual Phishing	826	1
Actual Ham	25	194

Table 4 Confusion Matrix for NB

Table 4 shows the Support Vector Classifier's results, which show that 826 instances were correctly classified as phishing emails (TP), 194 instances were correctly classified as regular or ham email (TN), 25 instances were classified as phishing but were not (FP), and 1 instance were classified as ham email but were phishing email (FN).

i. Accuracy

Our first metric, accuracy, is the simplest metric available to us. It expresses the number of correct predictions as a fraction of the total number of predictions. Accuracy is the rate of correct predictions that the model achieving when compared with the actual classifications in the dataset. On the other hand, Precision and recall are two evaluation techniques, which calculated based on confusion matrix as shown in Table 4.5.1 and computed according to Equations 4.5.1.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (4.5.1)$$

Where,

True Positive (TP): The number of phishing emails that were correctly identified.

False Negative (FN): The number of phishing emails was detected as ham emails.

False Positive (FP): The number of ham emails was detected as phishing emails,

True Negative (TN): The number of ham emails was detected as ham emails.

ii. Precision

A slightly more advanced performance metric is precision. Precision indicates the correct positive cases as a fraction of all predicted positives. This metric only penalizes false positives, meaning false negatives and true negatives have no effect. Precision is evaluated by Equation (4.5.2).

$$Precision = \frac{TP}{TP+FP} \quad (4.5.2)$$

iii. Recall

In addition to precision, we use a similar metric known as recall. Recall indicates the correct positives as a fraction of all positive cases. This metric only penalizes false negatives, meaning it is not affected by false positives and true negatives. Recall is evaluated by Equation (4.5.3).

$$Recall = \frac{TP}{TP+FN} \qquad (4.5.3)$$

iv. F – Measure

F – Measure is calculated as the harmonic mean of recall and precision.

$$F\text{-}measure = \frac{2*precision*Recall}{precision+Recall} \qquad (4.5.4)$$

In our work Naïve Bayes and Support Vector Machine classifier were implemented and compared to each other in terms of accuracy score, precision, recall and F-measure. The comparison of classifiers results is shown in the following table 4.5.2.

	Accuracy	**Precision**	**Recall**	**F-Measure**
Naïve Bayes	97.51%	97.07%	99.88%	96.1%
SVM	98.76%	98.8%	99.63%	98.1%

Table 5 Comparison of NB and SVM algorithm

The results show that Support vector machines outperforms to the Multinomial Naïve Base classifier in detection of phishing mails. Even though it is a minor difference and Multinomial

Naïve Base classifier also does a proper job we have to always build the better machine to solve our problems. Hence, SVM is better at filtering phishing mails from ham mails.

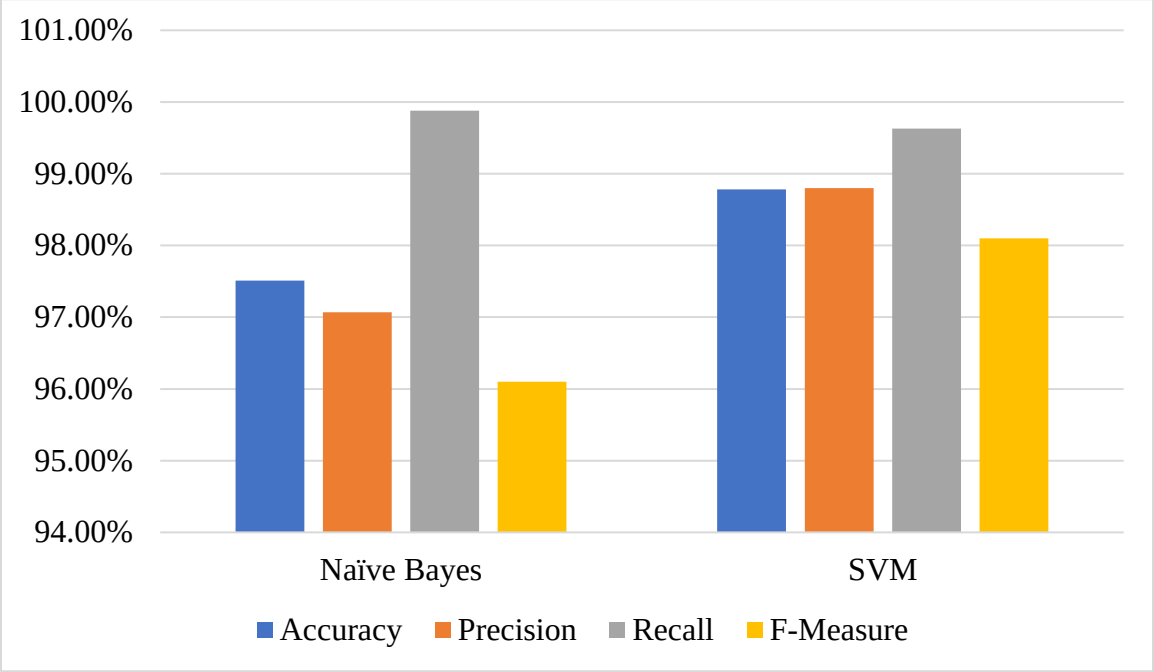

Figure 7 Comparison of NB and SVM algorithm in terms of Accuracy, Precision, Recall and the F-measure

Chapter Five

5. Conclusion, Recommendation and Future Work

5.1. Conclusion

In this paper we review some of the most famous machine learning methods and of their relevance to the problem of phishing email classification. In our research we investigated challenges of existing mail filtration methods to prevent bypass. The paper presents a survey of different existing email spam filtering systems regarding machine learning techniques such as Naive Bayes, SVM, K-Nearest Neighbor, Bayes Additive Regression, and KNN tree. Among all the existing methods, some are effective, and some are trying to implement another process to increase their accuracy rate.

In our research, we identified various types of email phishing methods and studied detection methods. It is a distinguishing feature between malicious links and malicious content that attackers use to bypass it. We used hybrid methods for the email classification algorithm (i.e., the Support Vector Machine Algorithm and the Naive Bayes algorithm). We validated the accuracy of the two algorithms. According to our experiment, Support Vector Machines outperformed the Naive Base in detecting phishing emails with an accuracy of 98.76% and 97.51%, respectively.

Using of single system method instead of using hybrid systems is one of the challenges. Hybrid methods look to be the most efficient way to generate a successful anti-spam filter nowadays.

The majority of present email filtration technologies do not particularly target phishing emails, instead attempting to distinguish between spam and ham emails, the latter of which is also known as ham emails. Our study focused on phishing email detection.

5.2. Recommendation and Future Work

In our study we identified the challenge of existing mail filtration methods which is unable to control bypass. We mention them below what are done in our work and recommended for future researcher.

- Swift adaption of phishers implements phishing to gain personal information about the user for fraudulent proposes and inflexibility of spam filters to adapt the changes. Still now hot research area of anti-phishing system. Due to the dynamic nature of the Web, there are no 100% secure systems around the world which can handle this problem.

- Some papers focused on feature-free methods for email spam filtering since it has proven to have higher accuracy than the feature-based technique. It should however be noted that feature-free techniques have a high computational cost since it usually takes much longer time in its email classification task. It also suffers from implementation complexity.

- Some researchers used the behavioral patterns of spammers as an important aspect of spam detection while machine learning algorithms were used for extracting the important features from the message body. Comprehensive feature engineering might be required for better accuracy.

References

[1] L. L. P. Davie and B. S., "Application in Computer Networks a systems approach," in *a systems approach*, USA, Elsevier, Inc, 2012, pp. 700-708.

[2] . M. Vergelis, T. Shcherbak, T. Sidorina and T. K. , "Spam and phishing in 2018," 11 October 2018. [Online]. Available: https://securelist. ru/spam-and-phishing-in-2018/93453. [Accessed 25 06 2021].

[3] H. Bhuiyan, A. Ashiquzzaman, T. I. Juthi and S. Biswas, "A Survey of Existing E-Mail Spam Filtering Methods Considering," *Global Journal of Computer Science and Technology: C Software & Data Engineering*, vol. 18, no. 2, pp. 0975-4350, 2018.

[4] E. G. Dada, S. B. Joseph , C. Haruna, O. A. Adebayo and E. A. Opeyemi, "Machine learning for email spam filtering: review, approaches and open research problems.," *Heliyon*, vol. 6, pp. Machine learning for email spam filtering: review, approaches and open research problems., 2019.

[5] R. Surmacz and Tomasz, "Reliability of e-mail delivery in the era of spam," in *In 2nd International Conference on Dependability of Computer Systems (DepCoS-RELCOMEX'07)*, Jun 2007.

[6] I. Vayansky and S. Kumar, "Phishing–challenges and solutions," *Computer Fraud & Security*, vol. 1, pp. pp.15-20, 2018.

[7] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg and E. Almomani, "A Survey of Phishing Email Filtering Techniques," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2070 - 2090, FOURTH QUARTER 2013.

[8] S. and P. , "Phishing Activity Trends for 2021," Polymer Solutions, New York, 2021.

[9] N. B. C. Moradpoor and B. Buchanan, "Employing machine learning techniques for detection and classification of phishing emails," in *In 2017 Computing Conference*, 2017.

[10] M. Chandrasekaran, K. Narayanan and S. Upadhyaya, "Phishing email detection based on structural properties," in *InNYS Cyber Security Conference*, 2006.

[11] C. P. Lueg, "From spam filtering to information retrieval and back: seeking conceptual foundations for spam filtering," *Proceedings of the American Society for Information Science and Technology*, vol. 42, no. 1, 2005.

[12] X.-L. Wang, "Learning to classify email: a survey," in *In 2005 International conference on machine learning and cybernetics*, 2005.

[13] H. e. Rahman, "Data mining applications for empowering knowledge societies," IGI Globa,

2008.

[14] M. N. Marsono, M. W. El-Kharashi and F. Gebali, "Binary LNS-based naïve Bayes inference engine for spam control:noise analysis and FPGA implementation," *IET Computers & Digital Techniques,* vol. 2, no. 1, pp. 56-62, 2008.

[15] Y. S. K. Ang, W. Y. Yuanchen He and D. Alperovitch, "Support vector machines and random forests modeling for spam senders behavior analysis," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference 2008*, 2008.

[16] M. Rathi and V. Pareek, "Spam mail detection through data mining-A comparative performance analysis," *international Journal of Modern Education and Computer Science,* vol. 5, no. 12, p. 31, 2013.

[17] D. DeBarr and H. Wechsler, "Spam detection using clustering, random forests, and active learning," in *In Sixth Conference on Email and Anti-Spam. Mountain View*, California, 2009.

[18] M. Sahami, S. Dumais, D. Heckerman and E. Horvitz, "A Bayesian approach to filtering junk e-mail," *Learning for Text Categorization: Papers from the 1998 workshop,* vol. 62, pp. 98-105, 1998.

[19] Saleh, A. Jabbar, A. Karim, B. Shanmugam, S. Azam, K. Kannoorpatti and a. F. D. B. Mirjam Jonkman, "An Intelligent Spam Detection Model Based onArtificial Immune System," *information,* vol. 10, no. 6, p. 209, 2019.

[20] A. S. Tanenbaum and D. J. Wetherall, "The application layer in Computer Networks," in *Pearson Education, Inc*, vol. 5th, USA, Pearson Education, Inc, 2011, pp. pp. 623-646.

[21] R. P. Singh and O. S. Lisa, "Blogs: Emerging knowledge management tools for entrepreneurs to enhance marketing efforts," *journal of Internet Commerce,* vol. 7, no. 4, pp. 470-484, 2008.

[22] Bandara, Arosha, D. Nicodemos, L. Emil, S. Morris and D. and Naranker, "Handbook of network and systems administration," in *Email*, Amsterdam, Elsevier, 2007, pp. 147-172.

[23] Verizon, "Verizon's 2019 Data Breach Investigations Report," Verizon Trademark Services LLC or it, USA, 2019.

[24] Gatefy, "What is a malicious URL?," Gatefy, UAE, 2021.

[25] Symantec, "Internet Security Threat Report (ISRT) - 2019," in *CYBER CRIMINALS TARGETPAYMENT CARD DATA*, USA, Symantec, 2019, p. 48.

[26] N. Lord, "Social Engineering," *Social Engineering Attacks: Common Techniques & How to Prevent an Attack,* pp. 3-6, 20 December 2020.

[27] A. A. Akinyelu and O. A. Aderemi, "Classification of phishing email using random forest machine learning technique," *Journal of Applied Mathematics,* vol. 2014, no. Special, 2014.

[28] M. R. Islam and W. Zhou, "Architecture of adaptive spam filtering based on machine learning algorithms," in *International Conference on Algorithms and Architectures for Parallel Processing*, Springer, Berlin, Heidelberg, 2007.

[29] V. Christina, S. Karpagavalli and G. Suganya, "Email spam filtering using supervised machine learning techniques," in *international Journal on Computer Science and Engineering (IJCSE),* Govindarajulu, 2010.

[30] G. L. Wittel and S. F. Wu, On Attacking Statistical Spam Filters, CEAS: Citeseer, 2004.

[31] P. M. V. Divya and U. R. Mouli, Web based optical character recognition application using flask and tesseract, Elsevier, 2021.

[32] W. A. Awad and S. M. ELseuofi, "Machine learning methods for spam e-mail classification," in *International Journal of Computer Science & Information Technology (IJCSIT)*, 2011.

[33] L. Ivan, "How Spam Filters Work (And How to Stop Emails Going to Spam)," seventh sense, 2020.

[34] T. Heinz, "How Many People Use Email Worldwide?," 2021.

[35] S. Youn and D. McLeod, "Efficient Spam Email Filtering using Adaptive Ontology," in *Fourth International Conference on Information Technology: New Generations (ITNG 2007)*, Las Vegas, Nevada, USA, 2007.

[36] F. O. Isinkaye, Y. O. Folajimi and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian informatics journal,* vol. 3, no. 16, pp. 261-273, 2015.

[37] A. Zainab, C. Hewage, L. Nawaf and I. Khan, "Phishing Attacks: Recent Comprehensive Study and a New Anatomy," *Frontiers in Computer Science,* vol. 3, p. 6, 2021.

[38] K. D. a. S. N. P. Tandale, "Different types of phishing attacks and detection techniques," *2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC),* pp. 295-299, 2020.

[39] M. Khonji, Y. Iraqi and A. Jones, "Phishing detection: a literature survey," *IEEE Communications Surveys & Tutorials,* vol. iv, pp. pp 2091-2121, 2013.

[40] E. M. Hutchins, M. J. Cloppert and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare & Security Research 1,* vol. i, p. 80, 2011.

[41] J. Von Eichborn, M. S. Murgueitio, M. Dunkel, S. Koerner, P. E. Bourne and R. Preissner, "PROMISCUOUS: a database for network-based drug-repositioning," *Nucleic acids research,* vol. 6, pp. D1060-D1066, 2010.

[42] A. A. Andronicus, "improved techniques for phishing email detection based on random forest and firefly-based support vector machine learning algorithms," *Doctoral dissertation,* 2014.

[43] O. A. Adewumi and A. A. Akinyelu, "A hybrid firefly and support vector machine classifier for phishing email detection," *Kybernetes,* 2016.

[44] N. A. A. Abdelhamid and F. Thabtah, "Phishing detection based associative classification data mining," *Expert Systems with Applications,* vol. 16, no. 41, pp. 5948-5959, 2014.

[45] A. Bergholz, S. G. Jan De Beer and G. P. a. S. S. Marie-Francine Moens, "New filtering approaches for phishing email," *Journal of computer security,* vol. vol. 18, pp. 7-35, 2010.

[46] A. M. N. Anutthamaa, M. M. S. F. M. Sathyavathy and P. Venkatesan, "A Framework for Predicting Phishing Websites Using Neural Networks," *International Journal of Computer Science Issues (IJCSI),* vol. 8, pp. 330-336, 2011.

[47] B. S. H. Adida and R. L. Rivest, "Lightweight encryption for email," *Proceedings of the USENIX Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI 2005),* pp. 93-99, 2005.

[48] J. D. T. R. Dhamija and M. Hearst, "Why phishing works," *Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI),* vol. 1, pp. 581-590, 2006.

[49] R. Dhamija and J. D. Tygar, "The battle against phishing: Dynamic security skins," *Proceedings of the 2005 symposium on Usable privacy and security,* pp. 77-88, 2005.

[50] W. L. Buntine, "A theory of learning classification rules," *Doctoral dissertation, School of Computing Science, University of Technology,* 1992.

[51] Y. I. M. Khonji and A. Jones, "Phishing Detection: A Literature Survey," *IEEE Communications on Surveys & Tutorials,* vol. 15, pp. 2091-2121, 2013.

[52] A. Almomani, T.-C. Wan, A. Altaher, A. Manasrah, M. A. Eman ALmomani, E. ALomari and S. Ramadass, "Evolving Fuzzy Neural Network for Phishing Emails Detection," *Journal of Computer Science,* vol. vol.8, p. 1099–1107, 2012.

[53] N. P. F. Schneider, M. C. R. Moll and B. Rakowski, "Phishing Protection Design Documentation," *Phishing Protection Design Documentation",* p. http://wiki.mozilla.org/PhishingProtection:_Design_Documentation, 2014 September 2007.

[54] A. Bergholz, G. P. Jeong Ho Chang, F. Reichartz and S. Strobel, "Improved Phishing

Detection using Model-Based Features," *Proceedings of the Conference on Email and Anti-Spam (CEAS)*, 2008.

[55] J. S. White, J. N. Matthews and J. L. Stacy, "A method for the automated detection phishing websites through both site characteristics and image analysis," *SPIE Defense, Security, and Sensing*, pp. 84080B-84080B-11, 2012.

[56] D. E. GCIH and P. C. QSA, "Cybersecurity Top 10 Types of Phishing Emails," 2011.

[57] V. M. M. D. Shahrivari and M. Izadi, "Phishing Detection Using Machine Learning Techniques," *arXiv preprint arXiv*, p. 11116, 2020.

[58] TenserFlow, "TenserFlow," TenserFlow, 21 03 2020. [Online]. Available: https://www.tensorflow.org/resources/learn-ml. [Accessed 06 june 2021].

[59] D. W. Forbes, "Forbes," 2020. [Online]. Available: https://www.forbes.com/sites/daveywinder/2020/02/28/google-confirms-new-ai-tool-scans-300-billion-gmail-attachments-every-week/#4a4d0813edd1. [Accessed 27 03 2021].

[60] L. B. a. A. Cutler, "Random forests–classification description," *Department of Statistics Homepage*, 2007.

[61] S. D. N. Abu-Nimeh, X. Wang and S. Nair, "A comparison of machine learning techniques for phishing detection," *In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*, pp. 60-69, 2007.

[62] S. Obumneme Dukor, "Neural Representation of AND, OR, NOT, XOR and XNORLogic Gates (Perceptron Algorithm)," 2018.

[63] Y. Freund and R. E. Schapire, "Large Margin Classification Using the PerceptronAlgorithm," *Machine Learning,*, vol. 37, no. 3, p. 77–296, 1999.

[64] V. Alto, "Neural Networks," *parameters, hyperparameters and optimizationstrategies*, 2019.

[65] C. M. Bishop, "Pattern Recognition and Machine Learning," *information scienceand statistics*, 2013.

[66] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learningand an application to boosting," vol. 904, 1995.

[67] T. Patil and S. Shereka, "Performance Analysis of Na¨ıve Bayes and Classification Algorithm for DataClassification," *International Journal Of Computer Science And Applicatio*, 2013.

[68] Saed, "saedsayad.com," Naive Bayesian, 2017. [Online]. Available:

http://www.saedsayad.com/naive_bayesian.htm. [Accessed 25 02 2021].

[69] N. F. Rusland, N. Wahid, S. Kasim and H. Hafit, "Analysis of Na ̈ıve Bayes Algorithm for Email SpamFiltering across Multiple Datasets," *International Research and Innovation Summit (IRIS2017),* vol. 226, pp. 2-4, 2017.

[70] T. Fawcett, "In vivo spam filtering," *a challenge problem for KDD Explorations Newsletter,* vol. 5, no. 2, pp. 40-148., 2003.

[71] W. A. Awad and a. S. M. ELseuofi, "Machine Learning methods for E-mail Classification," *International Journal of Computer Applications,* vol. 16, no. 1, pp. 39-45, 2011.

[72] S. Mohammed, O. Mohammed, S. F. Jinan Fiaidhi and T. H. Kim, "Classifying unsolicited bulk email (UBE) using python machine learning techniques," *International Journal of Hybrid Information Technology,* vol. 6, no. 1, pp. 43-56, 2013.

[73] M. Crawford, T. M. Khoshgoftaar, A. N. R. Joseph D. Prusa and H. A. Najada, "Survey of review spam detection using machine learning techniques.," *Journal of Big Data,* vol. 2(1), pp. 1-24, 2015.

[74] A. Zamir, W. M. Hikmat Ullah Khan, T. Iqbal and A. U. Akram, "A feature-centric spam email detection model using diverse supervised machine learning algorithms," *The Electronic Library.,* 2020.

[75] H. Faris, A.-Z. A. Ja'far Alqatawna and I. Aljarah, "Improving email spam detection using content based feature engineering approach," in *In 2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT),* 2017.

[76] P. C. a. A. Pandey and T. N. Ansari., "A Hybrid Algorithm for Malicious Spam Detection in Email through Machine Learning," *International Journal of Applied Engineering Research,* vol. 13, no. 23, pp. 6971-16979, 2018.

[77] U. K. Sah and N. Parmar, "An approach for malicious spam detection in email with comparison of different classifiers," *International Research Journal of Engineering and Technology (IRJET),* vol. 4, no. 8, pp. 2238-2242, 2017.

[78] R. Sharma and G. Kaur, "E-mail spam detection using SVM and RBF," *International Journal of Modern Education and Computer Science,* vol. 8, no. 8, p. 57, 2016.

[79] G. Schryen, "Anti-Spam Measures," in *nalysis and Design*, Springer Berlin Heidelberg, 2007.

[80] A. A. Akinyelu and A. O. Adewumi, "Classification of phishing email using random forest machine learning technique," *Journal of Applied Mathematics 2014,* 2014.

[81] A. Bergholz, J. H. Chang, G. Paass, F. Reichartz and S. Strobel, "improved Phishing

Detection using Model-Based Features," *Proceedings of the Conference on Email and Anti-Spam (CEAS)*, 2008.

[82] H.-H. Gao, H.-H. Yang and X.-Y. Wang, "Ant colony optimization based network intrusion feature selection and detection," in *international conference on machine learning and cybernetics*, 2005.

[83] I. Fette, N. Sadeh and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th international conference on World Wide Web*, 2007.

[84] N. Zhang and Y. Yuan, "Phishing Detection Using Neural Network.," *CS229 lecture notes*, 2012.

[85] Jakobsson and Myers., "Understanding the increasing problem of electronic identity theft," in *Phishing and countermeasures*, 2006, pp. 1,9-12.

[86] N. Akabar and Hartel, "Analysing Persuasion principles in phishing emails," 2014.

[87] R. Basnet, S. Mukkamala and A. H. Sung., "Detection of phishing attacks: A machine learning approach," in *presented at the Soft Computing Applications in Industry*, Berlin Germany, 2008.

[88] C.-W. Hsu, C.-C. Chang and C.-J. Lin, "A practical guide to support vector classification," pp. 1396-1400, 2003.

[89] S. Raschka, J. Patterson and C. Nolet, "Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence," *Information*, vol. 4, no. 11, p. 193, 2020.

[90] C. Room, "Data Science," *algorithms*, vol. 4, no. 37, p. 12, 2020.

[91] J. Hao and T. K. Ho, "Machine learning made easy: a review of scikit-learn package in python programming language," *Journal of Educational and Behavioral Statistics*, vol. 3, pp. 348-361, 2019.

[92] J. Silge and D. Robinson, "Term frequency and inverse document frequency (tf-idf) using tidy data principles".

[93] P. Huilgol, "Analytics Vidhya," 24 August 2019. [Online]. Available: https://medium.com/analytics-vidhya/accuracy-vs-f1-score-6258237beca2.

[94] E. S. V. M. f. S. D. A. Survey, "Efficient Support Vector Machines for Spam Detection: A Survey," *International Journal of Computer Science and Information Security*, vol. 1, no. 13, p. 11, 2015.