الجمهورية الجزائرية الديمقراطية الشعبية
The People's Democratic Republic of Algeria
وزارة التعليم العالي و البحث العلمي
Ministry of Higher Education and Scientific Research
University Saad Dahleb -Blida 1-

Final dissertation

department :Computer science

Major: Software Engeneering

Theme

Detection of Phishing Emails

Presented by : Miss BERDANE Ilhem

In front of the jury composed of :

Examinators: Mr. FERFERA SOFIANE

Mme. CHERIGUENE SORAYA

Promotors :Mme. BOUMAHDI FATIMA

Mr. Remmide mohammed abdelkarim

2021/2022

Acknowledgements

“Praise be to Allah. The cherisher and the Sustainer of the world”

Foremost, i would like to express my gratitude to my supervisor and co-supervisor **Phd.Boumahdi Fatima and Remmide Mohamed Abdelkrim** for the continuous support, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this dissertation. I could not have imagined having a better supervisors and mentors.

Finally, i cannot deny the debt of gratitude owed to my Parents, words can neither qualify nor quantify how helpful their guidance and advice have been, I am forever grateful for their support.

Abstract

In recent years, cyber criminals have successfully invaded many important information systems by using phishing mail, causing huge losses. The detection of phishing mail from big email data has been paid public attention. However, the camouflage technology of phishing mail is becoming more and more complex, and the existing detection methods are unable to confront with the increasingly complex deception methods and the growing number of emails. In this paper we transformed the probleme from classification of emails into similarity detection between two emails in order to classify them into 4 mains classes " Normal, Harrassment , suspicious and fraudulent" to solve this probleme we used Siamese Neural networks which gave us an accuracy of 95.13%.

Key words

NLP, Siamese network, deep learning, phishing attacks, phishing detection, similarity learning.

Résumé

Ces dernières années, les cybercriminels ont reussi a envahir de nombreux systèmes d'information importants en utilisant les emails d'hameçonnage, causant d'énormes pertes. La détection de l'email d'hameçonnage à partir de données de messagerie volumineuses a retenu l'attention du public. Cependant, la technologie de camouflage des emails d'hameçonnage devient de plus en plus complexe, et les méthodes de détection existantes sont incapables de faire face aux méthodes de tromperie de plus en plus complexes et au nombre croissant d'emails. Dans cet article, nous avons transformé le problème de classification des e-mails en détection de similarité entre deux e-mails afin de les classer en 4 classes principales "Normal, Harcèlement, suspect et frauduleux". Pour résoudre ce problème, nous avons utilisé le réseau de neurones siamese qui nous a donné une précision de 95,13 %.

Mots-clés

NLP ; apprentissage profond, reseau de neurone siamese, attaques de phishing, détection d'hameçonnage, apprentissage par similarité.

ملخص

في السنوات الأخيرة ، نجح مجرمو الإنترنت في غزو العديد من أنظمة المعلومات المهمة باستخدام بريد التصيد الاحتيالي ، مما تسبب في خسائر فادحة. تم إيلاء اهتمام الجمهور باكتشاف رسائل التصيد الاحتيالي من بيانات البريد الإلكتروني الضخمة. ومع ذلك ، أصبحت تقنية التمويه الخاصة ببريد التصيد الاحتيالي أكثر تعقيدًا ، وأساليب الكشف الحالية غير قادرة على مواجهة أساليب الخداع المتزايدة التعقيد والعدد المتزايد من رسائل البريد الإلكتروني. في هذا البحث قمنا بتحويل المشكلة من تصنيف رسائل البريد الإلكتروني إلى اكتشاف التشابه بين رسالتين إلكترونيتين من أجل تصنيفها إلى 4 فئات رئيسية "عادي ، مضايقات ، مشبوهة ، احتيالية" لحل هذه المشكلة استخدمنا شبكات سيامية العصبية التي أعطتنا دقة تبلغ 95.13٪.

كلمات مفتاحية:

معالجة اللغة الطبيعية, تعلم عميق, الشبكة العصبية السيامية ، هجمات التصيد ، كشف التصيد ، التعلم عن طريق التشابه.

Contents

List of Figures

List of Tables

List of Equations

Introduction

Since the beginning of the era of the internet phishing emails was a conning method that haunted the users all around the world from the infamous Nigerian prince to more sophisticated techniques that criminals are using to fool their victims. Phishing is technique used to steel personal information for the purposes of identity theft using fake email messages that appear to come from legitimate businesses. This is usually done by sending emails that seem to come from reliable source to gain access to person's confidential and private information. The email may look quite authentic, featuring corporate logos and formats similar to the ones used for legitimate messages. They often include official logos from real organizations and other identifying information taken directly from legitimate Web sites, but including a deceptive URL address linking to a scam web site. To make these phishing emails be like real, the phishers may place a link that appears to go to the legitimate web site, but it actually takes victims to a scam site(Ma et al., 2009). Phishing emails is considered as the fastest rising online crime method used for stealing personal financial data and commit identity theft. Individuals who respond to phishing emails, and input the requested financial or personal information into emails and websites put themselves and their institutions at risk. The damage caused by phishing led to substantial financial loss, According to the Aite Group, 47 percent of Americans experienced financial identity theft in 2020. The group's report, found that losses from identity theft cases cost $502.5 billion in 2019 and increased 42 percent to $712.4 billion in 2020. The group explains that the huge increase was fueled by the high rate of unemployment identity theft during the pandemic, as increased and extended unemployment benefits made the sector an attractive target for fraudsters. Losses are forecast to increase again in 2021 to $721.3 billion(Almomani et al., 2013). This style of identity theft is becoming more popular and important, because of the ease with which unsuspecting people often give personal information to phishers. However, phishing has become more and more complicated and sophisticated so that phishers can bypass the filter set by current anti-phishing techniques and cast their bait to customers and organizations. With the existing massive work for phishing email detection task there is no set of features that has been determined as the best to detect phishing. Finally, there is a need to keep on enhancing the accuracy of the detection techniques. Overall the problem carried out in this research is: How to enhance the performance of the best selected features and classifiers? Our focus in this research is to build an intelligent classifier at the email level that is capable of detecting

phishing emails , we believe that detecting phishing emails can make the internet users more secure by eliminating those emails and not relying on the users' vigilance to protect them from phishing attacks. Many studies concluded that depending on human factors is not a preferred option for combating phishing attacks especially for advanced and well prepared phishing attacks that are continuously adapting themselves to known defense mechanisms. This dissertation will try to cover all the techniques used to detect phishing email. In the first chapter we are going to explore the world of phishing attacks .The second chapter presents the state of the art in which some related research to our work were elaborated and in which, we talk briefly about the deep learning and the neural networks. In the third chapter we explain our proposed solution for this problem in details. The fourth chapter is devoted to experiments and results also we make a general vision on the methods of classification.

Chapter 1 Phishing Attacks

The digital world is rapidly expanding and evolving, as are cyber criminals who rely on the illicit use of digital assets, especially personal data, to harm others. One of the most threatening crimes for all Internet users is "identity theft" which is defined as the theft of an individual's identity in order to steal and use their personal information (e.g. bank details, Social Security numbers, credit card numbers, etc.) .

Cyber criminals have also improved their methods of stealing information, however social engineering-based attacks remain their preferred method. One of the social engineering crimes that allows attackers to perform identity theft is known as a phishing attack. Phishing is one of the biggest problems because many internet users fall victim to it. Phishing emails is a social engineering attack in which phishers attempt to trick users into obtaining their confidential information by illegally using public or trusted organizations address in an automated mode so that internet users trust the message and disclose the victim's confidential information to the attack (Jakobsson and Myers, 2006).

The rest of the chapter is organized as follows. Phishing Definitions are provided,along with a few real-word examples, along with an explanation of the types of phishing attacks.

1.1 Phishing definitions

Various definitions of the term "phishing" have been proposed and debated by cybersecurity experts, researchers and agencies. Although the term "phishing" has no established definition due to its evolving nature, the term has been defined in a number of ways depending on usage and context. The process of getting the recipient to do what the attacker wants is often seen as the definition of a phishing attack.

The study (Merwe et al., 2005) defines phishing as “a fraudulent activity that involves the creation of a replica of an existing web page to fool a user into submitting personal, financial, or password data”.The above definition describes phishing as an attempt to scam the user into revealing sensitive information such as bank details and credit card numbers, by sending malicious links to the user that leads to the fake web establishment.

For instance, (PishTank,2006) defines phishing as “a fraudulen attempt, usually made through email, to steal your personal information”.

A description for phishing stated by (Kirda & Kruegel, 2005) defines phishing as “a form of online identity theft that aims to steal sensitive information such as online banking passwords and credit card information from users”.

For instance, APWG[1] defines phishing as “a criminal mechanism employing both social engineering and technical subterfuge to steal consumers’ personal identity data and financial account credentials” (APWG, 2018). Moreover, the defifinition from the United States Computer Emergency Readiness Team (US-CERT) states phishing as “a form of social engineering that uses email or malicious websites (among other channels) to solicit personal information from an individual or company byposing as a trustworthy organization or entity” (CISA, 2018). A detailed definition has been presented in (Jakobsson & Myers, 2006), which describes phishing as “a form of social engineering in which an attacker, also known as a phisher, attempts to fraudulently retrieve legitimate users’ confidential or sensitive credentials by mimicking electronic communications from a trustworthy or public organization in an automated fashion. Such communications are most frequently done through emails that direct users to fraudulent websites that in turn collect the credentials in question.”

In this study we define phishing as a socio-technical attack, in which the attacker targets specific valuables by exploiting an existing vulnerability to pass a specific threat via a selected medium into the victim’s system, utilizing social engineering tricks or some other techniques to convince the victim into taking a specific action that causes various types of damages.

1.2 Real word phishing examples

Some real-world examples of phishing attacks are discussed in this section to present the complexity of some recent phishing attacks.

Recently, phishers take advantage of the Coronavirus pandemic (COVID-19) to fool their prey. Many Coronavirus themed scam messages sent by attackers exploited people’s fear of contracting COVID-19 and urgency to look for information related to Coronavirus ,the WHO stated that

[1] APWG Is “the international coalition unifying the global response to cybercrime across industry, government and law-enforcement sectors and NGO communities” (APWG, 2020) .

COVID-19 has created an Infodemic which is favorable for phishers (Hewage, 2020). Cyber-criminals also lured people to open attachments claiming that it contains information about people with Coronavirus within the local area. Figure 1 shows an example of a phishing e-mail where the attacker claimed to be the recipient's neighbor sending a message in which they pretended to be dying from the virus and threatening to infect the victim unless a ransom was paid (Ksepersky, 2020).

Figure 1-1 Screenshot of a coronavirus related phishing email (Ksepersky, 2020).

Another example is a phishing attack discovered by security researchers at Akamai in January 2019. This attack attempts to use Google Translate to mask suspicious URLs by prefixing the suspicious URL with a legitimate-looking www.translate.google.com address to trick users into logging in (Rhett, 2019). This attack was followed by phishing scams, such as asking for Netflix payment details or embedding them in promotional tweets, redirecting users to a legitimate-looking PayPal login page. Although in the latter case the tricky/fake page is very well designed, the lack of HTTPS (Hypertext Transfer Protocol Secure) blocks and misspellings in the URL are important red flags (or giveaways), it really is a time Phishing Attempts (Keck, 2018). Figure 2 shows a screenshot of a phishing email received from the Federal Trade Commission (FTC). The email prompts users to update their payment method by clicking a link and pretends that Netflix has a problem with users' billing information (FTC, 2018).

Figure 1-2 Screenshot of Netflflix scam email (Rhett, 2019).

1.3 Types and techniques of phishing attacks

Phishers psychologically manipulate people to reveal personal information or use technical methods to attack. However, phishers often prefer to exploit human psychology rather than technical methods for deceptive attacks. The diagram below illustrates the types of phishing and the techniques phishers use to conduct phishing attacks. Each type and technique is explained in the following sections and subsections.

Figure 1-3 Phishing attack types and techniques drawing upon existing phishing attacks (Dasgupta et al., 2022)

1.3.1 Technical Subterfuge

Technical subterfuge is the act of tricking individuals into disclosing their sensitive information through technical subterfuge by downloading malicious code into the victim's system. Technical subterfuge can be classified into the following types:

1.3.1.1 Malware-Based Phishing

As the name suggests, this is a phishing attack that runs malware on users' computers. Malware is downloaded onto a victim's computer through a social engineering technique or technically by exploiting a security hole (Jakobsson & Myers, 2006). Malware-based phishing attacks take many forms; some of them are discussed below:

1.3.1.1.1 Key Loggers and Screen Loggers

Loggers are a type of malware used by phishers that can be installed via Trojan email attachments or downloaded directly onto a user's PC. The software monitors the data and records the user's keystrokes, which are then sent to phishers. Phishers use keyloggers to collect victims' sensitive information such as names, addresses, passwords, and other confidential data (Jakobsson & Myers, 2006).

1.3.1.1.2 Viruses and Worms

A virus is a type of malware that is a piece of code that spreads in another application or program by replicating itself in a self-automated manner (Jakobsson & Myers, 2006). Worms are similar to viruses, but operate differently because worms operate by exploiting vulnerabilities in the operating system without modifying any other programs. Viruses spread from one computer to another through attached documents, while worms spread through infected hosts files (F5Networks, 2018).

1.3.1.1.3 Spyware

Spyware is malicious code designed to track the websites users visit in order to steal confidential information and conduct phishing attacks. Spyware can be delivered via email and, once installed on a computer, can take control of a device and change its settings or collect information that can be used for identity theft, such as passwords and credit card numbers or banking details (Jakobsson & Myers, 2006).

1.3.1.1.4 Adware

Adware is a type of malware that displays an endless pop-up window to the user containing advertisements that can affect the performance of the device. Adware can be annoying, but most of it is safe. Some adware may be used for malicious purposes, such as: Tracking the Internet pages a user visits, or even recording the user's keystrokes (Cisco, 2018).

1.3.1.1.5 Ransomware

Ransomware is malware that encrypts a user's data after the user runs an executable program on the device. In this type of attack, the decryption key is kept until the user pays the ransom (Cisco, 2018). To make matters worse, as new variants are developed, this is difficult to detect and thus easy to bypass many antivirus and intrusion detection systems (Latto, 2020): According to a report (PhishMe, 2016), 93% The phishing emails contain encryption-ransomware.

1.3.1.1.6 Rootkits

A rootkit is a collection of malicious programs that allow access to a computer or computer network. Intruders use these toolsets to hide their actions from system administrators by modifying system call code and changing functionality (Belcic, 2020). The term "rootkit" has a negative connotation due to its association with malware, and attackers use it to warn existing system tools that

they are evading detection. These kits allow people with little knowledge to launch phishing attacks. It includes coding and bulk email software; web development software and graphic design tools (Jakobsson & Myers, 2006).

1.3.1.1.7 Session Hijackers

In this type, attackers monitor user activity by embedding malware in browser components or performing network sniffing. Monitoring is designed to hijack a session so that an attacker can perform unauthorized operations on the hijacked session, such as financial transfers (Jakobsson & Myers, 2006).

1.3.1.1.8 Web Trojans

Web Trojans are malicious programs that collect user credentials by hiding on login screens (Jakobsson & Myers, 2006). When the user enters credentials, these programs capture the stolen credentials and transmit them directly to the attacker (Jakobsson et al., 2007).

1.3.1.1.9 Hosts File Poisoning

This is a method of tricking users into visiting phishing websites by poisoning (changing) the hosts file. When the user enters a specific URL in the URL bar, the URL is translated Enter a numeric (IP) address before you visit the website. Attackers take users to fake websites, this type of phishing is hard to detect, even for a smart person and attentive users (Ollmann, 2004)

1.3.1.1.10 System Reconfiguration Attack

In this phishing attack format, the phisher manipulates the settings on the user's computer Malicious activity so that information on that PC can be compromised. System reconfiguration can be changed in a number of ways, such as by reconfiguring the operating system and changing the user's Domain Name System (DNS) server address. Wireless Evil Twins is an example A system reconfiguration attack that monitors all user traffic over a malicious WLAN Access Points (APs) (Jakobsson & Myers, 2006).

1.3.1.1.11 Data Theft

Data theft is the unauthorized access and theft of a company or company's confidential information . Data theft can occur through phishing emails that cause malicious code to be downloaded to a user's computer to directly steal confidential information stored on that computer (Jakobsson & Myers, 2006). Stolen information, such as passwords, social security numbers, credit card information, sensitive emails, and other personal information, can be accessed directly from Phishers or indirectly through sales for other purposes.

1.3.1.2 Domain Name System Based Phishing (Pharming)

Phishing in any form can compromise the Domain Name System, redirecting users to malicious websites by polluting their DNS cache with false information DNS-based phishing. While the host's file is not part of DNS, the host's file poisoning is another form of DNS-based phishing. On the other hand, by breaking DNS servers, the real IP address was changed, causing the user to be inadvertently redirected to a fake location. Users can fall victim to domain spoofing even when they click on legitimate links, as a website's Domain Name System (DNS) can be hijacked by cybercriminals (Jakobsson & Myers, 2006).

1.3.1.3 Content Injection Phishing

Content injection phishing is the insertion of fake content into legitimate websites. it's vicious The content may redirect users to fake websites, trick users into sharing their sensitive information with hackers, or may cause malware to be downloaded on users' devices (Jakobsson & Meyers, 2006).

1.3.1.4 Man-In-The-Middle Phishing

A man-in-the-middle (MITM) attack is a form of phishing in which a phisher inserts communications between two parties (i.e. a user and a legitimate website) and attempts to gain information on both parties by intercepting the victim's communications (Allman, 2004) . Therefore, the message is sent to the attacker, not directly to the legitimate recipient. Using MITM, an attacker records the information and then abuses it. MITM attacks are performed by redirecting users to malicious servers through various techniques such as Address Resolution Protocol (ARP) poisoning, DNS spoofing, Trojan keyloggers, and URL obfuscation (Jakobsson & My-ers, 2006).

1.3.1.5 Search Engine Phishing

In this phishing technique, phishers create malicious websites with attractive offers and use SEO (Search Engine Optimization) tactics to legally index them for display to users when searching for products or services. This is also known as black hat SEO (Jakobsson & Myers, 2006).

1.3.1.6 ***URL and HTML Obfuscation Attacks***

In most phishing attacks, phishers aim to convince users to click on a specific link that connects victims to a malicious phishing server instead of the target server. This is the most common technique used by phishers today. This type of attack is performed by disguising the real link (URL) the user is trying to connect to (attackers try to make their URL appear legitimate). Bad domain names and hostname obfuscation are common methods used by attackers to spoof addresses (Ollmann, 2004).

1.3.2 Deceptive Phishing

Deceptive phishing is the most common type of phishing attack, in which attackers use social engineering techniques to deceive victims. In this type of phishing, phishers use social engineering techniques to trick victims into believing falsehoods by inventing scenarios (e.g. fake account updates, security upgrades) or technical methods (e.g. using legitimate trademarks, images, and logos) The Legality of Email (Jacobson and Myers, 2006). If users believe these situations, then they will fall victim and click on the given link, which will result in their personal details being leaked to phishers. Deceptive phishing is carried out through phishing emails, fake websites, social media phone phishing, and many other mediums. The most common types of social phishing are explained below:

1.3.2.1 Spoofed Website

Also known as a phishing site, this is where phishers fake websites that look real and resemble legitimate websites. Unsuspecting users are redirected to the site by clicking on a link embedded in the email or by advertising or otherwise. If users continue to interact with fake websites, sensitive information will be exposed and collected by phishers (CSIOnsite, 2012).

1.3.2.2 Phone Phishing (Vishing and SMishing)

This type of phishing is carried out via phone calls or text messages, where the attacker pretends to be someone the victim knows or another trusted source the victim is dealing with. Users may receive persuasive security warnings from banks that lure victims into contacting specific phone numbers to trick victims into sharing passwords or PIN numbers or other personally identifiable information (PII). Victims may be tricked into clicking on embedded links in text messages. The phisher can then use the credentials provided by the victim to log into the victim's instant messaging service, thereby phishing others on the victim's contact list. Phishers can also use caller identification (CID) spoofing to trick victims into believing the call is coming from a trusted source (Aburrous et al., 2008).

1.3.2.3 Social Media Attack (Soshing, Social Media Phishing)

Social media is the new preferred medium for cybercriminals to conduct phishing attacks. Social media threats can include account theft, impersonation attacks, scams, and malware proliferation. However, detecting and containing these threats takes more time than traditional methods because social media exists outside of network perimeters.

1.3.2.4 Phishing emails

The most common threat posed by attackers is deceiving people through email communications, which remains by far the most popular type of phishing. Phishing or spoofed emails are fake emails sent randomly to thousands of victims from unreliable sources. These fake emails pretend to be from individuals or financial institutions trusted by recipients in order to convince recipients to take steps to induce them to reveal sensitive information. More structured phishing emails targeting specific groups or individuals within the same organization are called spear phishing. Using the above types, attackers can gather information about victims, such as names and addresses, making emails from trusted sources credible (Wang et al., 2008).

1.4 Structure of phishing emails

In this section, we will discuss the common structure used for all phishing emails.

1.4.1 Spoofing of online banks and retailers

Because phishing emails must resemble online banks and online merchants to gain user trust in disclosure, phishers use emails to imitate the appearance of a reputable company. The companies most often deceived are Citibank, eBay and PayPal. The industry most affected is financial services. Internet retailers and Internet service providers were also attacked. This is mainly achieved through the use of company images in fake emails and links to the company's website.

1.4.2 Link in the text is different from the destination

In fake emails, the link text displayed in the email often differs from the actual link target. However, in the example below, the email refers to http://www.chase.com and directs the user to the website http://www.climagro.com.ar/agro/chase.htm http://account.earthlink.com.

1.4.3 Using IP addresses instead of URLs

In many cases, hackers obfuscate the target website's URL. The IP address instead of the website's hostname is used to obfuscate the target website. An example of an IP address used in the URL of a spoofed email is "http://210.14.228.66/sr/".

1.4.4 Generalization in addressing recipients

Because the success of email-based phishing attacks relies on the law of large numbers, most phishing emails do not contain personalized content when targeting their recipients. Additionally, unlike legitimate corporate communications, they do not address customers using their first names as identifiers and lack embedded encrypted information, such as the last four digits of account information. used to judge authenticity. Although phishers may include these information, through the use of social engineering and other abuses.

1.4.5 Usage of well-defined situational contexts to lure victims

Most phishing emails use underlying context to create a false sense of urgency, Threats, flattery and fear of tricking users into clicking on visited hyperlinks.

1.5 Life cycle of phishing email :

As shown in Figure 2, the phishing lifecycle typically begins with a flood of emails attempting to do so convince readers to follow the link included in the email. Phishers send mass emails Hopefully some readers will fall into the email bait trap by visiting the link provided that took him to a fake website where he entered his credentials accounts to which hackers gain access (Almomani et al., 2013).

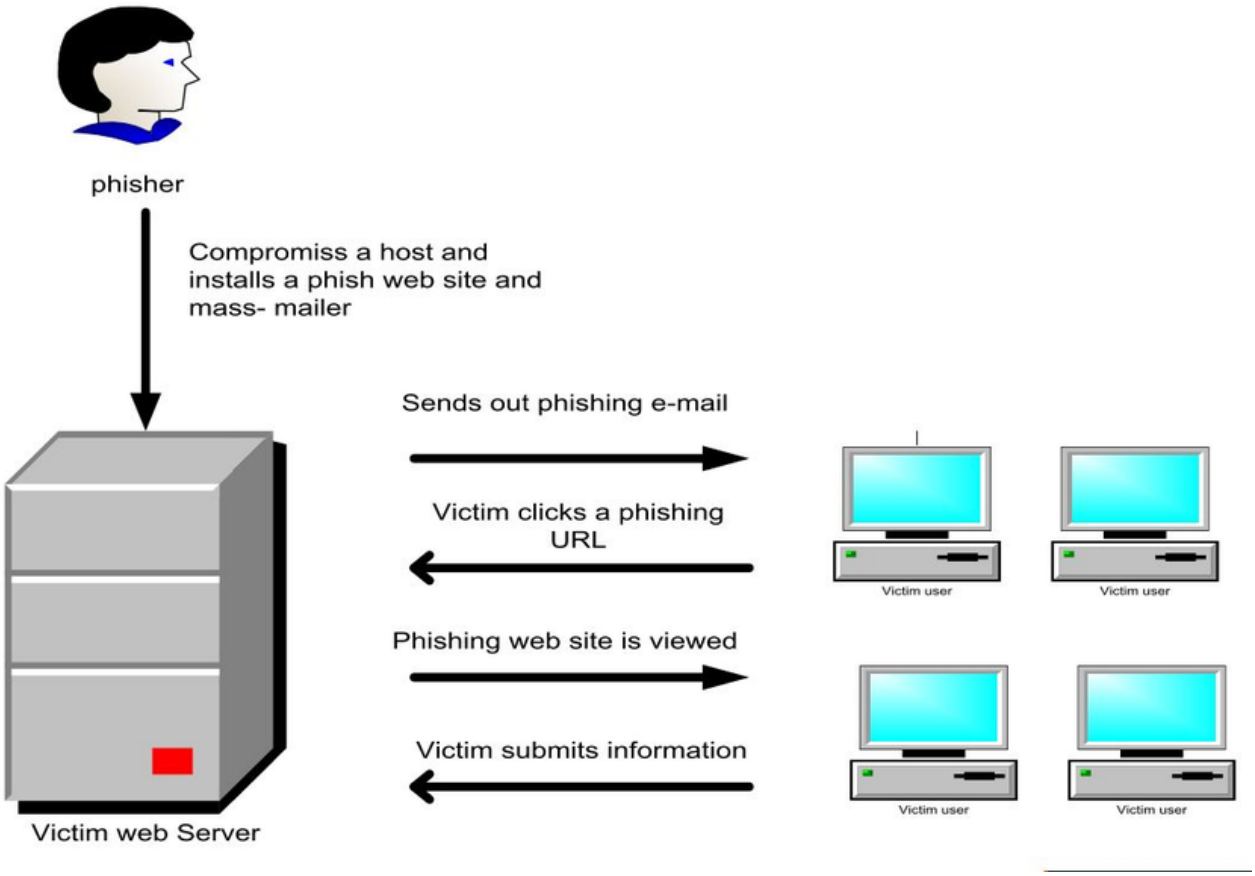
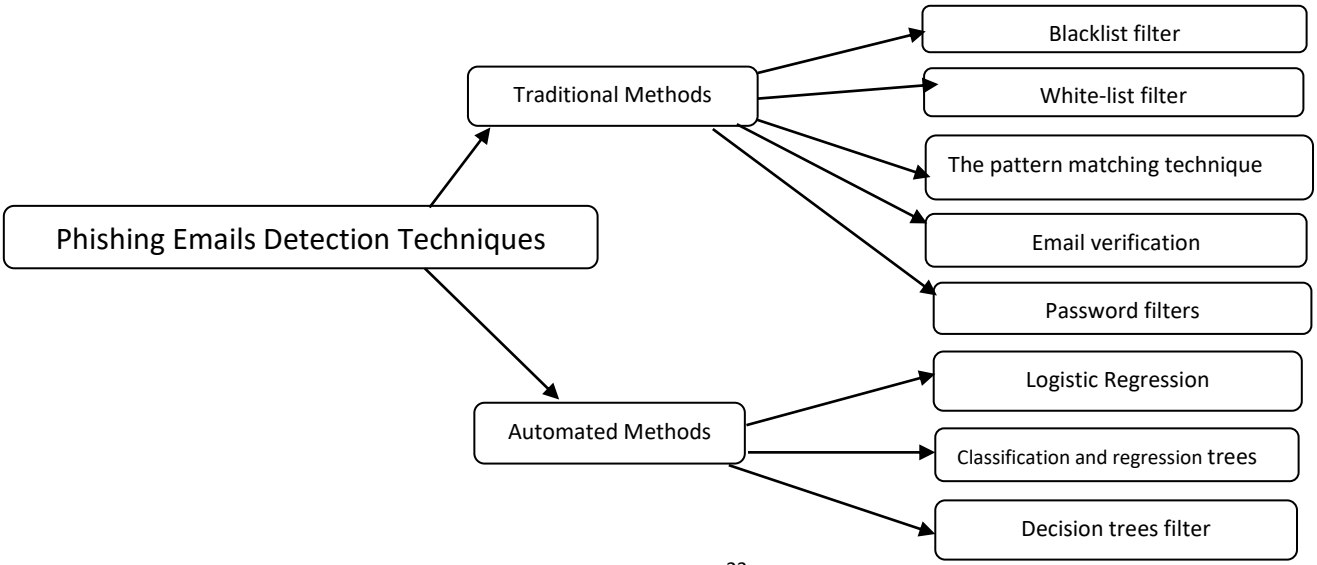


Figure 1-4 Life cycle of phishing email (Almomani et al., 2013)

1.6 Phishing Emails Detection Techniques

Experts develop extensive filters to predict and prevent phishing emails Rely on traditional techniques such as authentication protection or modern machine learning or data mining techniques to manage emerging threats.



1.6.1 Traditional methods

Traditional detection methods fall into two categories, network layer protection and authentication protection. The first type of protection at the network level includes blacklist filters and whitelist filters to prevent phishing by blocking suspicious IP addresses or domains access the web. In addition, there are pattern recognition filters and rule-based filters fixed detection rules based on manual input and updates (Ramanathan & Wechsler, 2012).

1.6.1.1 blacklist filter

Blacklist filtering technology provides network-level protection by classifying incoming email based on sender, IP or DNS address. These details are extracted from the email headers and compared to a predefined list, if any data matches the list; the email is rejected. The Internet Server Provider (ISP) is the organization responsible for providing and implementing this filter(Paass, 2009).

1.6.1.2 white-list filter

Whitelist filtering also provides network-level protection, but is different from blacklisting; this technique compares email data to a predefined list of static IP addresses and IP addresses of 10 legitimate domains (Cao, 2008). In this regard, only emails whose data matches the list are allowed access to the network to the user's inbox. Email addresses and IP addresses will be whitelisted if they belong to legitimate users or companies that have agreed to add their addresses to this list. Emails whose data matches this list are only classified as legitimate emails based on this filter, while other emails are considered phishing and blocked from accessing the network, this filter also acts as a legitimate email classifier.

1.6.1.3 The pattern matching technique

Pattern matching technology filters emails based on specific patterns (including words, text strings, and character sets) mentioned in the email body, subject, or sender. Filters search emails for these specified patterns to classify emails as phishing or legitimate. Although this technique provides network-level protection, it still provides some valuable and incorrect results due to the high volume of incoming emails that may contain prohibited words or text strings, but should not be prevented (Chhabra, 2005). The second category, authentication protection, provides user and domain level security. User-level protection requires users to provide authentication before sending messages. Verify email and password, while creating domain-level authentication protection for email servers (Ramathan, 2012).

1.6.1.4 Email verification

Email verification is a user-level authentication method that requires sender and recipient verification. Once the sender accepts the notification, the email is authenticated and deemed legitimate to the recipient's inbox. Otherwise, emails will be considered phishing and thus inboxes will not be accessible (Adida, 2006). This filter has its pros and cons. While this filtering process has been shown to be effective in fully (100%) detecting phishing emails, it takes a relatively long time as recipients must reply before receiving the email and there is a risk of email loss. The verification process generated traffic on the network or the same challenge was not detected.

1.6.1.5 Password filters

Protection is also provided through user-level authentication. This filter is used to receive any email in the subject line, email address, header field, or any part of the email only if the filter recognizes the retrieved password. Therefore, if the file manager cannot find the password or detects an incorrect password, the email will be rejected. So these passwords are not created by default; first-time users of this filter need to initiate a conversation with each other to set and activate passwords, and are then treated as legitimate users by the filter. The downside is that some legitimate emails may be lost if the password is not recognized, and the process also takes time (Ramanathan, 2012).

1.6.2 Automated Methods

The method applies an automatic classifier based on machine learning and data mining. The classifier works with the server and filters incoming emails as phishing or legitimate emails by checking various characteristics in the email header and body (Abu-Nimeh, 2007).

1.6.2.1 logistic regression

Logistic regression is a widely used method because of its easy interpretation and practical results. The model is useful in predicting binary data (0/1 responses) because it relies on statistics and Apply a generalized linear model. Although this method is simple, it has three disadvantages; First, it requires more statistical assumptions before it can be applied. second, more functions a variable with a linear relationship compared to a variable with a complex relationship. Finally, the accuracy and the prediction rate depends on the completeness of the data (Abu-Nimeh, 2007).

1.6.2.2 Classification and Regression Trees (CART)

The Classification and Regression Tree (CART) model developed in the 1980s was used to distribution of a tree split using two components and a T-tree split into two nodes A decision tree is repre-

sented by a series of yes or no questions into which the training sample is split into smaller parts. Unlike logistic regression methods, this model is used for complex relationships between variables rather than linear relationships. A binary tree is constructed by continuously dividing the prediction space into distinct homogeneous groups. Allocation depends on defines the partitioning rules associated with the internal nodes of the tree where each isomorphic group resides connected by an end node. This model results in a large binary tree, which provides easy-to-read interactions between predictors; it remains Due to its huge size, it is difficult to predict additive effects(Steinberg and Cora, 2009).

1.6.2.3 Decision Trees Filter

Decision tree filters are graphical classification models consisting of nodes and arrows. The base node is called the root of the originating decision tree. Each node in the network contains an "if-then" rule, a class and a feature, and leads to the next node via the indicated arrow as the edge. Decision trees end with leaf nodes called finalizers (Safavian, 2010)Various algorithms have been proposed to generate decision trees, including ID3 model, which computes entropy information as a heuristic function to evaluate the target. In this sense, a decision tree is generated in a subtree, every node in the tree has a parent leading to it (except the root node), each node results in a child node (except the leaf node), and the tree ends with a leaf node, which is the final solution to the problem posed.

1.6.2.4 Support vector machine(SVM)

Support vector machines are widely used by researchers in medical diagnosis, text classification, image classification, biological sequence analysis and other fields. This technique uses statistics, quadratic equations, and hard rules to separate data into two categories.By maximizing the bounding basis space on the kernel function by using a separating hyperplane, the data can be extracted and stored to reach the best solution.

1.7 Conclusion

Phishing attacks remain one of the major threats to individuals and organizations to date that has developed beyond obtaining sensitive information and financial crimes to cyber terrorism, hacktivism, damaging reputations, espionage, and nation-state attacks. In this chapter we introduce the phishing term and its types and we went further with one of it s types which is phishing emails. We presented it s life cycle and the techniques of detecting a phishing email . In the next chapter, we will present the related works in this domain.

Chapter 2 Deep Learning and literature review

In today's world, phishing is seen as a challenging threat growing rapidly every year. It is considered as a criminal act that integrates social-engineering and technical methods to steal confidential data of consumers such as usernames and passwords. In this chapter we will explain the deep learning, neural networks and present related research for detecting phishing emails.

2.1 Deep Learning

Deep learning is an AI-based machine learning technique that instructs computers to do tasks that falls naturally to mankind that is to learn by model. Deep learning is a crucial innovation behind automatic cars and autos empowering them to understand a stop signal or to differentiate between a person on foot and a street lamp post. It is the critical technology that enables voice control in gadgets like mobiles, laptops, Television, and headphones. In recent times, Deep learning is in the limelight because it is accomplishing results that were impractical earlier. With the help of deep learning, a computer-based model can self-learn classification tasks just like humans, directly from images, videos, texts, or voice. Deep learning models are capable of achieving superior accuracy that surpasses human-level results. These deep learning models are trained on an enormous amount of labeled data-set and neural architectures of multiple layers. Majority of deep learning techniques utilize neural networks, and therefore, deep learning models are also known as deep neural networks. The models that are built upon deep neural networks are trained on a massive amount of labeled data set, and the neural layers are capable of learning features directly from the data set without the need of manual feature extraction.

2.1.1 Artificial neural networks

Artificial neural networks(Guresen & Kayakutlu, 2011) are strongly connected networks of elementary processors operating in parallel. Each elementary processor calculates a unique output based on the information it receives.

Artificial neural networks, or neuromimetic networks, are models inspired by the functioning of the animal brain, and whose purpose is to see properties similar to the biological system arise.

they take up some main principles:

- **Parallelism:** neurons are entities performing a very simple function, but they are very strongly interconnected, which makes signal processing massively parallel.
- **Synaptic weights:** the connections between neurons have varying weights, which determine the strength of the interaction between each pair of neurons.
- **Learning:** these synaptic coefficients can be modified during learning, in order to make the network perform the desired function.

2.1.2 Structure of the neural network

Artificial neural networks are composed of elementary computational units called neurons combined according to different architectures. For example, they can be layered (multi-layer network), or have a connection topology. Layered networks consist of three layers as shown (Fig 1)

- **Input layer:** composed of n neurons (one for each input of the network).
- **Hidden layer:** made up of one or more hidden (or intermediate) layers made up of m neurons.
- **Output layer:** consisting of p neurons (one for each output of the network).

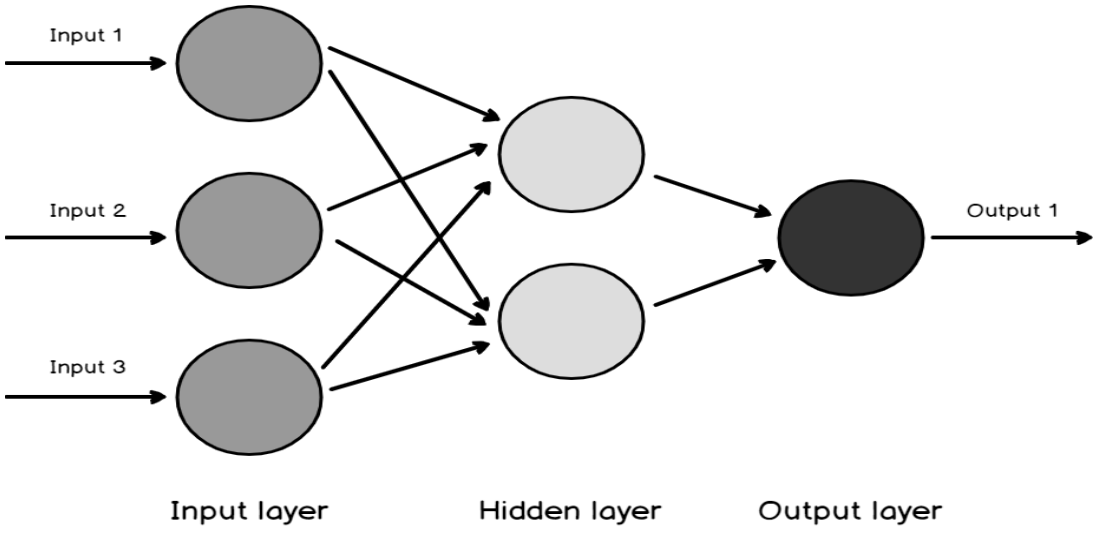


Figure 2-1 artificial neural network(Jensen et al., 1999)

2.1.3 Types of neural networks

The connections between the neurons that make up the network describe the topology of the model. It can be arbitrary, but more often some kind of regularity (with a fully connected network) can be distinguished. Different types of ANNs differ in connection type (network topology), choice of

transfer function (neuron type) and learning patterns (rules) associated with the network and how weights are estimated.

2.1.3.1 Recurrent Neural Network (RNN)

Is the most advanced algorithm for sequence data. It is the first algorithm to remember its inputs due to the internal memory, making it ideal for learning problems with automatic sequential data. It is one of the algorithms that has made amazing achievements in deep learning in recent years(Boualem & Meriem, 2021).

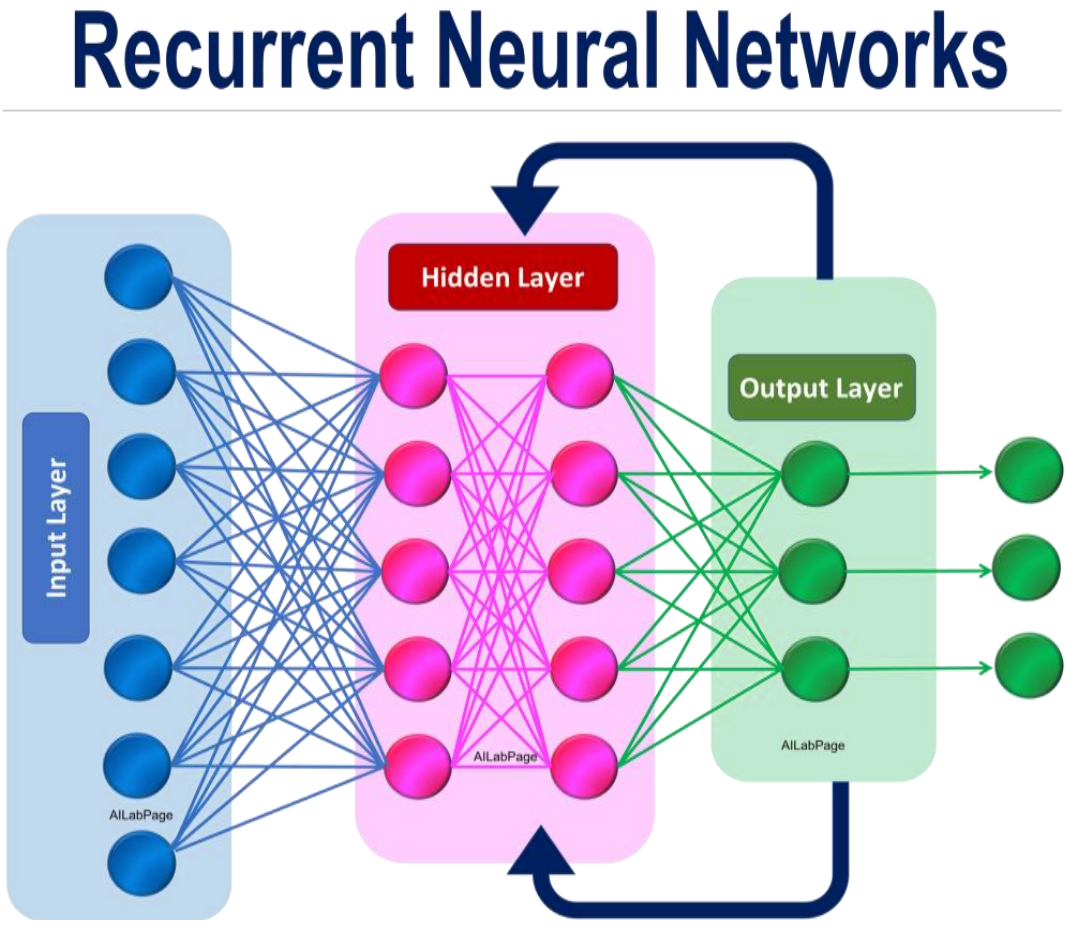


Figure 2-2 The Recurrent Neural Network (Boualem & Meriem, 2021)

2.1.3.2 convolutional neural network (ConvNet / CNN)

Is a deep learning algorithm that takes an image as input, assigns importance (learnable weights and biases) to different aspects/objects in the image, and distinguishes them from each other. ConvNet

requires much less preprocessing than other classification algorithms(Czum, 2020).

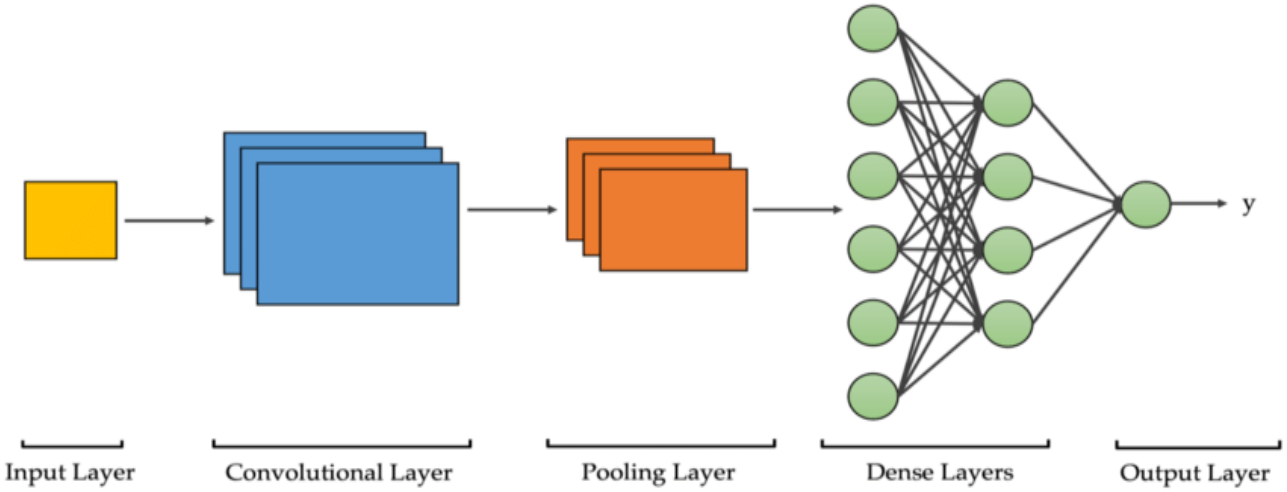


Figure 2-3 The convolutional neural network(Boualem & Meriem, 2021)

2.1.3.3 Long-term memory networks (LSTM)

Are a special type of RNN capable of long-term dependency learning, and they are explicitly designed to avoid long-term dependency problems. Remembering information for a long time is almost their default behavior. All RNNs take the form of chains of repeating neural network modules.

2.1.3.4 Bidirectional Recurrent Neural Network (BiLSTM)

In fact, it is to assemble two independent RNNs. This structure allows the network to have information upstream and downstream of the sequence at each time step. This approach differs from the one-way approach because in a reverse working LSTM you keep information from the future, and by combining two hidden states, you keep information from the past and future at all times.

2.2 Related Works

Several studies have been developed to detect phishing emails using different machine learning and deep learning approaches. Many novel features are introduced to filter phishing emails from legitimate emails. This section discusses various approaches, which were proposed by researchers to mitigate phishing emails.

Some of the studies that have been developed to detect phishing emails using machine learning approaches are:

(Akinyelu & Adewumi, 2014) used random forest machine learning algorithm in classification of phishing attacks on a data set consisting of 2000 phishing and ham emails. From this data a set of prominent phishing email features were extracted and used by the machine learning algorithm with a resulting classification accuracy of 99.7% and low false negative (FN) and false positive (FP) rates.

Another study made by (Yasin & Abuhasan, 2016) who applied a model that utilizes 23 hybrid features of the email header and body extracted from about 10000 emails divided equally between legitimate and phishing emails. They compared the predictive accuracy of several machine learning methods including MLP, Random forest, Bayes net and SVM . Their research showed that Random Forest algorithm gave the best result with an accuracy of 99.1%.

Another study made by (Ahamid et al., 2013) who proposed behavior-based features to detect phishing emails by observing sender behavior and used it on 5 sets of datasets randomly containing varying split percentage number of phishing and ham emails from the overall datasets. In order to treat the set equally, they fixed the number for each sets to 2000 data. By combining these datasets, they used Bayes Net algorithm to classify the datasets into phishing or ham emails. This hybrid feature selection approach produce promising result using 8 features with 94% accuracy.

(Alsufyani & Alzahrani, 2021) proposed to use natural language processing (NLP) along with machine learning techniques for text phishing detection . They started with 6,224 emails from an existing dataset that contains both phishing and legitimate emails. NLP was used for preparing the data before extracting features from it and using the features for training the classification models by machine learning algorithm and for testing these models. The features were extracted using the Continuous Bag of Words (CBOW) in the Word2Vec algorithm. They trained four models using four different machine learning algorithms which are k-nearest neighbors (KNN) that gave accuracy of 93%, Multinomial Naive Bayes (MNB) that gave accuracy of 80% , Decision Tree that gave accuracy of 90% and AdaBoost that gave accuracy of 92%.

Some of the studies that have been developed to detect phishing emails using deep learning approaches are:

One of the interesting methods titled THEMIS was proposed by (Fang et al., 2019). Based on an improved recurrent convolutional neural networks (RCNN) model with multilevel vectors and attention mechanism. They evaluated the accuracy on a set of 999 phishing emails and 7781 legitimate emails. The model gave an average accuracy rate of 99.848%,false positive rate of 0.043, average precision of 99.664%, average recall of 99% and average F-score of 99.331%.

(AbdulNabi & Yaseen, 2021) proved the effectiveness of word embedding in classifying spam emails by using Pre-trained transformer model BERT which uses attention layers to take the context of the text into its perspective. They compared the results to a baseline DNN model that contains a BiLSTM layer and two stacked Dense layers. In addition they compared the results to a set of classic classifiers k-NN and NB (Naive Bayes). Two open-source data sets were used, one to train the

model and the other to test the persistence and robustness of the model against unseen data. The proposed approach attained the highest accuracy of 98.67% and 98.66% F1 score.

Another work made by (Nguyen et al., 2018) who presented a framework with hierarchical long short-term memory networks (H-LSTMs) and supervised attention to model the emails simultaneously at the word and the sentence level. In their work they used two datasets, data-no-header that contains about 5721 emails and data-full-header which contains about 4585 emails. However their proposed approach attained the highest Precision of 99%, Recall of 99.2% and 99.1% F1 score using data-full-header and a precision of 98.1%,Recall of 97,9% and 97.9% F1 score using data-no-header.

(Li et al., 2020) proposed a LSTM based phishing detection method that includes two important stages, sample expansion stage where they combined KNN with K-Means to expand the training data set, so that the size of training samples can meet the needs of in-depth learning and testing stage where they first preprocess these samples ,including generalization, word segmentation and word vector generation. Then, the preprocessed data that contains about 8000000 of both phishing and ham emails is used to train a LSTM model and they classified the phishing emails. Experiments have been conducted to test the performance of the proposed method, and results show that our phishing detection method can reach a 95% accuracy rate.

(Hiransha et al., 2018) created a model based on Keras Word Embedding and Convolutional Neural Network and used it on two kind of datasets. The first one contains about 4583 emails with no header and the second one contains about 4082 emails with header. After the classification of the data for sub task 1 in which the emails didn't had header files the proposed model gave an accuracy of 96.8% and for sub task 2 in which header files were given their model gave an accuracy of 94.2 %.

Another study made by (Saha et al., 2020) who presents a data-driven framework for detecting phishing webpages using deep learning approach. More precisely, a multilayer perceptron, which is also referred as a feed-forward neural network is used to predict the phishing webpages and applied it on a dataset that was collected from Kaggle and contains information of ten thousand webpages and achieved an accuracy of 95% for training accuracy and 93% for test accuracy.

Another study made by (Hina et al., 2021) who proposed a novel efficient approach named Se-FACED that uses Long Short-Term Memory (LSTM) based Gated Recurrent Neural Network (GRU) for multiclass email classification which achieved 95% in accuracy,precision,recall and f1-score.

Below you find a table that summarize all related works:

Research	Classification Method	Dataset	Results
(Akinyelu & Adewumi, 2014)	Random forest	The ham corpora provided by spam assassin project The publicly available phishing corpus provided by Nazario	Accuracy :99.7% FN rate :2.50% FP rate of 0.06% Precision : 99.47% Recall : 97.50%
(Yasin & Abuhasan, 2016)	J48 Naïve Bayes SVM Multi-Layer Perceptron Random Forest	Data set consists of 10538 emails including 5940 ham emails from spam assassin project and 4598 spam emails from Nazario phishing corpus.	F-Measure : 0.984 F-Measure : 0.945 F-Measure : 0.969 F-Measure : 0.991 F-Measure : 0.977
(Ahamid et al., 2013)	Bayes Net algorithm	5 sets of datasets randomly containing varying split percentage number of phishing and ham emails from the overall datasets	Accuracy: 93%
(Alsufyani & Alzahrani, 2021)	KNN MNB Decision Tree AdaBoost.	They collected real data from as wide a range of sources as possible to create a varied dataset, and also created artificial data. The dataset contains two types of emails, which are full header email messages and no-header email messages and consists of 9172 legitimate messages and 1132 phishing emails.	Accuracy: 93% Accuracy:80% Accuracy:90% Accuracy:92%
(Fang et al., 2019)	THEMIS	The data was provided by IWSPAwhich contains two dataset: the experimental data which comes from the First Security and Privacy Analytics Anti-	**Accuracy**: 99.848% **FPR** :0.043% **recall** : 99.000% **precision** :99.664%

		Phishing Shared Task. The sources of the legitimate email include email collections from Wikileaks archives, such as the Demo cratic National Committee, Hacking Team, Sony emails, *etc*. There are also selected emails from the Enron Dataset and SpamAssassin. As for the phishing emails,they mainly come from the Information Technology (IT) departments of different universities	**F1-score** :99.331%
(AbdulNabi & Yaseen, 2021)	KNN NB BILSTM BERT	The first data set is the open source Spambase data set from the UCI machine learning repository the data set contains 5569 emails, of which 745 are spam. The second data set is the open source Spam filter data set from Kaggle which contains 5728 emails of which 1368 are spam.	**Accuracy** :0.9310 **F1-Score** :0.9081 **Accuracy :** 0.9540 **F1-Score** : 0.9408 **Accuracy :**0.9650 **F1-Score** :0.9556 **Accuracy :**0.9730 **F1-Score** :0.9696
(Nguyen et al., 2018)	H-LSTM H-LSTMs+supervised	The data was provided by IWSPAwhich contains two dataset: the first dataset involves emails that only have the body part while the second dataset contains emails with both bodies and headers	**Precision:** 0.9638 **Recall :** 0.9448 **F1:** 0.9542 **Precision:** 0.9784 **Recall:** 0.9466 **F1:** 0.9621
(Li et al., 2020)	LSTM	they collected emails from their email server, and mailbox data from some companies and organi-	**Accuracy:96%** **Precision:95%** **Recall:93%**

		zations as their experimental data.	**F1-Score:94%**
(Hiransha et al., 2018)	Word Embedding + CNN (Sub task1 no header) Word Embedding + CNN (Sub task2 with header)	Two sets of data sets were given one with header files for Task 1, having from, to addresses and one without header for Task 2, only the matter. For training data set, total number of 4,583 mails were given for Task 1 in which 4,082 were legitimate and 501 were phishing. For Task 2,total of 5,700 mails were given in which 5,088 were legitimate while 612 were phishing. For test data set total of 4,195 emails were given for Task 1 and 4,300 were given for Task 2.	**Accuracy:96.8%** **Accuracy:94.2%**
(Saha et al., 2020)	presents a data-driven framework for detecting phishing webpages using deep learning approach. More precisely, a multilayer perceptron, which is also referred as a feed-forward neural network is used to predict the phishing webpages	Data were collected from the website www.kaggle.com for this experiment. The website contains more than ten thousand phishing websites' information of numerous features.	**training accuracy:** 95% **test accuracy**: 93%
(Hina et al., 2021)	LSTM+GRU	The dataset used in this study is an amalgamation of four different datasets. The dataset contains Normal e-mails from Enron Corpora , Fraudulent e-mails provided by Phished e-mails corpora which contain misleading information,Harassment messages selected from Hate Speech, Offensive dataset.	**Accuracy**: 95% **recall** : 95% **precision** :95% **F1-score** :95%

		They enhance the dataset of Email Forensics by adding the suspicious emails data from our email sources,and twitter source.

Table 1 Abstract of related works

2.3 Conclusion

In this chapter, we presented the deep learning and different neural netwoks and some previously published works on phishig detection. We go through their guiding principles or architectures, as well as their outcomes. In the next chapter we are gonna present our proposed solution.

Chapter 3 The proposed solution

In order to achieve our goal of detecting phishing emails, we followed six main steps. Since the extracted data is textual data, we need to perform natural language processing cleanup and preprocessing steps. Then, we focus on the exploratory analysis data and feature extraction based on datasets. Once the features are selected and extracted, we select, train and evaluate classifiers to make predictions. In the following sections, we describe in detail the use of each step.

3.1 Methodology

We propose an approach to retrieve the semantically similar emails. As depicted in Figure 1, our approach is composed of six main modules. The first step corresponds to data acquisition. In the second step, we apply the preprocessing step on the data(cleaning the data, removing stop words and tokenization). In the third step, we transform data into another form so that we will be able to apply our model(siamese). In the fourth step we convert text to features using the word embedding (Word2vec). Then the word vectors of the emails are fed to the Siamese MALSTM to represent them in final hidden state encoding semantic meaning of the emails and calculate the similarity between emails and shows if they belong to the same class or not .

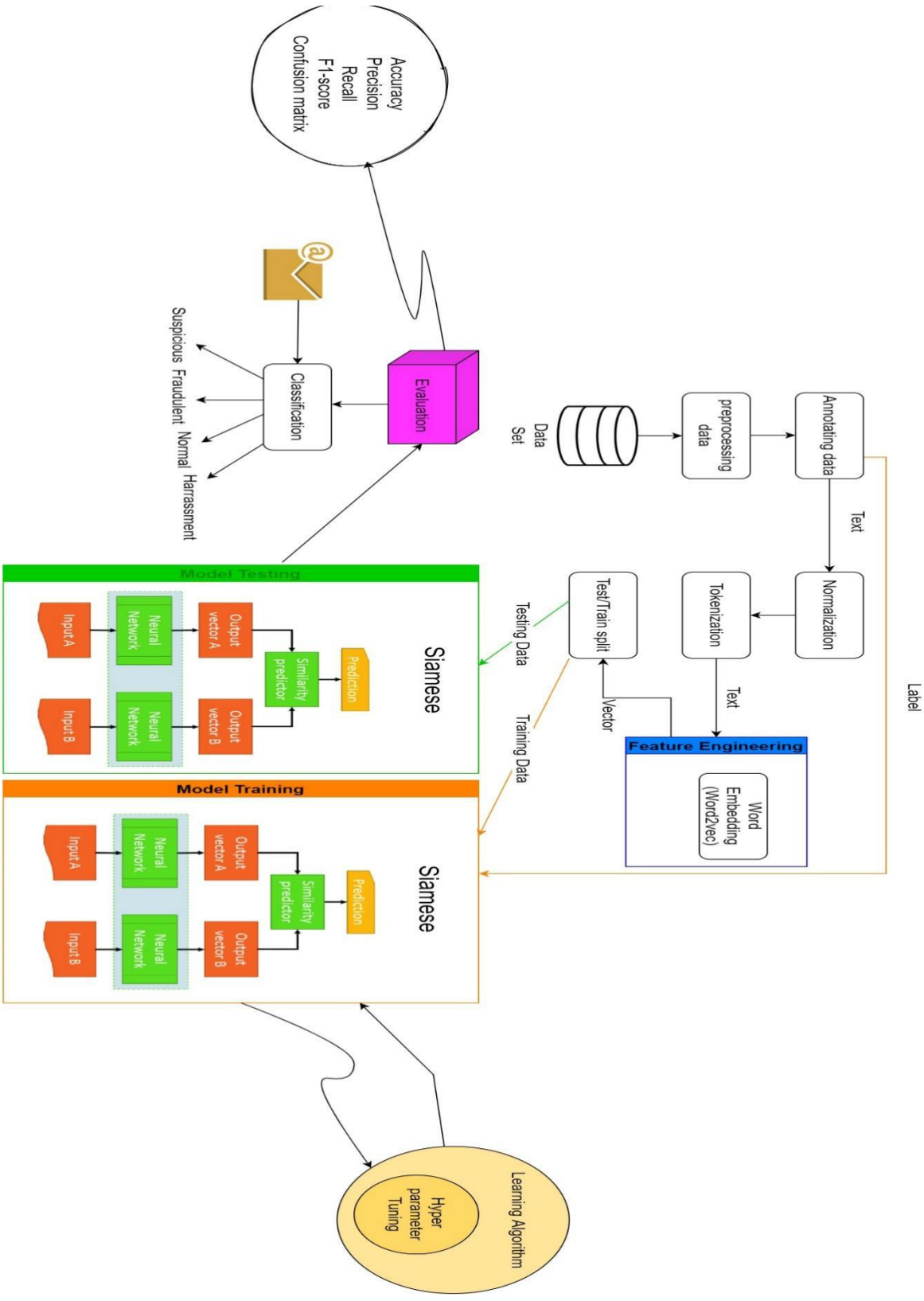


Figure 3-1 The proposed model

Algorithm 1: Multi-Class E-Mail Classification Using siamese

1: INPUT: Data ← E-mail Messages

2: OUTPUT: Normal, Harassing, Suspicious, Fraudulent

3: For each E-mail message E {Data Preprocessing}

4: Remove (tabs, punctuation, stopword, numbers, whitespaces from E)

5: For each E-mail message E {Tokenization}

6: For each token in E create a vector(word2vec)

7: **Calculating similarity using siamese network**

8: For each epoch do

9: Evaluate Loss, Validation Loss Evaluate Accuracy

10: end for

11: Evaluate Precision, Recall, F-score and Confusion Matrix

3.2 Preprocessing a text document

Data preprocessing is an integral part of the data mining process. Real world data is collected using different methods and is not specific to a particular domain, resulting in incomplete, unstructured, and unreliable data containing errors. Such data leads to irrelevant and erroneous predictions if analyzed directly.

In our framework, various methods are used during the preprocessing phase:

3.2.1 Clean up the text

The data cleaning process NLP is crucial. The computer doesn't understand the text. For the computer, it is just a cluster of symbols. To further process the data we need to make the data cleaner.

The goal of this method is to remove patterns, e.g. "characters, symbols, and numbers other than alphabets"; "empty strings"; "drop rows with NaN in the column"; "duplicate rows" etc.

For cleaning up our dataset we used regex which is a method where you specify the rules for the set of possible strings that you want to match.

```python
import re

# Clean text function
def text_clean(text):
    ''' Pre process and convert texts to a list of words '''
    text = str(text)
    text = text.lower()
    text = re.sub(r'(\\u[0-9A-Fa-f]+)',r"", text)
    text = re.sub(r'[^\x00-\x7f]',r" ",text)
    text = re.sub('((www\.[^\s]+)|(https?://[^\s]+))','URL',text)
    text = re.sub('@[^\s]+','AT_USER',text)
    text = re.sub(r'[^\w]', ' ', text)
    text = re.sub(r'#([^\s]+)', r'\1', text)
    text =  text.replace(':)',"")
    text = "".join([i for i in text if not i.isdigit()])
    text = re.sub(r"(\!)\1+", ' ', text)
    text = re.sub(r"(\?)\1+", ' ', text)
    text = re.sub(r"(\.)\1+", ' ', text)
    text = re.sub(':\)|;\)|:-\)|\(-:|:-D|=D|:P|xD|X-p|\^\^|:-*|\^\.\^|\^\-\^|\^\_\^|\,-\)|\)-:|:\'\(|:\(|:-\(|:\S|T\.T|\.\_\.|:<|:-\S|:-<|\*\-\*|:O|=O|=\-O|O\.o|XO|O\_O|:-\@|=/|:/|X\-\(|>\.<|>=\(|D:', "", text)
    text = re.sub(r"[^A-Za-z0-9^,!.\/'+-=]", " ", text)
    text = re.sub(r"what's", "what is ", text)
    text = re.sub(r"\'s", " ", text)
    text = re.sub(r"\'ve", " have ", text)
    text = re.sub(r"can't", "cannot ", text)
    text = re.sub(r"n't", " not ", text)
    text = re.sub(r"i'm", "i am ", text)
    text = re.sub(r"\'re", " are ", text)
    text = re.sub(r"\'d", " would ", text)
    text = re.sub(r"\'ll", " will ", text)
    text = re.sub(r",", " ", text)
    text = re.sub(r"\.", " ", text)
    text = re.sub(r"!", " ! ", text)
    text = re.sub(r"\/", " ", text)
```

Below we present the five first lines of our data set before and after the preprocessing.

	Text	Class_Label
0	Let me know when you get the quotes from Pauli...	Normal
1	---------- Forwarded by Phillip K ...	Normal
2	Steve, Please remove Bob Shiring and Liz River...	Normal
3	Go ahead and order the ac for #27. Can you ema...	Normal
4	Anymore details? Is the offer above or below 6...	Normal

Figure 3-2 Data set before preprocessing

	Text	Class_Label
0	let me know when you get the quotes from pauli...	Normal
1	forwarded by phillip k allen hou ect on pm er...	Normal
2	steve please remove bob shiring and liz rivera...	Normal
3	go ahead and order the ac for can you email or...	Normal
4	anymore details is the offer above or below wh...	Normal

Figure 3-3 Data set after preprocessing

3.2.2 The removal of stop words

Once the clean up process has been finalized, we note that the document presents certain tokens called "stop words" such as: articles (the, this ...), prepositions (of, for, with ...), determinants (my, the, these ...), adverbs (before, in front of, righthere ...). These stop words, have practically no impact on the meaning of the text, their removal makes it possible to reduce the number of characters to be treated and consequently the time. Below we present the first five lignes of our data set after removing the stop words.

	Text	Class_Label
0	let know get quotes pauline expecting pay some...	Normal
1	forwarded phillip k allen hou ect pm eric bens...	Normal
2	steve please remove bob shiring liz rivera rc ...	Normal
3	go ahead order ac email fax summary rents coll...	Normal
4	anymore details offer else clean cap good loca...	Normal

Figure 3-4 Data set after removing the stop words

3.2.3 The tokenization

The final step in processing text is to split the stream of characters into words, or more specifically, tokens. This is the basis for further analysis. It's hard to imagine extracting information from a doc-

ument without identifying the token. However, the frequency of occurrence of the token must always be referenced. However, computer programs would find this task more complicated. The reason is that some characters are sometimes token delimiters and sometimes not, depending on the application. Spaces, tabs, and newlines do not count as tokens. They are often collectively referred to as blanks (Kulkarni & Shivananda, 2019).

3.3 Data Transforming

The data set used contains two columns and a label with four classes fraudulent, harassment, suspicious and normal in our case we transformed the problem from classification into similarity detection and the data obtained contains three columns two for the texts and one for the label. We created pairs of data for siamese training label 1 if pairs from same class otherwise 0.Below we present the first five lignes of our new data set and the code to realise it.

```
d1 = list(data[data['Class_Label'] ==1]['Text'])
d2 = list(data[data['Class_Label'] ==2]['Text'])
d3 = list(data[data['Class_Label'] ==3]['Text'])
d4 = list(data[data['Class_Label'] ==4]['Text'])
d11 = d1[:1000]
d22 = d2[:1000]
d33 = d3[:1000]
d44 = d4[:1000]
import random
# Creating pairs of data for siamese training => label 1 if pairs from same class otherwise 0
df2 = pd.DataFrame(columns=['text1', 'text2', 'label'])

for data in d11:
  data1 = data
  data2 = random.choice(d11)
  data3 = random.choice(d22)
  data4 = random.choice(d33)
  data5 = random.choice(d44)

  df2.loc[len(df2)] = [data1, data2, 1]
  df2.loc[len(df2)] = [data1, data3, 0]
  df2.loc[len(df2)] = [data1, data4, 0]
  df2.loc[len(df2)] = [data1, data5, 0]
```

	text1	text2	label
0	let know get quotes pauline expecting pay some...	jacques amounts needed fill blanks exhibit b f...	1
1	let know get quotes pauline expecting pay some...	gracefield consult financial consultants notar...	0
2	let know get quotes pauline expecting pay some...	user cunt	0
3	let know get quotes pauline expecting pay some...	explore pussy user url	0
4	forwarded phillip k allen hou ect pm eric bens...	message dated pm eastern standard time mccormi...	1

Figure 3-5 The data set obtained after the transformation

3.4 Word embeding (word2vec)

Natural language texts cannot be directly interpreted by a classification algorithm. To do this, it is necessary to use an efficient rendering technique that allows the texts to be rendered in a machine-usable form. Word embedding is a technical technique to represent words as a vector of real numbers through learning using a set of words, which facilitates the semantic analysis of words. Word embedding is a method of dealing with a recurring problem in artificial intelligence, namely that of dimension. In fact, the representation of words with traditional methods (bag of words) is done with a vector that contains the entire dictionary. On the other hand, the word embedding technique decreases the number of these dimensions, thus facilitating learning tasks involving these words(Kulkarni & Shivananda, 2019).

3.4.1 Word embedding learning techniques

There are mainly two word embeddings techniques according to (Naili et al., 2017):

Continuous Bag of Words (CBOW): which trains the neural network to predict a word according to its context, ie words before / after in a sentence. In the CBOW process, three layers are used: the input layer corresponds to the context, the hidden layer corresponds to the projection of each word

of the input layer in the weight matrix which is itself projected on the third layer which is the output layer. This is illustrated in Figure 6.

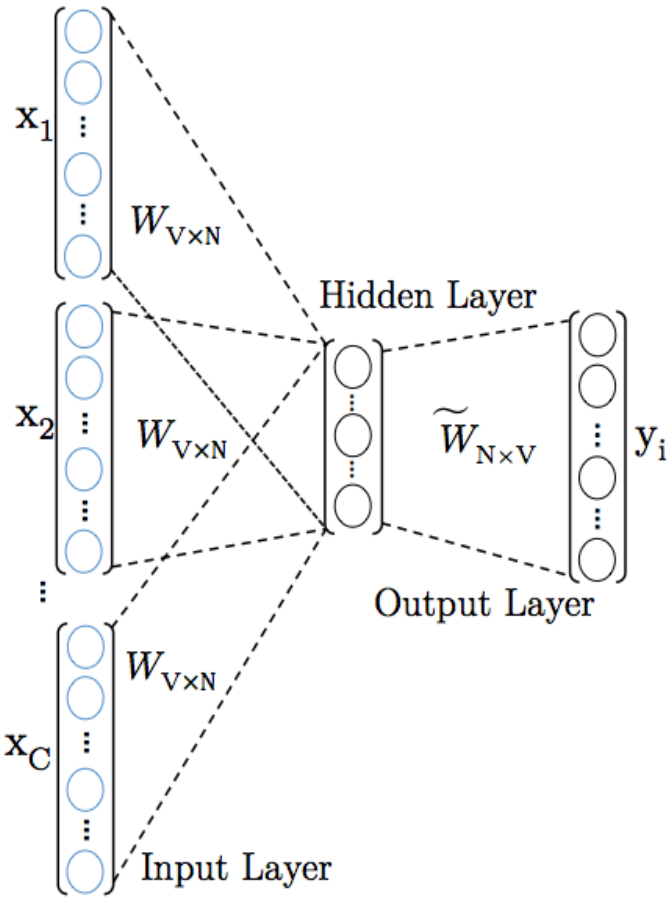


Figure 3-6 CBOW Neural Network (Russac et al., 2018)

SkipGram technique: it is the opposite of the CBOW model: the model tries to predict the context based on the word. Indeed, the input layer corresponds to the target word and the output layer corresponds to the context. Thus, Skip-Gram seeks context prediction given a word instead of word prediction given its context as is the case for CBOW as we can see in the picture below. This is illustrated in Figure 7.

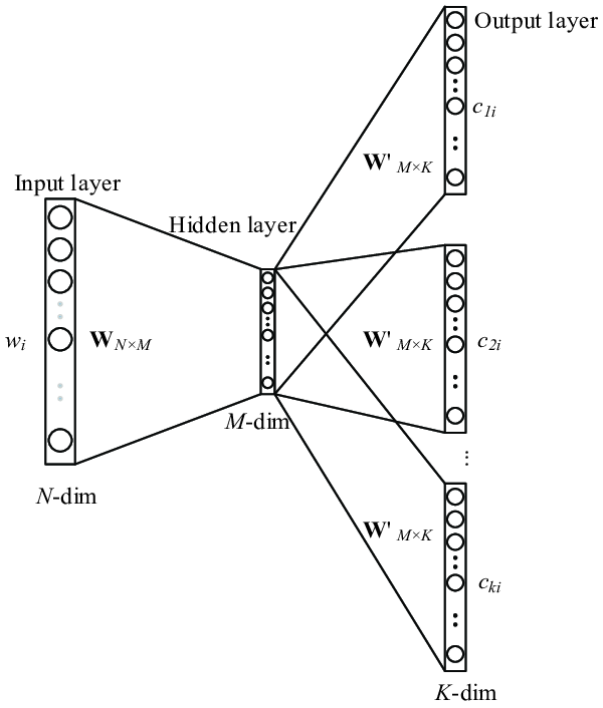


Figure 3-7 Skip-Gram Neural Network(Hu et al., 2018)

word2vec: Word2vec is a two-layer neural network that processes text documents. Its input is a corpus of text, and its output is a set of feature vectors for words in that corpus. While Word2vec is not a deep neural network, it converts text into a numerical form that a deep network can understand. The purpose and use of Word2vec is to group vectors of similar words in a vector space. In other words, it mathematically identifies similarities. Word2vec creates vectors, which are distributed numerical representations of word units, properties such as the context of individual words; this all happens without expert intervention(Kulkarni & Shivananda, 2019).

In our case we are going to use the CBOW technique to represent our text as a vector. Below we present the first five lignes of our data set after applying word2vec.

	text1	text2	label
0	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 4, 13,...	[43, 44, 45, 46, 47]	1
1	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 4, 13,...	[48, 49, 50, 48, 51, 52, 53, 54, 55, 56, 57, 5...	0
2	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 4, 13,...	[64, 66, 67, 68, 69, 70]	0
3	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 4, 13,...	[71, 71, 72, 73, 74]	0
4	[75, 42, 76, 77, 78, 79, 80, 81, 82, 64, 42, 7...	[92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102...	1

Figure 3-8 Data set after applying word2vec

3.5 The principle of the deep learning algorithm siamese

Siamese is an artificial neural network that processes two or more input vectors side by side and combines the output vector after subidentical neural network computation. The input provided to the siamese network can be in any form, such as numeric, image, or text data. The Siamese network is useful for various tasks that require discovering the relationship between two patterns, such as semantic similarity identification of sentences, forged signature recognition, pattern recognition, and paraphrase identification. Similar entries are processed with subidentical network models. Subnets extract features from inputs that are similar and comparable. The Siamese network applies binary classification on the output, which indicates whether the inputs are of the same class or not. If the entries belong to the same class, it means that they are somehow identical to each other and are considered duplicates. While binding to the output of the processed inputs, the neuron measures the distance between two feature vectors(Feng & Lu, 2019).

MALSTM : Long Short-Term Memory (LSTM) (Othman et al., 2019), which is a powerful type of RNN used in deep learning, has gained wide attention in recent years owing to its capacity to capture long-term dependencies and model sequential data.

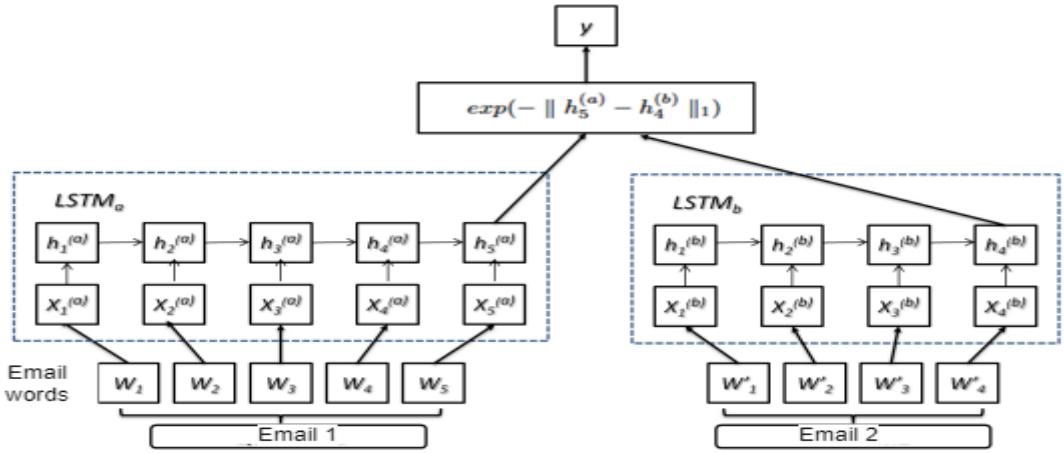


Figure 3-9 General architecture of the MaLSTM model (Othman et al., 2019)

Manhattan LSTM (MaLSTM) refers to the Manhattan distance used to compare the final hidden state of two standard LSTM layers, not another distance such as cosine and Euclidean distance. The goal of MaLSTM is to compare a pair of sentences to determine if they are semantically equivalent. MaLSTM uses the Siamese network architecture(Othman et al., 2019) , which is known to have the same sub-networks LSTMleft and LSTMright, which pass vector representations of two sentences and return hidden states that encode sentence semantics. These hidden states are then compared using a similarity measure to return a similarity score, as shown in Figure 8. In our work, MaLSTM is adapted to the context of the email. The LSTM learns a mapping from the space of variable-length sequences and encodes the input sequence into a fixed-dimensional hidden state representation . In other words, each email represented as a sequence of word vectors (e.g. Email 1 is represented by x1, x2, x3) is fed into the LSTM, which updates its hidden state at each sequence index. The final state of the LSTM for each email is a drep-dimensional vector, labeled h in Figure 8, which contains the semantic meaning of the email between sequence pairs. A main feature of the Siamese architecture is the shared weights across the subnetworks, which reduce not only the number of parameters but also the tendency of overfitting. MaLSTM uses the Siamese structure along with the Manhattan distance, hence the name MaLSTM model. Once we have the two vectors that capture the underlying meaning of each email, we calculate the similarity between them using the following Manhattan similarity function:

$$Y=\exp(-\| h(left)-h(right) \| 1)$$

Equation 1 similarity function

Note that since we have an exponent of a negative, the Manhattan function scores will be between 0 and 1.

3.6 Conclusion

In this chapter we have presented our proposed model and we explained in details every step we have been through. In the next chapter we are going to present the evaluation matrix that we have used and the results we got.

Chapter 4 Test and results

In this chapter, we present a comparative study between the detections of phishing emails, based on the basic algorithm (Siamese) using Phished emails corpora data-set. On the basis of the results obtained, we will proceed to an interpretation of the results and finally to the selection of the best algorithm among the algorithms studied for the classification of the emails.

For this, we first present a description of the resources used, the development tools and software as well as the methodology adopted and the different stages carried out in the experimentation process.

4.1 Data Set

The dataset used in this study is a merge of four different datasets. The dataset includes regular emails from Enron Corpora (The Enron Email Dataset, 2020), fraudulent emails provided by the Phishing Email Corpora (D. Radev, 2008) containing misleading information, harassment messages, selected from the Hate Speech, Offensive dataset. They extend our email forensics dataset by adding suspicious email data from email feeds and Twitter feeds. Suspicious records contain some horrific information that Twitter collects through the API. These different datasets were combined into a single structure file to enable multi-class email classification(Hina et al., 2021). TABLE 2 shows the composition of different E-mail corpus used for this study.

	Normal Emails	**Fraudulent Emails**	**Harrasement Emails**	**Suspicious Emails**
Number	9001	9001	9138	5287
Percentage	27.8%	27.8%	28.2%	16.3%

Table 2 Composition of Dataset

4.2 Experiments set up:

4.2.1 Word Embedding Learning

For word embedding training, we used the CBOW model, since it has proven through experiments to be more efficient and performs better than Skipgram with sizeable data.

The training parameters of the CBOW model were set after several tests:

– Size=300: feature vector dimension. We tested different values in the range [50, 500] but did not get significantly different precision values. The best precision was achieved with size=300.

– min-count=1: minimum number of words which we set to 1 to make sure we do not throw away anything.

– Context window=5: fixed window size. We tested different window sizes . The best accuracy was obtained with window equals 5.

```
import gensim
from gensim.models import Word2Vec
word2vec_model = Word2Vec(mes, size=300, window=5, min_count=1, workers=16)
print(word2vec_model)
```

4.2.2 Siamese Training

For Siamese training, we applied the sgd method for weights optimization to automatically decrease the learning rate. Gradient clipping was also used with a threshold value of 1.25 to avoid the exploding gradient problem . Our LSTM layers' size is 50 and embedding layer's size is 300. We used the back propagation and small batches of size equals 64, to reduce the cross-entropy loss and we resorted to the Mean Square Error (MSE) as a common regression loss function for prediction. We trained our model for several epochs to observe how the results varied with the epochs.

```
# Model variables
n_hidden = 50
gradient_clipping_norm = 1.25
batch_size =64
n_epoch = 120

def exponent_neg_manhattan_distance(left, right):
    ''' Helper function for the similarity estimate of the LSTMs outputs'''
    return K.exp(-K.sum(K.abs(left-right), axis=1, keepdims=True))

# The visible layer
left_input = Input(shape=(max_seq_length,), dtype='int32')
right_input = Input(shape=(max_seq_length,), dtype='int32')

embedding_layer = Embedding(len(embeddings), embedding_dim, weights=[embeddings], input_length=max_seq_length, trainable=False)

# Embedded version of the inputs
encoded_left = embedding_layer(left_input)
encoded_right = embedding_layer(right_input)

# Since this is a siamese network, both sides share the same LSTM
shared_lstm = LSTM(n_hidden)

left_output = shared_lstm(encoded_left)
right_output = shared_lstm(encoded_right)

# Calculates the distance as defined by the MaLSTM model
malstm_distance = Lambda(function=lambda x: exponent_neg_manhattan_distance(x[0], x[1]),output_shape=lambda x: (x[0][0], 1))([left_output, right_output])
```

4.3 Work environment and tools

4.3.1 Equipment :

Operating System	Windows 10 64bit
RAM	16Go
Processor	Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 2.59 GHz

Colab : is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members - just the way you edit documents in Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook.

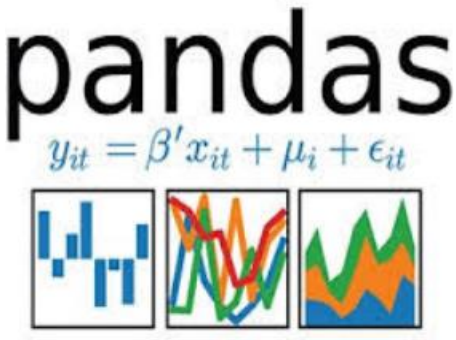
4.3.2 The choice of programming language

To choose a programming language that specializes in machine learning, and image processing, it must consider the skills listed in current job postings as well as the libraries available in different languages that can make the learning process deep. Python is the language of most affected programming in machine learning and deep learning.

Python

Python is a general-purpose programming language for high-level interpretation of programming languages. Developed by Guido van Rossum and first published in 1991, the Python design philosophy emphasizes code readability through the prominent use of large whitespace (Joseph Johnson, 2019). Its language constructs and object-oriented approach are designed to help programmers write clear and logical code for small and large projects, and it provides structures that enable clear programming on both small and large scales. Python has a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional, and procedural, and has a large and comprehensive standard library. Python interpreters are available for many operating systems.

4.3.3 Libraries used



Pandas: Pandas is an open source library licensed under BSD that provides powerful and easy-to-use data structures and data analysis tools for the Python programming language. Pandas is a NumFOCUS sponsored project. This will help Pandas successfully grow into a world-class open source project and allow donations to the project. Pandas is a Python library that allows you to easily manipulate data for analysis:

• Edit data tables with variable labels (columns) and people (rows).

• These tables are called DataFrames, similar to DataFrames under R.

• You can easily read and write these DataFrames from spreadsheet files.

• Thanks to Matplotlib, we can draw graphs from these DataFrames.

Matplotlib :

Matplotlib is a Python programming language library for plotting and visualizing data in the form of graphs. It can be used in conjunction with the scientific Python libraries NumPy and SciPy. Matplotlib is distributed for free under a BSD4-style license. The current stable version (2.0.1 in 2017) is compatible with Python version 3. There are a few things that make this library interesting:

• Export to various raster (PNG, JPEG...) and vector (PDF, SVG...) formats

• Extensive online documentation with many examples available on the Internet Strong and very active community

• Pylab interface: faithful reproduction of MATLAB syntax

• Advanced library: ideal for interactive computing

Nltk :

Natural Language Toolkit (NLTK) is a Python software library for automatic language processing developed by Steven Bird and Edward Loper of the Department of Computer Science at the University of Pennsylvania. In addition to the library, NLTK provides graphical demos, sample data, tutorials, and application programming interface (API) documentation.

scikit-learn :

Scikit-Learn (SKLearn) is an environment built into the Python programming language. This library provides various supervised algorithms suitable for this project (Rusland et al., 2017). This library provides high-level implementations to train "fit" methods and "predictions" from estimators (classifiers). It also provides to perform cross-validation, feature selection, feature extraction and parameter optimization (Saad et al., 2012).

Numpy :

Numerical Python provides an interface for storing and performing data operations. In a way, Numpy tables are similar to lists in Python, but Numpy makes operations more efficient, especially on large tables at the heart of the data science ecosystem (Jonathan A. Zdziar-ski., 2005).

Gensim :

Gensim is a Python library for topic modeling, document indexing, and similarity search on large corpora. The target audience is the natural language processing (NLP) and information retrieval (IR) communities.

Tensorflow :

TensorFlow is a Python library for fast numerical computation created and published by Google. It is a base library that can be used to build deep learning models directly or use wrapper libraries to simplify the process of building on top of TensorFlow.

Keras :

Keras is a powerful and easy-to-use free open source Python library for developing and evaluating deep learning models.

4.4 Evaluation metrics

A prediction model may be satisfactory, but it is not perfect. Choosing a competent phishing emails detection method requires performance evaluation and comparison of the best result generated. To do this, some performance estimation measures have been established.

Among these measures are the following:

4.4.1 Accuracy Report

The research was aimed at finding the highest accuracy for detecting the emails. The module from the Scikit-learn library called 'Accuracy' helped analyse the correct number of emails classified as 'normal' , 'suspicious' , 'Harrassment' and 'fraudulent'. This can be measured by equation- :

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

Equation 2 Accuracy Formula

Evaluating the Dataset for training and testing data to provide better accuracy and showed improvement. This could vary on the dataset size and the information separated during the split. It should be noted that the higher the rate of training data than testing data, better the performance achieved.

This is a good sign, since when considered as a real-world example, the models will have bigger weight for training data than testing.

4.4.2 The Confusion Matrix

To further analyze the quality of the classes produced by the categorization model, we can look at the confusion tables. A robust contusion framework is used to assess the quality of a classification. It is obtained by comparing the classified data with reference data which must be different from that used to carry out the classification.

• **TP True Positive:** number of cases where the model correctly predicts thepositive class.

• **FP False Positive:** number of cases where the model incorrectly predicts the positive class.

• **FN False Negative:** number of cases where the model incorrectly predicts the negative class.

• **TN True Negative:** number of cases where the model correctly predicts the negative class.

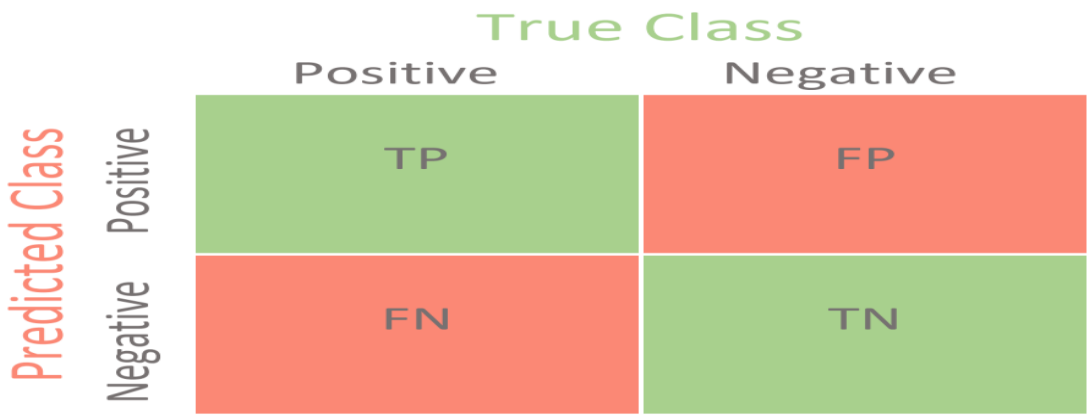


Figure 4-1 Confusion matrix

4.4.3 Precision

It is a measure that indicates the ability of the classifier to properly classify documents. Formally, precision is expressed as shown below

$$\text{Precision}=\frac{TP}{TP+FP}$$

Equation 3 Precision Formula

precision is a good way to determine if the cost of a false positive is high.

4.4.4 Recall

Recall how the recall is calculated as shown below.

$$\text{Recall}=\frac{TP}{TP+FN}$$

Equation 4 Recall Formula

So Recall actually calculates the number of positive points captured by our model by calling it positive (true positive). Applying the same interpretation, we know that Recall will be the model metric that we will use to select our best model when the costs associated with False Negative are high. For example, in the detection of sick patients. If a sick patient (real positive) goes through the test and predicts as not sick (negative predictor). The cost associated with False Negative will be extremely high if the disease is contagious.

4.4.5 F1 Score

The F1 score is a combination of the other two firsts.The figure below shows the formula to calculate F1 Score.

$$\text{F1-Score}=\frac{2*precision*Recall}{precision+recall}$$

Equation 5 F1-Score Formula

F1 score is necessary when you want to find a balance between precision and recall. We have previously seen that accuracy can be largely dependent on a large number of true negatives which, in most commercial circumstances, are not very targeted, while False negative and False positive usu-

ally has operating costs (tangible and intangible), so that F1 score could be a better measure to use if we have to find a balance between precision and recall AND if there is an unequal distribution of classes (large number of real negatives).

4.5 Results

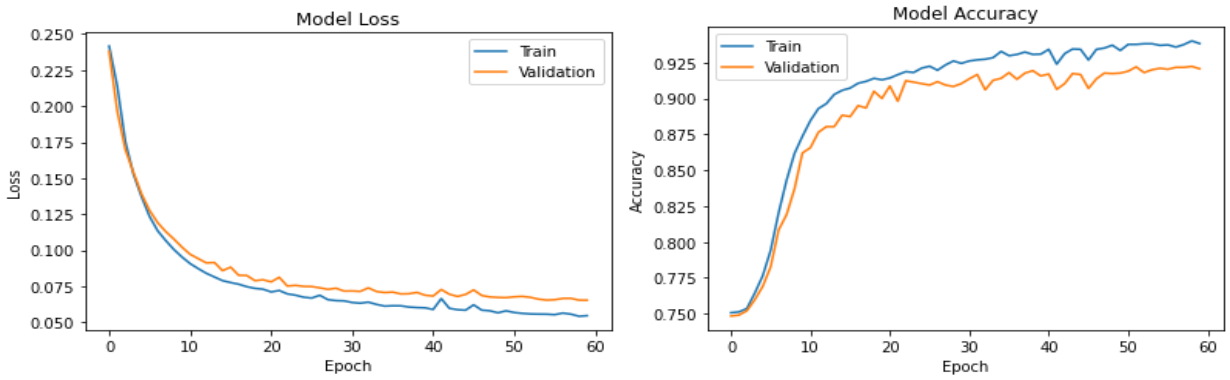


Figure 4-2 accuracy and loss curves

According to FIGURE 2, 95,13% accuracy is obtained by Siamese algorithm.

4.6 Comparison and discussion of results

From the results we obtained, we noticed that our model based on Siamese network gave better re-sylts comparing to our other models based on LSTM and CNN and to another published work that used the same data set as we did.Table 3 summarize the results obtained compared to another previous work that used the same dataset as we did.

Model	Accuracy
CNN	94.45%
LSTM	92%
(Hina et al., 2021)	95%
Our proposed model SIAMESE	**95.13%**

Table 3 Multiclass classification performance of algorithms

4.7 Conclusion

This chapter has encompassed the main results of the dissertation, the experiments have been detailed and explained, the results have been illustrated in the form of figures and tables in order to

fully understand the limitations and performances of the applied classification models. Results can be improved by focusing on the preprocessing steps.

Conclusion

Phishing emails is considered as the fastest rising online crime method used for stealing personal financial data and commit identity theft. Motivated by this aspect, we tackled in this dissertation the task of email classification. For this purpose, we proposed to use word embeddings to expand the emails and Siamese(malstm)to capture the semantic similarity between them. Experiments conducted on large dataset show that our approach can greatly improve the email matching task. Interestingly, we showed that MaLSTM is capable of modeling complex semantics and covering the context information of email pairs.

References

Abu-Nimeh, S., and Nair, S. (2008). “Bypassing security toolbars and phishing filters via dns poisoning,” in IEEE GLOBECOM 2008–2008 IEEE global telecommunications conference, New Orleans, LA, November 30–December 2, 2008 (IEEE), 1–6. doi:10.1109/GLOCOM.2008.ECP.386

Aburrous, M., Hossain, M. A., Thabatah, F., and Dahal, K. (2008). “Intelligent phishing website detection system using fuzzy techniques,” in 2008 3rd international conference on information and communication technologies: from theory to applications (New York, NY: IEEE, 1–6. doi:10.1109/ICTTA.2008.4530019

Almomani, A., Gupta, B. B., Atawneh, S., Meulenberg, A., & Almomani, E. (2013). A survey of phishing email filtering techniques. *IEEE Communications Surveys and Tutorials*, *15*(4), 2070–2090. https://doi.org/10.1109/SURV.2013.030713.00020

APWG,2018. https://docs.apwg.org/reports/apwg_trends_report_q2_2018.pdf

Boualem, B., & Meriem, L. (2021). *THEME Prédiction des manifestations publiques à l’aide de réseaux de neurones artificiels*.

CISA,2018. https://www.cisa.gov/sites/default/files/publications/CISA%20Global_2.1.21_508.pdf.

CISCO,2018. https://www.cisco.com/c/dam/en_us/about/annual-report/2018-annual-report-full.pdf.

Czum, J. M. (2020). Dive Into Deep Learning. *Journal of the American College of Radiology*, *17*(5), 637–638. https://doi.org/10.1016/j.jacr.2020.02.005

Dasgupta, D., Akhtar, Z., & Sen, S. (2022). Machine learning in cybersecurity: a comprehensive survey. *Journal of Defense Modeling and Simulation*, *19*(1), 57–106. https://doi.org/10.1177/1548512920951275

Dragomir, R.,Qazvinian,W.(2008). Scientific Paper Summarization Using Citation Summary Networks. https://aclanthology.org/C08-1087.pdf.

FTC (2018). https://www.ftc.gov/phishing-0.

F5 networks (2018). https://www.f5.com/company/news/press-releases/f5-networks-announces-

fourth-quarter-and-fiscal-year-2018-result.

Hina, M., Ali, M., Javed, A. R., Ghabban, F., Khan, L. A., & Jalil, Z. (2021). SeFACED: Semantic-Based Forensic Analysis and Classification of E-Mail Data Using Deep Learning. *IEEE Access*, *9*, 98398–98411. https://doi.org/10.1109/ACCESS.2021.3095730

Jakobsson, m. and myers, s. (2006) phishing and countermeasures: understanding the increasing problem of electronic identity theft.

Jakobsson,m et al (2007). Social phishing. https://www.researchgate.net/publication/220424040_Social_phishing

Joseph johnson(2019). "most prevalent spam content categories worldwide in 2019 number of clusters: 3 must know methods"

Keck (2018). https://www.researchgate.net/figure/Screenshot-of-the-A-Netflix-scam-email-and-B-fraudulent-text-message-Apple-Keck_fig4_349312504.

kirda ,s.Kruger,R(2005). An integrated approach to detect phishing mail attacks: a case study. https://dl.acm.org/doi/10.1145/1626195.1626244

kserpersky (2020). https://usa.kaspersky.com/about/press-releases/2020_phishing-grew-more-targeted-and-diverse-during-covid-19-outbreak.

Kulkarni, A., & Shivananda, A. (2019). Natural Language Processing Recipes. In *Natural Language Processing Recipes*. https://doi.org/10.1007/978-1-4842-4267-4

Latto, N. (2020). What is adware and how can you prevent it? Avast. Available at: https://www.avast.com/c-adware

Ma, L., Ofoghi, B., Watters, P., & Brown, S. (2009). Detecting phishing emails using hybrid features. *UIC-ATC 2009 - Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing in Conjunction with the UIC'09 and ATC'09 Conferences*, 493–497. https://doi.org/10.1109/UIC-ATC.2009.103

Merwe, A. v. d., Marianne, L., and Marek, D. (2005). "Characteristics and responsibilities involved in a Phishing attack, in WISICT '05: proceedings of the 4th international symposium on information and communication technologies. Trinity College Dublin, 249–254.

Nathan, G. (2020). What is phishing? + laws, charges & statute of limitations. Available

at: https://www.federalcharges.com/phishing-laws-charges/

Ollmann, G. (2004). The phishing guide understanding & preventing phishing attacks abstract. USA. Available at: http://www.ngsconsulting.com.Google Scholar

Othman, N., Faiz, R., & Smaïli, K. (2019). Manhattan siamese LSTM for question retrieval in community question answering. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11877 LNCS*, 661–677. https://doi.org/10.1007/978-3-030-33246-4_41

PhishMe (2016). Q1 2016 malware review. Available at: WWW.PHISHME.COM.

PishTank (2006). What is phishing. Available at: http://www.phishtank.com/what_is_phishing.php?view=website&annotated=true

Ramanathan, V., & Wechsler, H. (2012). PhishGILLNET-phishing detection methodology using probabilistic latent semantic analysis, AdaBoost, and co-training. *Eurasip Journal on Information Security*, *2012*, 1–22. https://doi.org/10.1186/1687-417X-2012-1

Rhett, J. (2019). Don't fall for this new Google translate phishing attack. Available at: https://www.gizmodo.co.uk/2019/02/dont-fall-for-this-new-google-translate-phishing-attack/

Russac, M. (2018). Phishing defintion. Available at: https://searchsecurity.techtarget.com/definition/phishing.

Wang, X., Zhang, R., Yang, X., Jiang, X., and Wijesekera, D. (2008). "Voice pharming attack and the trust of VoIP," in Proceedings of the 4th international conference on security and privacy in communication networks, SecureComm'08, 1–11. doi:10.1145/1460877.1460908