

# **An Automated System To Detect Social Engineering Attacks Using ML Algorithm**



MSC

Author

**Tooba Younis**

**Registration No: 00000362720**

Supervisor:

**Associate Prof Dr. Javed Iqbal**

A thesis submitted to the faculty of Computer Software Engineering, Military College of Signals (MCS), National University of Sciences and Technology, Rawalpindi in partial fulfillment of the requirements for the degree of MS in Software Engineering

(Oct - 2024)

**THESIS ACCEPTANCE CERTIFICATE**

Certified that final copy of MS/MPhil thesis written by Ms TOOBA YOUNIS, Registration No. 00000362720, of Military College of Signals has been vetted by undersigned, found complete in all respect as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial, fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the student have been also incorporated in the said thesis.

Signature: Javed  
Name of Supervisor: Assoc Prof Dr Javed Iqbal

Date: 10/10/24

Signature (HoD): [Signature]  
Date: 22/10/24

Signature (Dean/Principal): [Signature]  
Date: 24/10/24  
Brig  
Dean, MCS (NUST)  
(Asif Masood, Phd)

**NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY**  
**MASTER THESIS WORK**

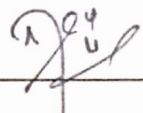
We hereby recommend that the dissertation prepared under our supervision by **Tooba Younis**, Regn No **00000362720** Titled: "**An Automated System to Detect Social Engineering Attacks Using ML Algorithm**" be accepted in partial fulfillment of the requirements for the award of **MS Software Engineering** degree.

**Examination Committee Members**

1. Name: **Assoc Prof Dr .Yawar Abbas Bangash**

Signature: 


2. Name: **Asst Prof Dr. Nauman Ali Khan**

Signature: 

Supervisor's Name: **Assoc Prof Dr Javed Iqbal**

Signature: 

Date: 22/10/24

  
 Head of Dept of CSE  
 College of Sigs (NUST)


Head of Department

22/10/24

Date

**COUNTERSIGNED**

Date: 24/10/24



Brig  
 Dean, MCS (NUST)  
 (Asif Masood, Phd)

Dean


## CERTIFICATE OF APPROVAL

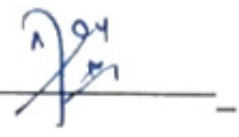
This is to certify that the research work presented in this thesis, entitled "An Automated System to Detect Social Engineering Attacks Using ML Algorithm" was conducted by Mr. Tooba Younis under the supervision of Assoc Prof Dr. Javed Iqbal. No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the Department of Computer Software Engineering in partial fulfillment of the requirements for the degree of Master of Science in Field of Computer Software Engineering Department of Military College of Signals, National University of Sciences and Technology, Islamabad.

Student Name: Tooba Younis

Signature: 

### Examination Committee:

a) External Examiner 1: Assoc Prof Dr. Yawar Abbas Signature:   
(Department of Computer Software Engineering)

b) External Examiner 2: Asst Prof Dr. Nauman Ali Khan Signature:   
(Department of Computer Software Engineering)

Name of Supervisor: Assoc Prof Dr. Javed Iqbal

Signature: 

Name of Dean/HOD: Col Usman Mahmood Malik, PhD

Signature: 

**Col  
Head of Dept of CSE  
Mil College of Sigs (NUST)**

# Plagiarism Undertaking

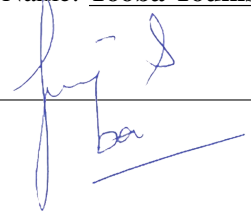
I solemnly declare that research work presented in the thesis titled “**An Automated System To Detect Social Engineering Attacks Using ML Algorithm**” is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero-tolerance policy of the HEC and National University of Sciences and Technology (NUST), Islamabad towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the University reserves the rights to withdraw/revoke my MS degree and that HEC and NUST, Islamabad has the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized thesis.

Author Name: **Tooba Younis**

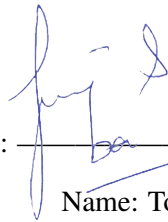
Signature: \_\_\_\_\_



## Author's Declaration

I, Tooba Younis, hereby state that my MS thesis titled “An Automated System To Detect Social Engineering Attacks Using ML Algorithm” is my own work and has not been submitted previously by me for taking any degree from National University of Sciences and Technology, Islamabad or anywhere else in the country/ world. At any time if my statement is found to be incorrect even after I graduate, the university has the right to withdraw my MS degree.

Student Signature: \_\_\_\_\_



Name: Tooba Younis

Date: 16-10-2024

# **Dedication**

“In the name of Allah, the most Beneficent, the most Merciful”

I dedicate this thesis to my family, friends, and teachers who supported me each step of the way, especially my parents.

This thesis is also dedicated to all the deserving children who do not have access to quality education especially young girls.

## **Acknowledgments**

Glory be to Allah (S.W.A), the Creator, the Sustainer of the Universe. Who only has the power to honor whom He please, and to abase whom He please. Verily no one can do anything without His will. From the day, I came to NUST till the day of my departure, He was the only one Who blessed me and opened ways for me, and showed me the path of success. There is nothing which can payback for His bounties throughout my research period to complete it successfully.

**Tooba Younis**



# Abstract

Socio technical threats remain a major problem in cybersecurity, especially since they involve getting information relating to security or even making the people perform a security-oriented act. This thesis proposes a new deep learning method, named SEAP (Social Engineering Attack Prevention), for discovering and protecting against such attacks. As the nature of the SEAP model is based on the unsupervised pre-training and supervised fine-tuning, it results in the above-mentioned benefits of higher capacity of the system to identify nonlinear, latent relationships connected with the occurrence of social engineering attacks. Concerning the SEAP architecture, we also have ConvMix blocks to improve the detection while not making calculations of large datasets unmanageable. The employed dataset in this study is Phishing which comprises of record 10000 and predictor 50; which includes URL syntactical structure and content. Data preprocessing is then carried out on the database to prepare it for training involving process including scaling or feature engineering or normalization. In this respect, the Social Engineering Attack Prevention ([SEAP](#)) model pre-infuses a Deep Belief Network ([DBN](#)) architecture of multiple layers of Restricted Boltzmann Machines (RBMs). The self-adjusting phase of the variables and then learning of the parameters through the use of the Contrastive Divergence algorithm is then followed by an improvement in the classification through the application of the back-propagation supervised fine-tuning. The effectiveness of the SEAP model is checked through experiments and it meets high results, namely 96% of the accuracy with the help of the

dataset of the detection of phishing. Indicators of performance such as the exactness, the rate of recall, and the two types of the f-scores where each of them was equal to 0.96. Thus, while adopting the identification of anomalous behavior to the given subject, this thesis highlights the necessity of employing additional ML approaches to prevent social engineering attacks. Overall, there are two main advantages found in the novel SEAP model: this suggestion in its architecture and methodology is quite effective for a broad scale of enhancement of cybersecurity and can be employed efficiently in real-life scenarios like designing the system for detecting the phishing. Further investigative efforts can develop more of such approaches that improve the identification processes constituting the formation of secure and resilient cyber realms.

Keywords: social engineering; Cyber threats; Identification; Network node; Cyber protection

# Table of Contents

Dedication .....	vii
Acknowledgments .....	viii
Abstract .....	ix
List of Tables.....	xiii
List of Figures .....	xiv
List of Abbreviations and symbols .....	xv
Chapter 1: Introduction .....	01
1.1 Research Motivation .....	06
1.2 Research Contribution.....	10
1.3 Thesis Organization .....	10
Chapter 2: Types of Social Engineering Attack.....	12
2.1 Phishing Attacks .....	12
2.2 URL Manipulation.....	13
2.3 Obfuscated URLs .....	14
2.4 Resource Manipulation .....	14
2.5 Abnormal URL Behaviors.....	15
2.6 Content-Based Features .....	16
2.7 Summary of Features.....	16
Chapter 3: Related Work.....	18
Chapter 4: Research Methodology .....	21
4.1 Dataset .....	24

4.2	Data Preprocessing .....	25
4.3	Architecture .....	29
Chapter 5:	Results .....	34
Chapter 6:	Discussion .....	37
Chapter 7:	Conclusion and Future Work .....	42
Bibliography	.....	47

# List of Tables

3.1	Other work done by different ex-researchers on the enactment of a social engineering attack. ....	20
4.3	Algorithm 1: SEAP Implementation for feature extraction. ....	33

# List of Figures

4.1	Dataset Representation of Binary Classification.....	25
4.2	This symbolizes the Preprocessing of the dataset used in algorithms.....	27
4.3	Some of the labels' names included in the heat map result are shown in the map.	27
4.4	Some of the labels' names included in the heat map result are shown in the map.	28
4.5	Some of the labels' names included in the heat map result are shown in the map.	28
4.6	Some of the labels' names included in the heat map result are shown in the map.	29
4.7	Explaining the components of SEAP architecture using a diagram.....	31
5.1	Stands for the training and validation loss and accuracy .....	35
5.2	Stands for the output of the phishing detection system.....	36

# List of Abbreviations and Symbols

## Abbreviations

<b>DDos</b>	Denial-of-service
<b>RBM</b> s	Restricted Boltzmann Machines
<b>SEAP</b>	Social Engineering Attack Prevention
<b>DBN</b>	Deep Belief Network

## CHAPTER 1

# Introduction

Social engineering threats are evolving at an alarming pace in today's technologically advanced world, posing severe consequences for societies, organizations, and individuals alike. These threats are diverse and can manifest in multiple spheres of life, ranging from monetary damages to psychological and social impacts. One of the primary motives behind social engineering attacks is financial gain, often accomplished by gaining the trust of unsuspecting victims. Attackers typically build a relationship with their targets and manipulate them into revealing confidential information. Once trust is established, the victims' money is often transferred to unauthorized accounts without their consent, leading to significant financial losses. The direct consequences for victims include unauthorized transactions, identity theft, and fraudulent purchases [8]. The problem is compounded by the fact that many people remain unaware of these deceptive techniques, making them more susceptible to such attacks. For organizations, social engineering can lead to data breaches, loss of sensitive records, and unauthorized access to critical systems, creating a cascade of negative effects on operational stability and reputation [7] [15].

The repercussions of social engineering attacks extend beyond immediate financial losses. In



cases where sensitive information such as records, passwords, or financial data is stolen, victims may experience long-term damage in the form of identity theft or unauthorized access to their personal accounts. This stolen information can be utilized for various unlawful purposes, including the opening of new bank accounts, generation of fraudulent identities, or even the submission of loan applications in the victim's name [13][22]. Social engineering, therefore, poses a unique risk by exploiting the human factor, making it difficult to detect and prevent using conventional security measures. Organizations, in particular, are at heightened risk, as cybercriminals often leverage stolen credentials to infiltrate corporate networks. Once inside, they can access sensitive data, steal trade secrets, or even launch more devastating attacks, such as planting malware or ransomware, which can cripple entire systems [1] [9]. The financial implications are not limited to the immediate losses suffered by victims but also include reputational damage, potential regulatory penalties, and the cost of legal settlements [5] [6].

The nature of social engineering attacks is particularly dangerous because they manipulate users into willingly providing sensitive information, making them feel responsible for the consequences. The psychological aspect of social engineering attacks is crucial in understanding their effectiveness. Criminals often employ a variety of tactics, such as creating a sense of urgency, invoking fear, or presenting seemingly authoritative requests to exploit the target's natural inclinations to trust or act quickly [11]. By the time the victim realizes what has happened, it is often too late to recover the stolen assets or mitigate the damage. Moreover, specific identity sets stolen in social engineering attacks can be sold on the dark web or used for a wide range of unlawful activities, including financial fraud, false identity generation, or even impersonation for criminal purposes [14][12]. Such activities undermine public trust in digital systems and create a ripple effect, making it more difficult for legitimate organizations to assure customers of the safety of their personal information. [23]

Organizations are not immune to these attacks; on the contrary, they are frequent targets. Social engineering attacks on businesses can compromise client information, employee credentials, or intellectual property, resulting in severe operational and reputational harm. For instance, gaining access to an employee's login details might allow a cybercriminal to access sensitive systems and conduct further attacks, such as data exfiltration or injecting malicious software into the network [17] [19]. Once inside, these intruders can remain undetected for long periods, systematically collecting valuable information or disrupting the organization's operations. Such breaches often lead to significant losses, both in terms of direct financial impact and the erosion of customer and partner trust [16] [20]. When customers feel their information is at risk, they are likely to discontinue business relationships, leading to loss of revenue and market share. The fear of potential security breaches and the long-term impact of being associated with high-profile data breaches are long-standing vulnerabilities that can cripple a company's growth and success [3].

In addition to monetary losses and reputational damage, social engineering attacks can also have legal and regulatory consequences. Organizations that fail to protect their clients' data adequately may find themselves facing hefty fines and legal actions. This is particularly true in industries that handle sensitive personal information, such as finance or healthcare, where regulatory bodies impose strict data protection requirements. The legal liabilities incurred due to data breaches can include regulatory penalties, litigation costs, and compensation for affected parties [10]. Furthermore, in regions with stringent data protection laws, such as the General Data Protection Regulation (GDPR) in Europe, organizations found guilty of failing to secure customer data can face penalties that amount to a significant percentage of their global revenue. The cumulative financial burden, along with the cost of rebuilding trust, can be overwhelming, especially for small to medium-sized enterprises [21].

One of the most troubling aspects of social engineering is that it is not limited to digital interactions. Attackers often employ non-technical tactics, such as pretexting or quid pro quo, to achieve their goals. Pretexting involves creating a fabricated scenario, where the attacker impersonates a trusted figure, like a police officer or IT technician, to manipulate the victim into disclosing confidential information. This method is particularly effective because it capitalizes on the victim's natural inclination to comply with authority figures. Another common tactic is the use of quid pro quo, where the attacker offers something in return for information or access [18]. For example, an attacker might pose as a technical support agent, offering to fix a non-existent issue in exchange for the victim's credentials. These tactics highlight the diverse nature of social engineering and the creativity of attackers in exploiting human psychology [2].

To further complicate matters, attackers often target specific individuals within organizations, such as executives or IT personnel, using a strategy known as spear phishing. This targeted approach increases the likelihood of success, as the attacker customizes the attack based on the victim's role and personal details, making the communication appear legitimate [3]. In such scenarios, even well-trained individuals can fall prey to the deception, resulting in severe security breaches. Once the attacker gains access to an executive's account, they can exploit it to issue commands, authorize transactions, or even disseminate false information within the organization, causing widespread confusion and disruption [24]. The sophistication of these attacks makes it imperative for organizations to not only implement technical security measures but also focus on training employees to recognize and respond to social engineering attempts.

The consequences of large-scale social engineering attacks can be even more severe, affecting public infrastructure, government systems, and critical services. A well-executed attack on a power grid or water supply system, for instance, could disrupt vital services and pose serious risks to public safety. Such attacks erode societal trust and create a climate of fear and

uncertainty. Furthermore, attackers targeting public sector entities can access classified information or disrupt national security operations, leading to geopolitical ramifications [4]. The interconnected nature of modern digital infrastructure means that a breach in one area can have far-reaching consequences, potentially affecting multiple sectors simultaneously.

Preventing social engineering attacks requires a multi-faceted approach that includes raising awareness, implementing robust security protocols, and regularly updating these protocols to adapt to emerging threats. Awareness and education are crucial, as even the most sophisticated security systems can be rendered useless if the human element is compromised. Regular training sessions can help employees and individuals recognize the common tactics used in social engineering attacks and respond appropriately [18]. In addition, organizations should implement stringent access control measures, such as multi-factor authentication (MFA), to minimize the risk of unauthorized access. Security protocols should also include regular updates and audits to ensure that they remain effective against the latest attack strategies.

Monitoring and incident response are also vital components of a comprehensive defense strategy. Organizations should establish systems to monitor suspicious activities and have a clear incident response plan to contain and mitigate potential breaches. By isolating affected systems and notifying relevant stakeholders, organizations can prevent further damage and recover more quickly [9]. Finally, technical defenses, such as spam filters, intrusion detection systems, and endpoint security solutions, should be configured to detect suspicious behavior and flag potential threats before they escalate [4]. However, technical solutions alone are insufficient; a culture of security awareness and vigilance is essential to counter the ever-evolving nature of social engineering threats.

In conclusion, social engineering is a powerful and pervasive threat that preys on human vulnerabilities rather than technological weaknesses. Its impacts are far-reaching, affecting financial

stability, personal privacy, and organizational reputation. Combating social engineering requires a holistic approach that encompasses awareness, education, technical defenses, and a strong organizational culture of security. By understanding the methods used by attackers and remaining vigilant, both individuals and organizations can reduce their susceptibility to these deceptive tactics and safeguard against the potentially devastating consequences.

## **1.1 Research Motivation**

Research in the domain of technical social engineering attacks is driven by the increasing prevalence of these attacks and the lack of robust detection mechanisms capable of identifying and mitigating them effectively. Technical social engineering attacks leverage the human factor, exploiting inherent trust, psychological manipulation, and lack of awareness among users to gain unauthorized access to systems, networks, or sensitive data. These attacks have evolved significantly over the years, from simple phishing emails to more sophisticated techniques such as spear phishing, baiting, and pretexting, making it difficult for conventional security systems to detect and neutralize them. Consequently, this necessitates a comprehensive and practical approach to identify, analyze, and mitigate the impacts of these attacks using modern technologies and interdisciplinary research.

One of the core motivations for conducting research in this domain is the inadequacy of traditional cybersecurity measures in addressing the complex nature of technical social engineering attacks. Traditional approaches typically focus on technical vulnerabilities within software and hardware systems, overlooking the human aspect, which is often the weakest link in security frameworks. Human users are prone to deception, manipulation, and trust exploitation, making them prime targets for attackers. Attackers often bypass advanced firewalls, encryption, and intrusion detection systems by manipulating the behavior of individuals within an organi-

zation. As a result, technical social engineering attacks present unique challenges that cannot be effectively addressed using conventional defense mechanisms alone. This gap highlights the urgent need for research into innovative solutions that incorporate behavioral analysis, machine learning, and artificial intelligence to detect and prevent these threats proactively.

Machine learning and artificial intelligence offer promising solutions for tackling social engineering attacks due to their capability to identify subtle patterns and anomalies in large datasets. For instance, behavioral analytics can be used to monitor user activity and detect deviations from normal behavior that may indicate the presence of an attack. Machine learning models can be trained on datasets comprising legitimate and malicious interactions to distinguish between genuine user behavior and actions that resemble social engineering attempts. Such models can evolve over time, adapting to new attack patterns and becoming more accurate as they are exposed to larger datasets. By leveraging AI-based pattern recognition techniques, it is possible to identify indicators of compromise that are not apparent through traditional rule-based systems. This adaptability is crucial in a landscape where attackers are constantly refining their tactics to evade detection.

Another key motivation for research in this field is the integration of natural language processing (NLP) techniques to analyze communication channels for signs of social engineering attacks. Many technical social engineering attacks are conducted through email, instant messaging, or voice communications, where the attacker impersonates a trusted individual or organization to extract sensitive information from the victim. NLP can be employed to analyze the content and context of these communications, looking for red flags such as urgency, unusual requests, or inconsistencies in language use. By developing NLP-based models that can automatically parse and analyze textual and voice-based communications, it becomes possible to detect potential social engineering attempts before the victim responds. This proactive approach is particularly

valuable for organizations with a high volume of daily communications, where manually reviewing every interaction is impractical.

Furthermore, incorporating advanced threat intelligence into the detection of social engineering attacks can significantly enhance an organization's defense capabilities. Threat intelligence involves gathering and analyzing information about current and emerging threats, including tactics, techniques, and procedures (TTPs) used by attackers. By integrating threat intelligence feeds with machine learning and behavioral analytics, organizations can develop a more comprehensive view of the threat landscape and identify potential social engineering attacks more accurately. This approach enables security teams to stay ahead of attackers by understanding how social engineering tactics evolve over time and adapting their defenses accordingly. The fusion of threat intelligence with AI and machine learning models allows for the creation of dynamic and context-aware defense systems that can automatically adjust their strategies based on the latest threat information.

The importance of conducting research in this domain also stems from the potential societal impact of social engineering attacks. Technical social engineering attacks are not limited to targeting individuals or organizations; they can have far-reaching consequences for critical infrastructure, public safety, and national security. For instance, a successful social engineering attack on a government agency could result in the compromise of sensitive data, disruption of essential services, or even manipulation of public opinion. Similarly, attacks on financial institutions or healthcare organizations can lead to financial losses, identity theft, and compromise of personal health information, resulting in severe emotional and psychological distress for victims. Given the potential scale and impact of such attacks, there is an urgent need for research that focuses on developing holistic solutions that address not only the technical aspects of security but also the human and organizational factors that contribute to vulnerability.

Moreover, the dynamic nature of social engineering tactics demands research into adaptive and resilient defense mechanisms. Attackers continuously modify their strategies to exploit new vulnerabilities and evade detection. This requires the development of security systems that are not only effective at detecting known attack patterns but are also capable of identifying previously unseen tactics. Research in this area must focus on creating flexible models that can generalize from existing data and recognize novel social engineering techniques. This includes developing unsupervised learning models that can identify anomalies without requiring labeled data, as well as reinforcement learning models that can learn optimal defense strategies through simulated interactions with potential attackers.

In conclusion, the motivation for research in the field of technical social engineering attacks is driven by the need to address the growing sophistication and prevalence of these threats. Traditional security measures are no longer sufficient to counter attacks that exploit human vulnerabilities. By leveraging advanced technologies such as machine learning, artificial intelligence, natural language processing, and threat intelligence, it is possible to develop comprehensive defense mechanisms that are capable of detecting, analyzing, and mitigating social engineering attacks in real time. This research is not only vital for protecting individual users and organizations but also for safeguarding critical infrastructure and ensuring the security and stability of society as a whole. The interdisciplinary nature of this research, which combines elements of cybersecurity, psychology, and artificial intelligence, makes it a challenging yet essential field of study that holds the potential to significantly enhance the effectiveness of modern security frameworks.



## 1.2 Research Contribution

We are introducing a novel deep-learning model designed to tackle the challenge of social engineering attacks. Below are the key features of our proposed system:

- Traditional models often balance complexity and interpretability. Introducing a new deep learning model SEAP, which leverages unsupervised pre-training followed by supervised fine-tuning, could potentially enhance detection performance, particularly in capturing non-linear and complex patterns that simpler models might miss.
- The SEAP model system architecture integrates convmix blocks, optimizing its ability to detect attacks linked with social engineering.
- SEAP model can be computationally less expensive to train and fine-tune, especially on large datasets.

## 1.3 Thesis Organization

This thesis is divided into seven chapters:

**Chapter 1:** This chapter includes the basic introduction, background, research motivation and research contribution.

**Chapter 2:** This chapter presents a comparative analysis between our research and the latest advancements in the field.

**Chapter 3:** This chapter provides an overview of relevant literature, encompassing articles pertinent to the scope of this study.

**Chapter 4:** This chapter presents the material and methods.

**Chapter 5:** This chapter delivers the experimental results.

CHAPTER 1: INTRODUCTION

**Chapter 6:** This chapter describes the discussion.

**Chapter 7:** This chapter concludes the report and highlights the direction for future work.

## CHAPTER 2

# Types of Social Engineering Attack

Types of social engineering attack that can be identified using the features in the provided dataset:

## 2.1 Phishing Attacks

### 2.1.1 Description

Phisher attacks resemble various other social engineering ploys such as the approach of how the victim is induced to hand over his or her login credentials or monetary data and other details.

### 2.1.2 Features

- NumDots: If the number of points in the 'URL' is greater, it may be said that such a site is in the 'phishing' category because the attackers register new subdomains that resemble the authorized sites.
- SubdomainLevel: Most subdomains can give the URL a look of legitimacy hence making the users believe in the site.

- **UrlLength:** The long URL strings are misleading, and a URL containing a virus link does not look like a virus at all.

## **2.2 URL Manipulation**

### **2.2.1 Description**

It is a method in which the attackers modify the URLs and make them look like the other authentic ones. This can in effect mislead users into thinking, they are on a site they can trust.

### **2.2.2 Features**

- **NumDash:** Dashes are often used to create URLs that look similar to legitimate ones.
- **NumDashInHostname:** Dashes in the hostname can be a red flag, as legitimate domains rarely use them.
- **AtSymbol:** '@' is another character that can reach trick-swallowing URLs up to the '@' character, included, no part of the URL is visible to the browser.
- **TildeSymbol:** There are also somethings like the symbol “~” which is quite queer especially when put in real URLs and it is more rampant in phishing scams.
- **NumUnderscore:** Underlined parts of the URLs should also be noted as one of the strategies of deception to create a variety of URLs.

## **2.3 Obfuscated URLs**

### **2.3.1 Description**

It's a category of URL that looks like an innocent and genuine Website, which in reality, has a complex coding that makes it difficult to determine the intention of the Website.

### **2.3.2 Features**

- SubdomainLevelRT: The findings also show that when the subdomain levels are high in real-time analysis, then the field is most likely to be obscured.
- UrlLengthRT: In this approach, URL length is monitored in real-time to determine URLs that have many characters and are typically used in concealing the authentic URL behind a pleasant look-and-feel with ineffective content.
- PctExtResourceUrlsRT: If there are many resource URLs listed, and most of them are from external web pages, it can be considered the web page has obfuscation or contains malicious code.

## **2.4 Resource Manipulation**

### **2.4.1 Description**

Some of these adaptations of the webpage resources conceal their malevolent plans as the iframes that are always used to load the content from the other site with undesirable scripts.

## 2.4.2 Features

- **IframeOrFrame:** It is believed that iframes can be an indication that an effort is being made to load some content from another site even if the iframes do not look suspicious.
- **MissingTitle:** A missing title can be a sign of a hastily put-together phishing page.
- **ImagesOnlyInForm:** Forms that rely heavily on images can be designed to deceive users visually.

## 2.5 Abnormal URL Behaviors

### 2.5.1 Description

For instance, if when using a browser, the fingerprints of URLs lie in the unusual realm, then it might mean a social engineering attack; such a state consists of unexpected redirects as well as awkward-form actions.

### 2.5.2 Features

**AbnormalExtFormActionR:** High values of the External Form actions may be suggestive of the fact that the data is being forwarded to a third party.

**PctExtNullSelfRedirectHyperlinksRT:** It is also noteworthy that some attempts to manipulate the client's browsing experience can be indicated by features such as the ratio of external, null, or self-redirect hyperlinks.

## 2.6 Content-Based Features

### 2.6.1 Description

Properties that reflect on the content of a webpage to find out whether or not they contain aspects that may be used to deceive the users such as numerous links to other sites and many scripts.

### 2.6.2 Features

- PctExtResourceUrlsRT: If the percentage of URLs belonging to resources outside the site is high this would mean that the site is utilizing other outside resources and probably is a phishing site.
- ExtMetaScriptLinkRT: As for the real-time analysis, meta, script, and link tags coming from the related external URLs are indicative of a webpage loading malicious scripts and metadata.

## 2.7 Summary of Features

- NumDots: The digitized number of the dots in the URL.
- SubdomainLevel: This is relating to the number of subdomains in the URL structure.
- PathLevel: Segment of the URL that distinguishes the level of the path.
- UrlLength: Length of the URL.
- NumDash: Number of dashes in the URL.
- NumDashInHostname: Number of dashes in the hostname.
- AtSymbol: Presence of "@" symbol.

- TildeSymbol: Presence of " " symbol.
- NumUnderscore: Number of underscores in the URL.
- IframeOrFrame: Presence of iframe or frame.
- MissingTitle: Stating that for this path no good title is given.
- ImagesOnlyInForm: The decision to use pictures while only citing the source of images in forms.
- SubdomainLevelRT: The scope is attributed to the real-time subdomain level.
- UrlLengthRT: A large number of characters in the URL in real time.
- PctExtResourceUrlsRT: The URL share of external resources in the material in real time.
- AbnormalExtFormActionR: External form activities include those activities that are regarded as exceptional for an organization.
- ExtMetaScriptLinkRT: Function to add metadata, script, and link tags from the external tags list for the actual HTML file without reopening the file.
- PctExtNullSelfRedirectHyperlinksRT: Real-time comparison of the speed of the 'external', 'null', or 'self-redirect' hyperlink.

These in combination help in naming the forms of attack and or segmentation of SE by URL features and or the content of the Web page.



## CHAPTER 3

# Related Work

Over the recent past, very much focus has been accorded to coming up with means of identifying and combating sundry types of cyber threats such as malware and botnet attacks. According to the literature review, there are several significant works found in this area. Lysenko et al. (DESSERT 2020) have presented a technique for cyberattack detection using the evolutionary algorithms. Their work is mainly concerned with detecting possible cyber threats by studying the patterns in traffic or system activity on time [24]. Expanding on this approach, Savenko et al. (IDAACS 2021) focused on the detection of DNS tunneling botnets which is one of the most effective techniques employed by the attackers to bypass standard protection tools. In this way, their work helps expand knowledge about effective means of creating new detection mechanisms, which in turn would help improve the ability of networks to counteract covert communication channels that may be used by attackers [25]. In the meanwhile, Lysenko, Savenko, and Bobrovnikova (CEUR-WS 2018) proposed a method for detecting the distributed denial-of-service Denial-of-service (DDoS) botnets via semi-supervised fuzzy c-means clustering. Being built on the clustering algorithms, their approach also allows for developing the method of differentiation between the aggressive and non-aggressive traffic flows, which would help to timely detect and prevent DDoS attacks [21]. Also, Bobrovnikova et al. (Radioelectronic and Com-

puter Systems 2022) presented the Technique of Malware Detection for IoT Devices using the Control Flow Graph. It also works to emphasize the need to evaluate the program flow, not only to find symptoms of an attack but also to examine the inoculation of IoT systems with malware [2]. Continuing the development of the field of malware detection, Markowsky et al. (CEUR-WS 2018) described the approach to detect metamorphic viruses with the help of analysis of the obfuscation features. Moreover, by analyzing the features of the obfuscated code, the [9] work improves the prospects of identifying and categorizing certain types of metamorphic viruses, contributing to the fight against new forms of threats. In addition, the approach to the identification of the botnet was developed by Lysenko, Bobrovnikova, and Savenko (DESSERT 2018) based on the clonal selection algorithm. Based on the philosophies of the immune system, their approach presents a novel way of searching and destroying missions on botnet viruses affecting computer networks to boost the networks' security against such coordinated attacks [4]. Altogether, it can be seen that a wide variety of strategies and methods have been used to address cyber threats with the assistance of EE, CA, CFG analysis, and immune-inspired algorithms. The given contributions shed more light on the field of malware and botnet research that would enable academicians and researchers to design more effective and enhanced security solutions that can counter the emerging threats in the dynamic world of internet and computer technologies given in 3.1.

Ref	Methods	Techniques
[24]	Cyberattack detection using evolutionary algorithms	Evolutionary algorithms
[25]	Detection of DNS tunneling botnets	DNS Tunneling Botnet Attack Model
[21]	DDoS botnet detection using semi-supervised fuzzy c-means clustering	Semi-supervised fuzzy c-means clustering
[2]	IoT malware detection based on control flow graph analysis	Control flow graph analysis
[9]	Metamorphic virus detection based on obfuscation features	Obfuscation features analysis
[4]	Botnet detection using the clonal selection algorithm	Clonal selection algorithm

**Table 3.1:** Other work done by different ex-researchers on the enactment of a social engineering attack.

## CHAPTER 4

# Research Methodology

### **Deep Belief Network (DBN)**

A Deep Belief Network (DBN) is a generative graphical model or a type of deep neural network, composed of multiple layers of stochastic, latent variables. These variables typically represent hidden features in the data. Each layer in a DBN is trained to capture the statistical features of the input data, and the network can be used for various tasks such as classification, regression, and dimensionality reduction. A DBN framework is constructed by stacking multiple layers of Restricted Boltzmann Machines (RBMs).

### **Restricted Boltzmann Machine (RBM)**

An RBM is a stochastic neural network that can learn a probability distribution over its set of inputs.

It consists of two layers:

- Visible Layer: This layer contains the visible units (or nodes) that represent the input data.
- Hidden Layer: This layer contains the hidden units that capture latent features in the input data.

**Structure and Working of an RBM**

Symmetric Bipartite Graph: In an RBM, every visible unit is connected to every hidden unit, but no visible unit is connected to another visible unit, and no hidden unit is connected to another hidden unit. This forms a symmetric bipartite graph. Energy Function: The RBM defines a joint distribution over the visible and hidden layers using an energy function. The energy function for an RBM is given by:

$$E(v, h) = -i\sum a_i v_i - j\sum b_j h_j - i, j\sum v_i h_j w_{ij} \quad (4.0.1)$$

where  $v_i$  and  $h_j$  are the states of visible unit  $i$  and hidden unit  $j$  respectively,  $a_i$  and  $b_j$  are their biases, and  $w_{ij}$  is the weight between visible unit  $i$  and hidden unit  $j$ .

**Probability Distribution:**

The probability of a configuration  $(v, h)$  is given by:

$$P(v, h) = \frac{1}{Z} e^{-E(v, h)} \quad (4.0.2)$$

where  $(Z)$  is the partition function, a normalizing constant to ensure the probabilities sum to 1.

**Training:**

The most common training algorithm for RBMs is Contrastive Divergence (CD), which approximates the gradient of the log-likelihood.

**Deep Belief Network (DBN)**

A DBN is formed by stacking several RBMs on top of each other:

- Layer-wise Training: The first RBM is trained on the input data to learn the first layer of hidden features. The second RBM is then trained on the hidden features learned by the first RBM, and this process continues for all subsequent layers.
- Greedy Layer-wise Training: Each RBM is trained independently in a greedy, layer-wise

manner. Once a layer is trained, its parameters are fixed, and the next layer is trained on the transformed data (hidden features) from the previous layer.

- **Fine-tuning:** After the layer-wise pre-training, the entire DBN can be fine-tuned using backpropagation or other gradient-based optimization techniques to improve performance on a specific task, such as classification.

### **Advantages of DBNs**

- **Unsupervised Pre-training:** The layer-wise pre-training of RBMs can effectively initialize the weights, helping to avoid poor local minima and improving the overall training process.
- **Feature Extraction:** Each layer of the DBN can learn increasingly complex features, capturing hierarchical representations of the input data.
- **Flexibility:** DBNs can be used for both generative and discriminative tasks.

### **Applications of DBNs**

- **Image Recognition:** DBNs can learn to extract features from images for tasks like classification and object detection.
- **Speech Recognition:** They can model temporal dependencies in audio signals, making them useful for speech-to-text applications.
- **Dimensionality Reduction:** DBNs can reduce the dimensionality of data while preserving important features, making them useful for visualization and data compression.

### **Example Workflow**

- **Data Preparation:** Normalize and preprocess the input data.

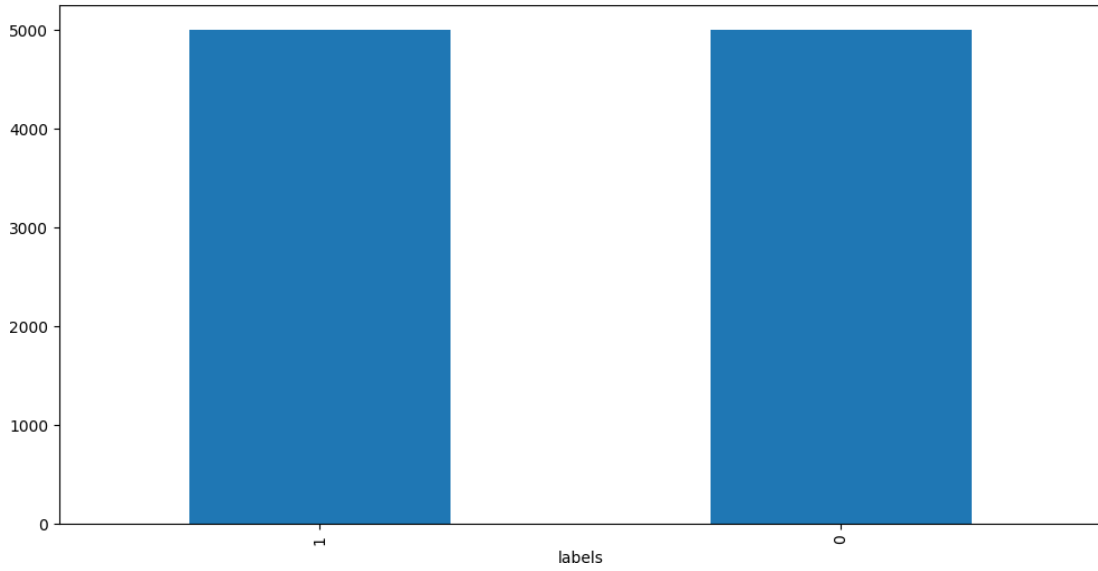
- RBM Layer 1: Train the first RBM on the input data to capture the first set of hidden features.
- RBM Layer 2: Train the second RBM on the hidden features obtained from the first RBM.
- Stacking Layers: Continue stacking and training RBMs in this manner.
- Fine-tuning: Once all layers are trained, fine-tune the entire network using supervised learning techniques if necessary.
- Evaluation: Evaluate the performance of the DBN on the desired task and make adjustments as needed.

By leveraging the layer-wise training of RBMs, DBNs can effectively learn complex data representations, making them a powerful tool for various machine learning applications.

## 4.1 Dataset

In this research, the authors have chosen the “Phishing\_Legitimate\_full.csv” dataset which is a detailed list to distinguish between legitimate and phishing URLs based on structural, syntactical as well as content features [18]. The data collection contains 10,000 records and 50 attributes, thus, comprehensively characterizing features. Each entry is also individual and includes structural factors such as the quantity of dots, dashes, and special characters and size characteristics of the URLs, hosts, paths, and queries. It documents the occurrence of basic features, including the IP address, the presence of HTTPS in the hostname, and the integration of brand names into the link since they are typical indicators of phishing scams. A content analysis indicates the content of the linked webpage and the activity in it whereas real-time characteristics reveal the status of the URL at the time of access. Other signs reveal other general phishing strategies like locking the mouse or displaying pop-ups. The variable “CLASS\_LABEL” represents the

ground truth or the genuine labels separating the URLs as legitimate (Class label = 0) or phishing, (Class label = 1). This set of features helps in building complex machine-learning models that result in highly accurate ways of identifying and classifying cases of phishing, which is why it is a highly useful dataset for researchers in the field of cybersecurity and creators of anti-phishing tools.



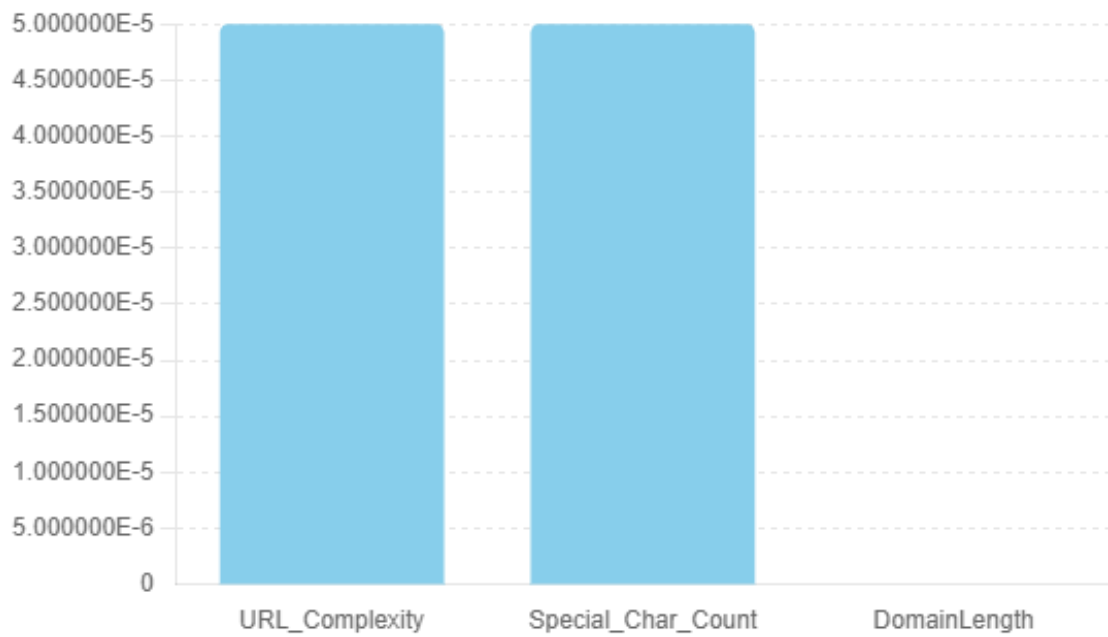
**Figure 4.1:** Dataset Representation of Binary Classification.

## 4.2 Data Preprocessing

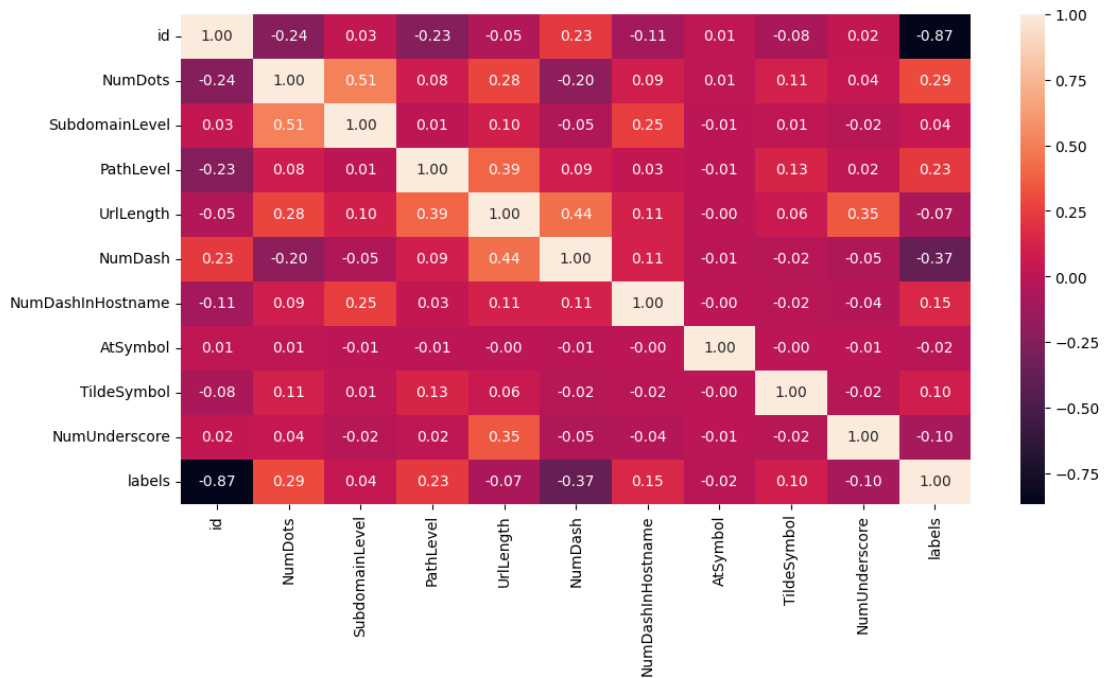
When doing the data preprocessing of the statistical data given, several sophisticated methods were used to improve the used dataset or make it suitable for analysis or model training. First, the RobustScaler was used to transform the numerical variables and thus bring the data into a better format for the machine learning models. This scaler works by seasonally differencing the features, which usually implies the removal of the median before scaling based on the interquartile range. Then, feature engineering was carried out to help in the extrication of further informative features. Henceforth, a new feature, ‘URL\_Complexity’, was created us-



ing the following features: count of dots, subdomain levels, path levels, and number of dashes and hyphens in the hostname. Another feature, 'Special\_Char\_Count', counts various special characters that are, @, , \_, %, and query components which shows that there are some other unnecessary components of the URL. Finally, the 'DomainLength' was calculated as the ratio of the length of the domain name to the total length of the URL in the dataset to indicate the relative length of the domain part. Altogether, these preprocessing steps improved the dataset's strength and diversity, which aided in analysis and forecasting. The intention of the correlation heatmaps produced in the given code is to analyze the covariance between the features in the phishing detection data set and the goal variable, referred to as 'labels'. These types of heat maps assist in the determination of the strength and direction of relationships between features, remove or retain redundant features, and assist in the process of making feature selection vital for improved model performance mainly in predictive modeling. For instance, in the first heatmap, 'NumDots', 'SubdomainLevel', 'PathLevel', and 'UrlLength' are displayed, whereas the 'labels' have a perfect negative relationship with 'id'; -0.87. In the second heatmap, it becomes clear that if 'NumQueryComponents' has a high value then 'NumAmpersand' also most likely will have a high value 0.87 of correlation. The third heatmap presents a correlation matrix between features such as 'HttpsInHostname', 'HostnameLength', 'PathLength', and 'labels', in which 'PctExtHyperlinks' and 'PctExtResourceUrls' are positively related with 0.46. The fourth heatmap describes the correlation between 'InsecureForms', 'RelativeFormAction', 'ExtFormAction', and 'labels', and among those 'SubmitInfoToEmail' + 'FrequentDomainNameMismatch' has a strong negative correlation (-0.36). In summary, these heatmaps are useful for data discovery and to guide feature creation and selection that would improve the model's strength and credibility.



**Figure 4.2:** This symbolizes the Preprocessing of the dataset used in algorithms.



**Figure 4.3:** Some of the labels' names included in the heat map result are shown in the map.

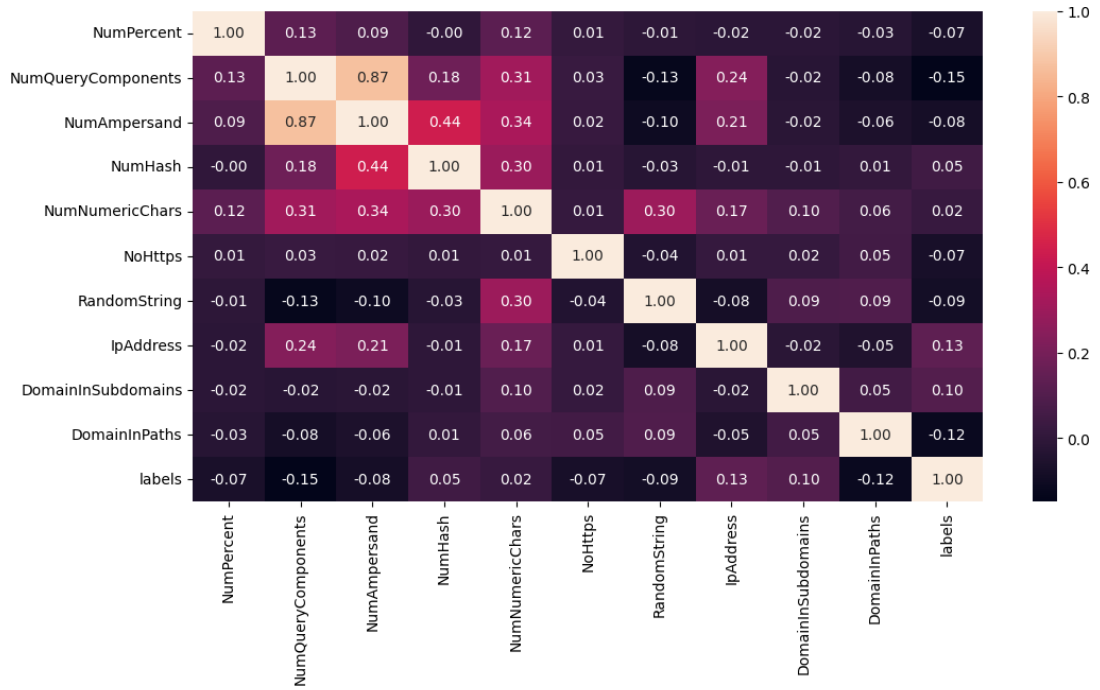


Figure 4.4: Some of the labels' names included in the heat map result are shown in the map.

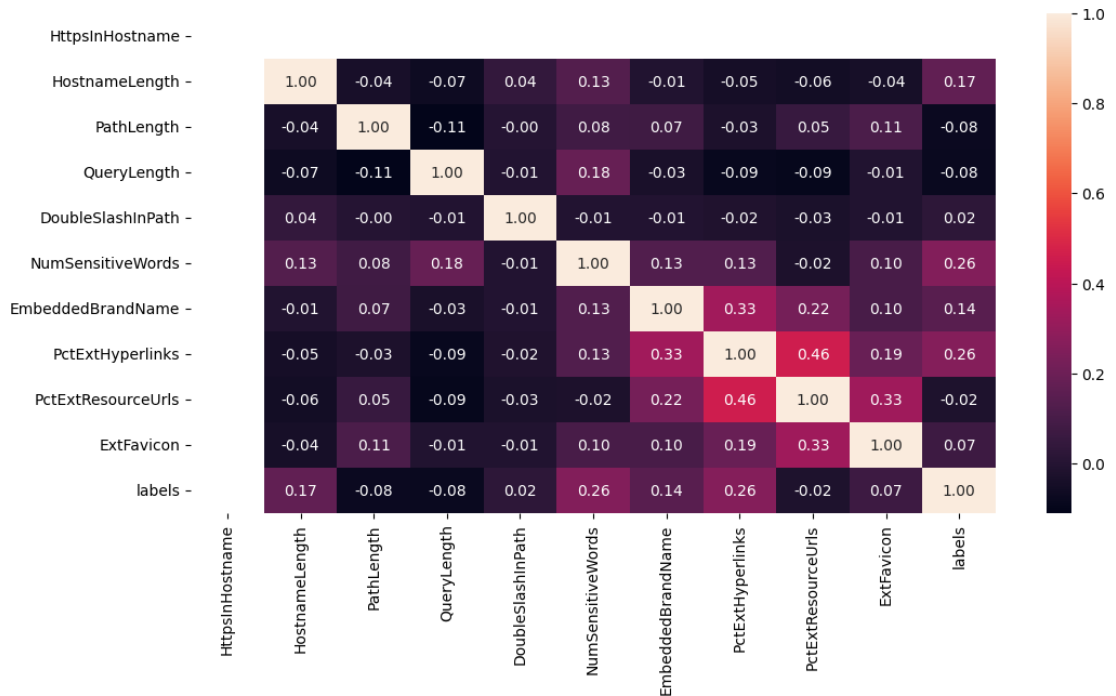


Figure 4.5: Some of the labels' names included in the heat map result are shown in the map.



Figure 4.6: Some of the labels' names included in the heat map result are shown in the map.

### 4.3 Architecture

In this study, we employ a Deep Belief Network (DBN) for the task of classification, utilizing a structured approach that combines unsupervised pre-training with subsequent supervised fine-tuning. The basic framework of the DBN is made up of layers of Restricted Boltzmann Machines Restricted Boltzmann Machines (RBMs), which are themselves the basic units of the structure that learn the features of the input data. First of all, each RBM layer is trained independently in an unsupervised way using the Contrastive Divergence, which approximates the log-likelihood gradient. This phase entails computing the conditional probabilities of the hidden units given the visible units and vice versa while utilizing the energy-based model that is characteristic of the RBMs. The conditional probabilities of the hidden units concerning the visible states are calculated in the sigmoid activation function of the sum of weights of the inputs plus the bias factor as given in the equation below 4.3.1, which enables the sampling of

the hidden states. After the unsupervised training, the RBMs are stacked in such a way that the hidden nodes of one RBM act as input nodes in the next RBM in a manner that a deeper level of features is gained at each layer of the network. This layered architecture allows the network to learn a hierarchical representation of the data, with more abstract features at higher layers. Once the layer-wise pre-training is completed, the entire network undergoes a supervised fine-tuning phase. During this phase, a final output layer, typically a softmax layer, is added to perform multi-class classification. The network is then trained end-to-end using backpropagation to minimize the classification error. This fine-tuning adjusts the weights and biases across all layers of the DBN, refining the feature detectors and aligning them more closely with the discriminative tasks at hand 4.3.2. The integration of both unsupervised pre-training and supervised fine-tuning harnesses the strengths of generative and discriminative approaches, aiming to boost the overall predictive performance of the model on structured data inputs. This methodology highlights a robust approach to leveraging deep architectures for complex classification tasks, addressing both the initial feature representation learning and the final task-specific optimization.

$$p(h_j = 1|v) = \sigma(b_j + \sum_i(v_i w_{ij})) \quad (4.3.1)$$

$$E(v, h) = -\sum_i(a_i v_i) - \sum_j(b_j h_j) - \sum_{(i, j)}(v_i w_{ij} h_j) \quad (4.3.2)$$

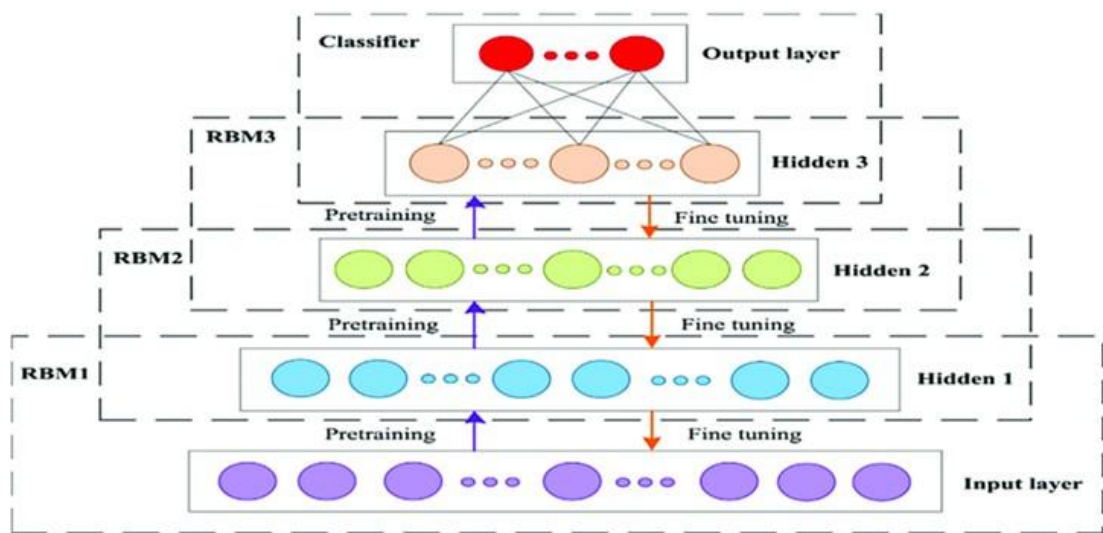


Figure 4.7: Explaining the components of SEAP architecture using a diagram.

Steps	Explanation	Input	Output	Details
1. Initialization of RBM Parameters	Each RBM is initialized with random weights and zero biases to prepare for training.	Number of visible ( $n_{vis}$ ) and hidden ( $n_{hid}$ ) units for each RBM.	Initialized RBM with weights (W) and biases ( $v_{bias}, h_{bias}$ ).	Weights W initialized using a normal distribution, biases initialized to zero.
2. Visible to Hidden Sampling ( $v, o_h$ )	Calculates hidden unit probabilities from visible units using a sigmoid function, then samples binary states for the hidden units.	Visible unit activations ( $v$ ).	Sampled hidden unit activations.	$p(h/v) = \sigma(W_v + h_{bias})$ where $\sigma$ is the sigmoid function. Sampling is done using Bernoulli
3. Hidden to Visible Sampling ( $h, o_v$ )	Computes visible unit probabilities from hidden units and reconstructs the visible units' binary states.	Hidden unit activations ( $h$ ).	Reconstructed visible unit activations.	$p(v/h) = \sigma(W^T h + v_{bias})$ Reconstructed using Bernoulli sampling
4. Forward Pass of an RBM	Processes the visible units through the RBM to hidden units and back to visible units, useful during unsupervised pre-training.	Original visible units.	Reconstructed visible units.	Forward pass involves two steps: $h = \text{sample}_{from}(\sigma(W_v + h_{bias}))$ , $v' = \sigma(W^T h + v_{bias})$

5. Stacking RBMs into a DBN	Layers of RBMs are stacked where each layer's output serves as the next layer's input, allowing the network to learn hierarchical features.	Input data (e.g., flattened image vectors).	Final hidden unit activations after all RBM layers.	Each layer's output $h$ of layer $n$ becomes the input to layer $n+1$ . This is repeated for each stacked RBM.
6. Classification Layer	The final layer of hidden units is fed into a linear classifier which computes raw scores for each class.	Final hidden unit activations from the last RBM layer.	Raw class scores.	Linear transformation: $scores = W_{(classifier)}h_{final} + b_{classifier}$
7. Log Softmax Activation	Converts the raw classification scores into log probabilities which are suitable for computing loss during training.	Scores from the classifier.	Log probabilities for each class.	Log Softmax operation: $\log(\text{softmax}(\text{scores}))$ , where Softmax is defined as: $Softmax(x_i) = \frac{\exp(x_i)}{(\sum_j \exp(x_j))}$

**Table 4.3:** Algorithm 1: SEAP Implementation for feature extraction.

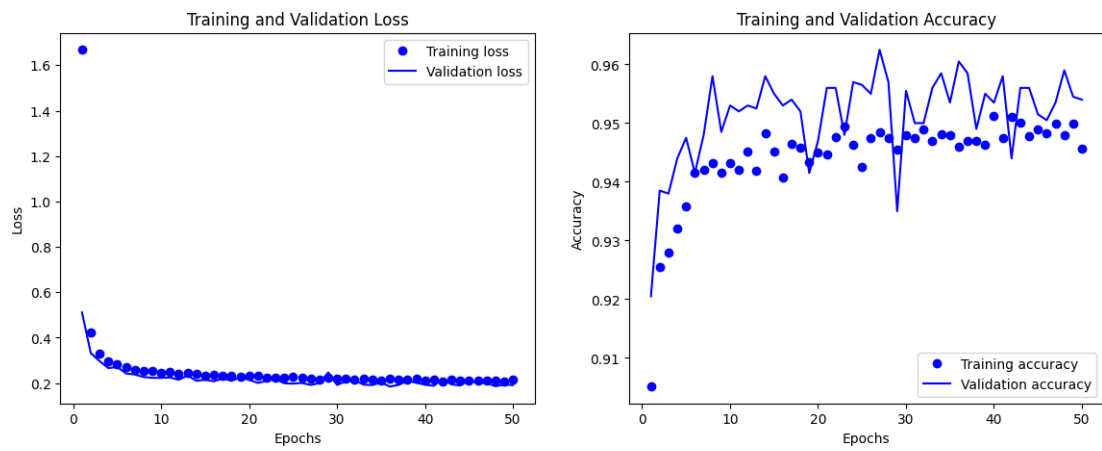


## CHAPTER 5

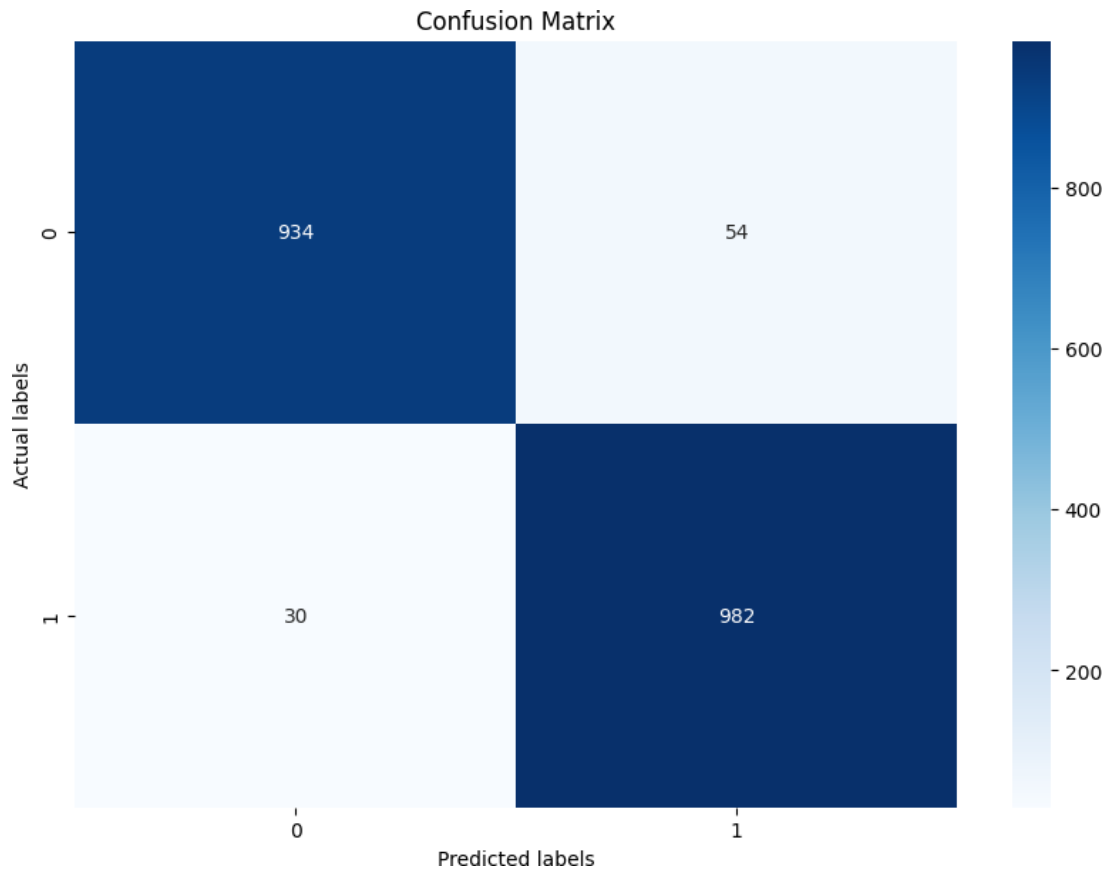
# Results

To test the efficiency of the proposed approach, the set of experiments was carried out using the phishing detection dataset with different instances of legitimate and phishing activities. Some changes were made to the data to facilitate its usage; all datatypes were converted for better run time and some of the features were normalized based on M1 scores. As a result of such selection, the top 30 features with the greatest M1 scores were chosen and normalized for the subsequent calculations. The data was then reshaped to meet the input of the Conv1D neural network model that was defined and trained using TensorFlow/Keras. We start with two Conv1D layers, followed by a dropout and two dense layers. The model was split 80/20 on the data. The trained model checked its accuracy on the test set with a very high accuracy of 96%. Other performance measures, which are a classification report, revealed precision, recall, and f1-Score of 0.96 across both classes. The classification report released described that in class 0 a precision of 0.97, a recall of 0.95, and an f1-score of 0.96 while class 1 achieved a precision of 0.95, a recall of 0.96, an average precision of 0.95, a recall of 0.97, and f1-score of 0.96 while the overall support stands at 2000 samples. The findings were subsequently, presented by using M1 score plots and confusion matrices for an enhanced understanding of the model's performance. The Conv1D model also showed convincing and reliable results concerning phishing detection and

thus, it can be concluded that the use of this model in the context of the real-world application in the problem of phishing detection can be effective.



**Figure 5.1:** Stands for the training and validation loss and accuracy.



**Figure 5.2:** Stands for the output of the phishing detection system.

## CHAPTER 6

# Discussion

Social engineering attacks have become increasingly prevalent in the modern digital landscape, affecting individuals, organizations, and entire societies. These attacks are particularly insidious because they exploit human psychology and trust, manipulating unsuspecting victims into divulging sensitive information or performing actions that compromise their security. The effects of social engineering can be catastrophic, ranging from financial losses and data breaches to identity theft, operational interruptions, reputational damage, legal complications, and even psychological impacts. As digital communication becomes more integrated into everyday life, the sophistication and frequency of social engineering attacks have grown, underscoring the need for robust defenses and innovative research methodologies to combat these threats.

One of the major concerns in dealing with social engineering attacks is the lack of comprehensive models capable of effectively detecting and mitigating these threats. Existing models often fall short due to their limited ability to capture the complex, non-linear patterns associated with such attacks. Traditional security frameworks are primarily designed to address technical vulnerabilities, focusing on firewalls, encryption, and antivirus software. However, social engineering attacks circumvent these defenses by targeting the human element, exploiting behaviors

and emotions such as curiosity, fear, or urgency. This makes it imperative to develop models that go beyond mere technical countermeasures, incorporating advanced techniques like behavioral analytics, machine learning, artificial intelligence (AI), natural language processing (NLP), and threat intelligence to identify and neutralize social engineering threats.

The proposed SEAP (Social Engineering Attack Prevention) model addresses these shortcomings by leveraging a combination of unsupervised and supervised learning methods. The SEAP model uses a novel approach, involving pre-training with unsupervised learning followed by fine-tuning with supervised learning, to enhance its feature extraction capabilities. This methodology allows the model to detect intricate patterns that simpler models might miss, making it more effective at identifying the subtle indicators of social engineering attacks. The use of ConvMix blocks, a key component of the SEAP architecture, enables the model to capture multi-dimensional data representations, thus improving its accuracy and robustness. Moreover, the lightweight nature of the model ensures that it can be efficiently trained and fine-tuned even with large and complex datasets, making it suitable for real-world applications in anti-phishing and cybersecurity systems.

In addition to its technical sophistication, the SEAP model's design also considers practical deployment scenarios. By integrating behavioral analytics and NLP, the model can analyze user activity and communication patterns to detect anomalies indicative of social engineering attempts. For example, NLP-based analysis can identify linguistic cues associated with phishing emails or suspicious requests, while behavioral analytics can monitor deviations from typical user behavior, such as unusual login locations or unexpected data access patterns. This holistic approach significantly enhances the model's ability to detect and prevent a wide range of social engineering attacks, from phishing and vishing to more complex schemes like spear phishing and baiting. The versatility of the SEAP model makes it a valuable tool for organizations looking

to strengthen their defenses against ever-evolving social engineering tactics.

The research conducted in this domain is crucial because it not only contributes to the academic understanding of social engineering attacks but also has real-world implications for cybersecurity. By developing models like SEAP, researchers aim to provide organizations with practical tools that can be integrated into existing security infrastructures. The experimental results presented in the study demonstrate the model's effectiveness, achieving a high accuracy rate of 96% on a phishing detection dataset. This performance is a significant improvement over conventional methods, which often struggle to achieve similar results due to the complexity and variability of social engineering attacks. The precision, recall, and F1-score metrics further validate the model's ability to accurately classify malicious activities, thereby reducing the risk of false positives and negatives.

One of the key strengths of the SEAP model is its adaptability. The model's architecture is designed to accommodate new data inputs and evolving attack strategies, making it resilient against emerging threats. This is particularly important given the dynamic nature of social engineering attacks, where attackers are constantly refining their tactics to bypass security measures. By continuously learning from new data and incorporating insights from threat intelligence feeds, the SEAP model can stay ahead of attackers, providing a proactive defense against both known and unknown threats. This adaptability is enhanced by the model's use of deep learning techniques, which allow it to generalize from existing patterns and recognize previously unseen attack vectors.

The study also highlights the importance of feature engineering and data preprocessing in improving the performance of machine learning models. The use of techniques like feature scaling, normalization, and correlation analysis helps to refine the input data, ensuring that the model can effectively learn from the relevant features while minimizing the impact of noise and irrel-

evant information. In the context of the SEAP model, these preprocessing steps were critical in transforming the dataset, which contains over 10,000 records and 50 features, into a suitable format for training and evaluation. By selecting the top 30 features based on mutual information scores, the researchers were able to optimize the model's performance without sacrificing accuracy or interpretability.

The thesis also makes a significant contribution by comparing the SEAP model with existing approaches in the field of cybersecurity. Techniques such as evolutionary algorithms for cyber-attack detection, semi-supervised clustering for DDoS botnet detection, CFG analysis for IoT malware detection, and obfuscation feature analysis for metamorphic virus detection are reviewed in detail. While these methods have been successful in addressing specific types of cyber threats, they lack the flexibility and comprehensiveness required to tackle the full spectrum of social engineering attacks. The SEAP model addresses this gap by providing a unified framework that can handle diverse attack scenarios, making it a more versatile and effective solution.

Another noteworthy aspect of the study is its emphasis on real-world applicability. The SEAP model is designed to be lightweight and scalable, making it feasible for deployment in various settings, from small businesses to large enterprises. The model's ability to operate efficiently on limited computational resources ensures that it can be implemented even in environments with constrained hardware capabilities. This is a crucial consideration for organizations that may not have access to high-performance computing resources but still need to protect themselves from sophisticated social engineering attacks.

The discussion section of the thesis effectively ties together the theoretical and practical aspects of the research, providing a comprehensive overview of the challenges, solutions, and future directions in the field of social engineering attack prevention. By integrating cutting-edge tech-

nologies with practical considerations, the study offers a valuable roadmap for researchers and practitioners alike. The SEAP model represents a significant step forward in the fight against social engineering, demonstrating that it is possible to develop effective defenses by combining advanced machine learning techniques with domain-specific insights.

In conclusion, the increased prevalence of social engineering attacks in today's digital environment necessitates a proactive and multifaceted approach to cybersecurity. The SEAP model, with its innovative use of deep learning, behavioral analytics, NLP, and threat intelligence, provides a promising solution to this growing problem. By addressing the limitations of traditional frameworks and incorporating advanced detection techniques, the model offers a comprehensive defense against a wide range of social engineering threats. As attackers continue to refine their tactics, ongoing research in this field will be essential to stay one step ahead and protect individuals and organizations from the devastating consequences of social engineering attacks. The findings of this research have the potential to significantly enhance the security of digital systems, making them more resilient to both current and future threats.



## CHAPTER 7

# Conclusion and Future Work

The conclusion of this thesis draws attention to the critical importance of addressing the rising threats posed by social engineering attacks in the digital world. Social engineering attacks leverage human psychology and the intrinsic trust built within individuals and organizations to manipulate and exploit users into revealing confidential information or performing actions detrimental to their security. The consequences of these attacks are far-reaching and diverse, impacting financial stability, organizational reputation, personal privacy, and even mental health. Given the scale and sophistication of these threats, there is a pressing need for comprehensive detection and countermeasure strategies that can effectively mitigate the risk posed by social engineering. This research has made significant contributions towards this goal by proposing the SEAP (Social Engineering Attack Prevention) model, which leverages deep learning and innovative architectures to enhance the detection capabilities of social engineering attacks.

The SEAP model, designed using a combination of unsupervised pre-training and supervised fine-tuning, offers a robust framework for identifying complex, non-linear patterns in social engineering attacks that are often missed by traditional models. The integration of ConvMix blocks within the architecture enhances the model's ability to process large datasets and extract

meaningful features, making it more effective in detecting subtle indicators of phishing and other social engineering schemes. The evaluation of the SEAP model on a real-world phishing detection dataset, demonstrated its superior performance, achieving an impressive accuracy rate of 96

The research presented in this thesis not only showcases the effectiveness of the SEAP model but also highlights the broader implications for the field of cybersecurity. By demonstrating that complex social engineering attacks can be effectively countered using advanced machine learning techniques, this research opens the door for further exploration into more sophisticated and adaptive defense mechanisms. The use of deep learning, in particular, provides a pathway for developing models that are not only capable of detecting known attack patterns but also possess the ability to generalize and recognize previously unseen tactics. This adaptability is crucial in a threat landscape where attackers are constantly refining their strategies to bypass existing defenses.

However, despite the successes achieved in this research, there are still several areas that warrant further exploration and development. One of the primary limitations identified is the reliance on labeled datasets for supervised learning. While the SEAP model has been shown to perform well on structured datasets, the scarcity of high-quality, labeled data for certain types of social engineering attacks remains a challenge. Future research should focus on developing semi-supervised or unsupervised models that can learn from unlabeled data, thereby expanding the range of threats that can be detected without relying on extensive manual labeling. Additionally, enhancing the model's ability to detect multi-modal attacks, which combine multiple forms of social engineering such as email phishing, voice-based phishing, and in-person tactics, is another promising avenue for future work.

Another key area for future research is the integration of real-time threat intelligence into the

SEAP model. While the current implementation leverages static datasets for training and evaluation, incorporating dynamic threat intelligence feeds can significantly improve the model's responsiveness to emerging threats. Real-time data can provide valuable context about new phishing campaigns, evolving attack vectors, and the latest tactics used by cybercriminals, enabling the SEAP model to adapt its detection strategies accordingly. This would involve developing a pipeline for continuously updating the model's training data and fine-tuning its parameters based on the latest threat information. Implementing such a system would not only enhance the model's detection capabilities but also ensure that it remains relevant and effective in a rapidly changing threat environment.

The thesis also underscores the need for developing models that are both accurate and explainable. While deep learning models such as the SEAP model are highly effective in detecting complex patterns, their "black-box" nature often makes it difficult to interpret the reasoning behind their predictions. This lack of interpretability can be a significant barrier to adoption in critical sectors such as finance, healthcare, and government, where trust and transparency are paramount. Future research should focus on integrating explainability techniques, such as attention mechanisms or feature attribution methods, into the SEAP model to provide insights into which features or data points contributed to a particular classification. This would not only increase user trust in the model's predictions but also help security analysts understand and respond to new attack patterns more effectively.

Another promising direction for future work is the exploration of multi-agent systems and reinforcement learning for social engineering detection. By simulating interactions between attackers and defenders in a controlled environment, reinforcement learning models can be trained to develop optimal defense strategies in response to various attack scenarios. This approach could lead to the development of adaptive models that can anticipate attacker behavior and preemp-

tively deploy countermeasures, significantly reducing the impact of social engineering attacks. Furthermore, the use of multi-agent systems can enable collaboration between different models, each specializing in a specific type of attack or defense, to create a more holistic and resilient defense framework.

In addition to the technical advancements discussed, future research should also consider the human factors involved in social engineering attacks. While machine learning and AI can provide powerful tools for detecting malicious behavior, human awareness and training remain critical components of an effective defense strategy. Research should explore the integration of human-in-the-loop systems, where AI models work in conjunction with human analysts to provide real-time recommendations and feedback. This collaborative approach can leverage the strengths of both human intuition and machine precision, leading to more accurate and context-aware detections.

Moreover, expanding the scope of research to address the psychological and sociological aspects of social engineering can provide valuable insights into the motivations and tactics used by attackers. Understanding the psychological triggers that make individuals susceptible to manipulation can inform the development of more targeted awareness and training programs, helping users recognize and respond to social engineering attempts more effectively. Such interdisciplinary research, combining elements of psychology, sociology, and cybersecurity, has the potential to significantly enhance the overall effectiveness of social engineering defenses.

Finally, future research should aim to validate the SEAP model in diverse operational environments to assess its scalability and generalizability. While the model has demonstrated strong performance on a specific phishing dataset, its effectiveness in different contexts, such as detecting spear-phishing or multi-step social engineering campaigns, remains to be tested. Conducting extensive field tests in various organizational settings, including finance, healthcare, and critical

infrastructure, would provide valuable insights into the model's strengths and limitations. This would also help identify areas where further optimization is needed, paving the way for the development of a universally applicable social engineering detection system.

In conclusion, the research presented in this thesis represents a significant step forward in the fight against social engineering attacks. The SEAP model's innovative architecture and high performance provide a strong foundation for future work in this area. However, ongoing research and development are essential to address the evolving nature of social engineering threats and to build more resilient and adaptable defense systems. By exploring new methodologies, incorporating real-time intelligence, and enhancing model interpretability, future research can continue to advance the state-of-the-art in social engineering detection and prevention, ultimately contributing to a safer and more secure digital environment for all.

# Bibliography

- [1] P.A. Lawson, A.D. Crowson, and C.B. Mayhorn. “Baiting the Hook: Exploring the Interaction of Personality and Persuasion Tactics in Email Phishing Attacks”. In: *Springer* (2018).
- [2] N. Abe and M. Soltys. “Deploying Health Campaign Strategies to Defend Against Social Engineering Threats”. In: *Procedia Computer Science* 159 (2019), pp. 824–831.
- [3] R. Banu, A. Kamath, and H.S. Ujwala. “Detecting Phishing Attacks Using Natural Language Processing and Machine Learning”. In: *ICCS* (2019).
- [4] J. Hatfield. “Virtuous Human Hacking: The Ethics of Social Engineering in Penetration-Testing”. In: *Computers and Security* 83 (2019).
- [5] M. Lansley, N. Polatidis, and S. Kapetanakis. “SEADer: A Social Engineering Attack Detection Method Based on Natural Language Processing and Artificial Neural Networks”. In: *Computational Collective Intelligence* (2019), pp. 686–696. DOI: [10.1007/978-3-030-28377-3\\_57](https://doi.org/10.1007/978-3-030-28377-3_57).
- [6] T. Lin, D.E. Capecchi, and D.M. Ellis. “Susceptibility to Spear-Phishing Emails: Effects of Internet User Demographics and Email Content”. In: *ACM Transactions on Computer-Human Interaction (TOCHI)* 26.5 (2019).

## BIBLIOGRAPHY

- [7] Ponemon Institute LLC and Accenture. “Ninth Annual Cost of Cybercrime Study”. In: *Accenture* (2019).
- [8] F. Salahdine and N. Kaabouch. “Social Engineering Attacks: A Survey”. In: *Future Internet* 11.4 (2019), p. 89. DOI: [10.3390/fi11040089](https://doi.org/10.3390/fi11040089).
- [9] S.M. Albladi and G.R.S. Weir. “Predicting Individuals’ Vulnerability to Social Engineering in Social Networks”. In: *Cybersecurity* (2020).
- [10] D. Jampen, G. Gur, and T. Sutter. “Don’t Click: Towards an Effective Anti-Phishing Training, a Comparative Literature Review”. In: *Humancentric Computing and Information Sciences* (2020).
- [11] M. Lansley, S. Kapetanakis, and N. Polatidis. “SEADer++ v2: Detecting Social Engineering Attacks Using Natural Language Processing and Machine Learning”. In: *2020 International Conference on Innovations in Intelligent Systems and Applications (INISTA)*. 2020. DOI: [10.1109/INISTA49547.2020.9194623](https://doi.org/10.1109/INISTA49547.2020.9194623).
- [12] M. Lansley, F. Mouton, and S. Kapetanakis. “SEADer++: Social Engineering Attack Detection in Online Environments Using Machine Learning”. In: *Journal of Information and Telecommunication* 4 (2020), pp. 346–362. DOI: [10.1080/24751839.2020.1747001](https://doi.org/10.1080/24751839.2020.1747001).
- [13] Proofpoint Annual Report. “2020 State of the Phish: An In-Depth Look at User Awareness, Vulnerability and Resilience”. In: *Proofpoint* (2020).
- [14] Z. Wang, L. Sun, and H. Zhu. “Defining Social Engineering in Cybersecurity”. In: *IEEE Access* 8 (2020), pp. 85094–85115. DOI: [10.1109/ACCESS.2020.2992807](https://doi.org/10.1109/ACCESS.2020.2992807).
- [15] A.A. Alsufyani and S.M. Alzahrani. “Social Engineering Attack Detection Using Machine Learning: Text Phishing Attack”. In: *INDJCSE* 12 (2021).

## BIBLIOGRAPHY

- [16] K.J. Mridha, S.D. Hasan, and A. Ghosh. “Phishing URL Classification Analysis Using ANN Algorithm”. In: *2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON)*. 2021.
- [17] Proofpoint Annual Report. “2021 State of the Phish: An In-Depth Look at User Awareness, Vulnerability and Resilience”. In: *IntelligentCIO* (2021).
- [18] Shashwat. *Phishing Dataset for Machine Learning*. 2021. URL: <https://www.kaggle.com/datasets/shashwatwork/phishing-dataset-for-machine-learning>.
- [19] Z. Wang, H. Zhu, and P. Liu. “Social Engineering in Cybersecurity: A Domain Ontology and Knowledge Graph Application Examples”. In: *Cybersecurity* 4.1 (2021). DOI: [10.1186/s42400-021-00094-6](https://doi.org/10.1186/s42400-021-00094-6).
- [20] Z. Wang, H. Zhu, and L. Sun. “Social Engineering in Cybersecurity: Effect Mechanisms, Human Vulnerabilities, and Attack Methods”. In: *IEEE Access* (2021).
- [21] K. Chetioui, B. Bah, and A.O. Alami. “Overview of Social Engineering Attacks on Social Networks”. In: *Procedia Computer Science* 198 (2022), pp. 656–661.
- [22] S. Mondal, S. Ghosh, and A. Kumar. “Spear Phishing Detection: An Ensemble Learning Approach”. In: *Data Analytics, Computational Statistics, and Operations Research for Engineers* (2022).
- [23] W. Syafitri, Z. Shukur, and U. Mokhtar. “Social Engineering Attacks Prevention: A Systematic Literature Review”. In: *IEEE Access* 10 (2022), pp. 39325–39343. DOI: [10.1109/ACCESS.2022.3162594](https://doi.org/10.1109/ACCESS.2022.3162594).
- [24] N. Yathiraju, G. Jakka, and S.K. Parisa. “Cybersecurity Capabilities in Developing Nations and Its Impact on Global Security: A Survey of Social Engineering Attacks and Steps for Mitigation of These Attacks”. In: *Cybersecurity Capabilities in Developing Nations and Its Impact on Global Security* (2022).



## BIBLIOGRAPHY

- [25] M.F. Alghenaim, N.A.A. Bakar, and F.A. Rahim. “Awareness of Phishing Attacks in the Public Sector: Review Types and Technical Approaches”. In: *Proceedings of the 2nd International Conference on Emerging Technologies and Intelligent Systems (ICETIS)*. 2023.