Real-Time Phishing URL Detection Using Machine Learning Techniques

By

SAAD UL HAQ

NATIONAL UNIVERSITY OF MODERN LANGUAGES ISLAMABAD

September, 2024

Real Time Phishing URL Detection Using Machine Learning Techniques

By

SAAD UL HAQ

MSEE, National University of Modern Languages, Islamabad, 2024

A THESIS IN PARTIAL FULLFILMENT OF THE REQUIRMENT FOR THE DEGREE

OF

MASTER OF SCEINCE

In Electrical Engineering

To

FACULTY OF ENGINEERING AND COMPUTING

MSEE, NATIONAL UNIVERSITY OF MODERN LANGUAGES, ISLAMABAD,

SEPTEMBER 2024

NATIONAL UNIVERSITY OF MODERN LANGUAGES FACULTY OF ENGINEERING AND COMPUTING

THESIS AND DEFENSE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defense, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering computing.

Thesis Title: Real Time Phishing URL detection Using Machine learning Techniques

Submitted by: Saad Ul Haq Registration #: NUML-S21-009

Master of Science in Electrical Engineering

Electrical Engineering

Discipline

Dr. Madah Ul Mustafa

Research Supervisor

Signature of Supervisor

Dr. Sheraz Alam Khan

Research Co-Supervisor

Signature of Co- Supervisor

Dr. Farhan Sohail	
HOD (EE)	Signature of HOD (EE)
Dr. Noman Malik	
Dean (FEC)	Signature of Dean (FEC)

Septemebr-4th-2024

Date

AUTHOR'S DECLARATION

I Saad Ul Haq
Son of Zia Ul Haq
Registration # NUML-S21-009
Discipline Electrical Engineering

Candidate of **Master of Science in Electrical Engineering (MSEE)** at the National University of Modern Languages do hereby declare that the thesis **Real Time Phishing URL Detection using Machine learning Techniques** submitted by me in partial fulfillment of MSSE degree, is my original work, and has not been submitted or published earlier. I also solemnly declare that it shall not, in future, be submitted by me for obtaining any other degree from this or any other university or institution. I also understand that if evidence of plagiarism is found in my thesis/dissertation at any stage, even after the award of a degree, the work may be cancelled and the degree revoked.

Signature of Candidate

Saad Ul Haq
Name of Candidate

4th Septemebr, 2024
Date

Dedication

"To my father, who taught me that the best kind of knowledge is that which is learned for its own sake, and to my mother, who showed me that even the largest task can be accomplished when taken one step at a time, this thesis is dedicated."

Acknowledgments

For the successful completion of this project, We are thankful to Almighty Allah, for enabling us to complete this project and making everything possible for the project to be a success.

We would like to express our sincere gratitude to our project supervisor, Dr.Madah Ul Mustafa for his sincere guidance, successive cooperation and useful suggestions.

We are thankful to all the "faculty of engineering and computing" for providing valuable information and for helping us. We are also thankful to the National University of Modern Languages Islambad for providing a platform to successfully complete this project.

Abstract

Real-time phishing Uniform Resources Locator URL detection is important due to the growing threat of phishing attacks on individuals and businesses. These attacks seek usernames, passwords, and credit card numbers. Fake emails and websites enable these attacks. The consequences can include money loss, identity theft, and reputation damage. Real-time phishing URL detection systems that use machine learning can reduce these risks. These technologies detect phishing websites by analyzing URLs and content. They quickly block these websites to prevent harm. This method helps adapt to evolving phishing attacks. In this research, we proposed a hybrid model that uses convolutional neural network CNN and long short-term memory networks LSTM. CNNs are used to predict phishing URL attacks using several feature engineering methodologies, while LSTM works on classification. This technology produces a more precise model than previous methods. Precision reduces false positives, preventing genuine websites from being misinterpreted for phishing attacks. The research effectively addresses phishing attacks by implementing a real-time detection system that boosts security and mitigates cyber dangers.

Keywords: Phishing attacks, URL detection, Deep learning, Risk reduction, Cybersecurity, CNN, LSTM.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

1 INTRODUCTION

1.1 Overview

1.1.1. Cybersecurity:

Cybersecurity is a critical aspect of protecting computers, networks, programs, and data from cyberattacks, which are often aimed at stealing sensitive information, extorting money from users, or disrupting business operations. Cybersecurity measures are employed to safeguard data centers and other computerized systems from unauthorized access and to ensure the integrity, confidentiality, and availability of data [1]. Phishing remains one of the most prevalent methods used by cybercriminals to target individuals and businesses. In phishing attacks, malicious actors create URLs that mimic legitimate websites to trick users into divulging their login credentials, financial information, or other sensitive data. The sophistication of phishing techniques has evolved, making it increasingly difficult to distinguish between legitimate and fraudulent URLs.

Identifying and blocking phishing URLs is critical for several reasons. Firstly, it helps protect users' personal and financial information from being compromised. This is particularly important in an era where identity theft and financial fraud can have devastating consequences. Secondly, it safeguards businesses from potential data breaches that can result in significant financial losses, legal liabilities, and damage to reputation. Thirdly, it contributes to the overall security of the internet by reducing the prevalence of malicious sites [2].

To enhance the effectiveness of phishing detection, machine learning and deep learning techniques are increasingly being employed. These techniques can analyze large datasets of URLs and learn to identify patterns and features that are indicative of phishing attempts. By continuously updating the models with new data, they can adapt to the evolving tactics used by cybercriminals. In addition to technical solutions, educating users about the risks of phishing and how to recognize suspicious URLs is an essential component of a comprehensive cybersecurity strategy. This includes promoting best practices such as verifying the authenticity of websites, using secure connections (HTTPS), and being cautious with unsolicited emails or messages [3]. As phishing attacks continue to pose a significant threat, the importance of robust cybersecurity measures, including advanced detection algorithms and user education, cannot be overstated. By combining technological solutions with informed user behavior, it is possible to create a safer online environment and mitigate the risks associated with phishing and other cyber threats.

1.1.2. Cybersecurity Issues

One of the top challenges for organizations today is Cyber Security. A complex feature about which many questions can be raised. These are complicated challenges, vast in scope and nuanced to require a suite of measures if sensitive information is goingto be protected and technological infrastructure kept intact. A data breach is a large or unauthorized release of the security compromising sensitive, secret and business critical information. Causing financial losses, reputational harm and legal burdens for individuals & organizations alike. They can ruin the systems and steal data malware attacks like viruses, worms & ransomwares etc. They are attacks that can be used to shut down vital infrastructure, cause a breach of data and require an enormous effort in remediation.

Phishing and social engineering rely on human psychology to funnel unsuspecting victims into giving up sensitive personal data or even their logins. Said tactics are evolving all the time, which I guess outlines how hard it is to effectively protect against them. Insider: these are threats from employees or legitimated users of a system that abuses their privileges to perform malicious acts. Unfortunately, as the attackers are authorized, these remain some of the most difficult threats to detect and prevent. APTs, on the other hand, are much more sophisticated — managed by seasoned attackers (often state entities or organized crime) for prolonged periods of time. Similarly, APTs (Advanced Persistent Threat) target infiltration of networks to dwell in the network and remain hidden for longer periods [2].

The proliferation of Internet of Things (IoT) devices has introduced new vulnerabilities, as many of these devices lack robust security measures. This makes them easy targets for attackers seeking to exploit weak points in a network. Supply chain attacks target interconnected networks, exploiting vulnerabilities in third-party vendors or software to compromise the security of the primary organization. These attacks can have far-reaching consequences, affecting multiple entities within the supply chain. To address these cybersecurity issues, a multi-layered approach is necessary. Technical safeguards such as firewalls, encryption, and intrusion detection systems are essential for protecting digital assets. Security awareness training for employees and stakeholders helps build a culture of cybersecurity and reduces the risk of human error. Risk management practices involve identifying, assessing, and mitigating risks to prioritize security efforts effectively. Engaging stakeholders, including employees, customers, and partners, ensures a collective commitment to cybersecurity and enhances the overall resilience of the organization. Cybersecurity issues are complex and ever-evolving, requiring a proactive and comprehensive approach to safeguard digital systems and data. By implementing technical safeguards, promoting security awareness, managing risks, and fostering stakeholder engagement, organizations can mitigate risks and enhance their cybersecurity resilience in the face of diverse threats.

1.1.3. Cybersecurity Virus Types:

These are the computer programs that replicate themselves and infect other files/programs of a Computer means Viruses, which may harm your system. Meanwhile, worms are basically malware designed to self-replicate and spread over networks — using their own initiative instead of relying on human intervention like a Trojan above — taking advantage of software or operating system vulnerabilities. Trojans: Trojans get their name from the Trojan Horse that was used to trick its way into Troy, by masquerading as something desired. After deployed, they allow cybercriminals to get hold of victims devices illegitimately and either take critical information or do destruction. Ransomware is a malware kind that encrypts the files on the victim's computer while asking for payment, normally in cryptocurrencies to decrypt them. It is a type of malicious software that collects information about you, often without your knowledge or consent.

Adware, while not always malicious, inundates users with unwanted advertisements, which can lead to performance issues and a poor user experience. Defending against these diverse threats requires a comprehensive approach. This includes the use of firewalls to block unauthorized access, antivirus software to detect and remove malware, and intrusion detection systems to monitor network traffic for suspicious activity. Additionally, educating users about good cybersecurity practices, such as being cautious with email attachments and regularly updating software, is crucial in raising awareness and preventing the spread of malware.

First, it prevents identity theft by protecting personal and organizational data. By preventing hackers from stealing important data and payments, it reduces financial losses. Detecting and blocking phishing URLs prevents malware infections and data breaches, protecting computer systems and networks. By preventing successful phishing attacks, it protects businesses and organizations' reputations [1].

Phishing assaults are difficult owing to their social engineering aspect, even though malware detection has improved. Using the internet's vastness and transitory harmful websites, phishing attempts proliferate. Despite heightened awareness of cyber dangers, many firms are poorly protected, leaving important data vulnerable to breaches and abuse. Our research suggests using deep learning models for URL analysis to improve accuracy and reduce phishing assaults. These models automate the detection of dangerous web pages in real time, strengthening cybersecurity against evolving cyber threats. As the cyber landscape becomes more hostile, proactive techniques like deep learning URL analysis are needed to protect critical data and maintain digital ecosystem confidence.

The frameworks that we employ today have already reached a point where they are able to differentiate between malware with an unusually high degree of precision. Through their efforts, we were able to effectively remove the human element from the scenario, which ultimately led to the emotional decline of websites that facilitate infection. On the other hand, thanks to the social engineering component of phishing, it is more difficult to achieve the same results using phishing websites. It is possible that this fact is the primary rationale behind why phishing endeavors continue to increase in number. Boycotts were a feasible approach for monitoring fraudulent websites in the past, when the Internet was still in its infancy. The perpetrators of phishing attacks, on the other hand, are ready to overwhelm the internet with phishing attempts that are only temporary. Due to the fact that they typically do not continue for a sufficient amount of time to make it on the boycott list, the brief life expectancy of such malicious endeavors might, in reality, play in favor of themselves [3].

The world is gauge to burn through $133.7 billion in 2022 on network safety. 62% of organizations experienced social designing assaults, include phishing assaults, and 68% of business pioneers feel their dangers connected with cybercrimes are expanding. Nonetheless, just 5% of organizations' envelopes on normal are fittingly safeguarded. 52% of breaks have included hacking as an assault vector, 28% involved malware, and 32-33% include phishing or social designing as an assault vector [4]. As the internet becomes increasingly fraught with malicious activities, there is a growing urgency to identify and detect malicious web pages. One promising method is URL analysis, which is capable of identifying phishing attacks, malware, and other forms of cyberattacks. However, the use of time-consuming lookups can lead to delays in real-time systems. To overcome this issue, our research proposes used deep ai learning models to increase the accuracy and give the better result to minimize the phishing attack.

1.2. Background

The whitelist and boycott records are the two records that are used in the list-down-based methodology. These records are used to organise legitimate and phishing URLs in a different order. Access to websites is granted in on the condition that the URL in question is included in the whitelist [5]. One strategy that is frequently utilised in the fight against phishing assaults is known as boycotting [6]. The Heuristic-Based method, which entails dissecting the structure of a phishing URL, is a strategy that has proven to be relatively successful. For instance, a current example of phishing URLs is utilised in order to classify URLs based on the association that they have with this example. When it comes to effectively recognising counterfeit websites, the methods that are utilised to address and address the elements of the URL play a crucial function.

Assessing the visual similarities between different web sites is how the visual likeness-based technique gets its results. Taking a server-side perspective of the pages is what determines whether or not a website is considered to be a phishing website [6]. Image processing is one of the many methods that are utilised in the process of identifying fake web pages. This procedure entails comparing the fake pages to the real content of the website. As a result of the fact that false pages are intended to be very similar to the real ones, even minute variations in the photos can be identified through the use of image processing techniques, which are not visible to the user. The content-based method, on the other hand, entails conducting an analysis of the material that is contained inside the pages. Using this method, features are extracted from the items that are present on the website as well as from third-party services like search engines and DNS service providers.

On the other hand, thanks to the social engineering component of phishing, it is more difficult to achieve the same results using phishing websites. It is possible that this fact is the primary rationale behind why phishing endeavours continue to increase in number. Boycotts were a feasible approach for monitoring fraudulent websites in the past, when the Internet was still in its infancy. The perpetrators of phishing attacks, on the other hand, are ready to overwhelm the internet with phishing attempts that are only temporary. It is also possible to use the findings of this category to provide assistance to end-users who fall under the first group. A human-focused strategy and a software-focused approach are the two basic methodologies that are included in the existing research on phishing detection. It is the goal of human-focused techniques to raise the level of awareness and understanding among users, so that they are better equipped to make judgements when they are presented with phishing efforts. On the other side, software-focused techniques work towards improving the efficiency of software-based solutions in detecting and stopping phishing assaults that are carried out.

Using a recognition approach that analyses the weights of words retrieved from URLs and HTML content is one method that the authors of [7] described as a tool for identifying websites that are used for phishing. In order to provide the impression that the URL is real, attackers may utilise these terms, which may contain brand names. The frequency of the terms and their position within the URLs are taken into consideration when determining the weights of the words. A comparison is made between the characteristics of the space name and the decision of whether or not the website is a phishing site. Phishing attacks have demonstrated remarkable variety in the face of several cautious endeavours; in addition, perpetrators continue to develop sophisticated phishing websites that are designed to impersonate authentic websites with great precision. One of the most important assumptions that must be made when utilising AI methods is that the process of gathering information for preparation is devoid of any activity carried out by the aggressors [8]. An in-depth analysis is performed using the approaches that are currently in use to identify common characteristics shared by phishing websites while also providing assistance in distinguishing them from genuine websites. Calculations performed by artificial intelligence make use of these traits, which form the basis of essential components. This strategy is illogical since adversaries use a variety of techniques to create a phishing site that is similar to a certified objective site, rather than other phishing instances. The competence of the model planner and the types of attacks that the calculation is able to identify are frequently taken into consideration when making decisions on the selection of elements and how they are depicted. In order to circumvent the existing knowledge models and render the pieces that are currently in use obsolete, adversaries are constantly looking for alternative attack routes. It is necessary to update the existing models whenever new developments come about. The numbers [9]-[12].

In the modern era, where technology has seen significant advancements and the volume of data generated by social networks, online activities, and Internet of Things (IoT) devices has skyrocketed, the importance of safeguarding data privacy, thwarting cyberattacks, and maintaining network security cannot be overstated. The digital landscape has evolved, leading to an increased attack surface for cybercriminals to exploit. Phishing, one of the oldest methods employed by attackers, remains a persistent threat despite the constant emergence of new tactics to gain unauthorized access to resources such as networks, programs, and data.

The COVID-19 pandemic has further exacerbated the situation, as a significant portion of work and business-related tasks have shifted to remote settings, with employees relying heavily on internet connectivity to perform their duties from home. This transition has expanded the attack surface for cybercriminals, as home networks and personal devices may not have the same level of security as corporate environments. The increased reliance on digital communication and collaboration tools also presents more opportunities for phishing and other cyberattacks. In response to these challenges, organizations and individuals must prioritize cybersecurity measures to protect their data and networks. This includes implementing robust security protocols, such as multi-factor authentication, regular software updates, and secure VPN connections for remote access. Additionally, raising awareness and providing education on cybersecurity best practices can empower users to recognize and avoid phishing attempts and other threats.

As technology continues to evolve and the digital landscape becomes more complex, the need for advanced cybersecurity solutions and strategies becomes more critical. Artificial intelligence and machine learning are increasingly being leveraged to enhance threat detection and response capabilities. Collaboration between governments, industries, and cybersecurity experts is also essential to develop and enforce regulations and standards that promote data privacy and security.Those who commit cybercrime are always improving their strategies and investigating the technological obstacles that they encounter when trying to protect data while they are away from their offices. While working from home, there has been an increase in the number of instances of data theft, fake emails, spam, and attempts to steal personal information. As a result of the epidemic, businesses, corporations, and employees were directed to perform their jobs away from the office. There has been a considerable rise in the number of cyberattacks as a result of cybercriminals taking advantage of the pandemic occurrence. As we go farther into the era of the Internet of Things (IoT), the number of devices that are connected to the internet continues to increase, and with it, the security risks associated with cyberattacks. According to research and an article that was published by Deloitte in Kuala Lumpur, it is stated that 91% of all cyberattacks begin with a phishing email to an unexpected victim, and 32 percent of all successful breaches involve the use of phishing techniques.

Despite the fact that thousands of phishing attacks have been carried out over the course of the years, people continue to fall prey to this most traditional form of cyberattack. Phishing is a type of fraud in which an attacker sends an email to a person while pretending that the email is coming from a reputable organisation in order to gain access to sensitive information such as login data and account information. Phishing happens when an attacker sends an email to anyone. In most cases, a person who falls victim to phishing is unaware that the email that was sent to them contains dangerous software or that it will link them to fraudulent websites in an attempt to fool them into giving personal or financial information such as account IDs and credit card numbers. Phishing is a form of cybercrime in which the perpetrators attempt to deceive victims into clicking on a malicious link that appears to be real [13]. The utilization of machine learning (ML) techniques as anti-phishing tools has been demonstrated through a series of studies that have been carried out over the course of several years [14]. They contact a customer and offer genuine support to assist them in fixing web-based issues or bank-related issues, thereby getting access to steal bank security codes, personal details, and a vast array of other information. This is a frequent tactic used in phishing assaults. The employment of classification algorithms such as ANN, KNN, and Decision Trees as a tactic to protect against phishing assaults has been discussed in previous research. The article [15] provides an overview of the five distinct stages that are involved in the lifecycle of a phishing assault. His research examines the structure of these attack phases, the traits that are associated with phishing victims, the hazards and vulnerabilities that are related to phishing, and the innovative phishing methods that are used.

1.3. Motivation

Phishing attacks are a serious threat to individuals and organizations alike, as they can result in financial loss, data breaches, and other security risks. Detecting these attacks in real-time is crucial for preventing their harmful effects. Machine learning techniques offer a promising approach to identifying and blocking phishing URLs before they can cause harm. By analyzing patterns and features of known phishing URLs, machine learning algorithms can quickly identify new ones and alert users or security systems. With the increasing prevalence and sophistication of phishing attacks, developing effective real-time detection deep learning is more important than ever. By harnessing the power of deep learning methods, we can better protect ourselves and our digital assets from these dangerous threats.

Phishing is an internet-based wrongdoing that attempts to deceive unsuspected clients into uncovering their delicate (and significant) individual data, for instance, usernames, passwords, monetary account subtleties, postage information, SSN, and social connections, to the lowlife, frequently for noxious reasons. Phishing is regularly executed by camouflaging as a reliable element in Web correspondence, which is accomplished by consolidating social designing and specialized stunts. Much of the time, the instruments taken advantage of by assailants incorporate conveying mocking messages and setting up misdirecting sites to allure clients to uncover data. The mocking messages normally imply to be from lawful organizations, planned to lead clients to fake sites that draw the client to enter delicate data.

Identifying URLs that are used for phishing is an important step in the process of stopping phishing attempts. The detection of phishing URLs has been accomplished through the development of a number of different technologies, such as blacklists, DNS filters, machine learning algorithms, and user awareness training. It is possible for individuals and organisations to protect their sensitive information and stay safe from phishing attempts with the assistance of these strategies.

1.4. Problem Statement

To steal passwords, phishers send suspicious URLs. Personal data is always at risk from phishing. Crooks target daily multi-website visitors. Malicious websites steal digital wallet and social media credentials [3]. This process is fast; therefore spam URLs must be clicked to validate. Recently, SVMs, random forests, and decision trees have identified phishing URLs. As phishing assaults get increasingly sophisticated, these simple approaches fail.

Despite their complex architecture and processing needs, deep learning models may enhance fraudulent URL identification [4], [7].

1.5. Aim and Objectives

1.5.1. Aim

The aim of this research is to develop an improved method for real-time phishing URL detection using machine learning techniques. This approach seeks to enhance the accuracy and efficiency of detecting phishing threats. The goal is to create a more reliable solution to protect users from malicious online threats.

1.5.2. Objectives

- To develop and implement a deep learning-based real-time phishing URL detection system to achieve high accuracy and minimize false positives, ensuring genuine websites are not misclassified as phishing attacks.
- To evaluate the proposed system's ability to adapt to evolving phishing tactics and enhance cybersecurity by quickly identifying and blocking phishing websites to mitigate cyber risks.
- To analyze the effectiveness of machine learning and deep learning algorithms in real-time phishing URL detection by examining their ability to analyze URLs and website content for identifying phishing websites.

Concluding this Chapter, the thesis provide an extensive introduction to cybersecurity specifically phishing attacks; they are a common and soon becoming outstanding threat for individuals as well as organizations. This chapter begins by defining what cybersecurity is describing the importance of this in securing computers, networks programs and data from cyberattacks. One such tactic that cybercriminals use significantly to steal sensitive through this method is phishing attack, which means they will make fake URLs for original legitimate sites. Outlining how phishing URLs can be blocked at critical points throughout the internet and illustrating why doing so is essential to protect personal, financial information and prevent data breaches for businesses while improving overall security on-line.

It then explores common cybersecurity problems in the chapter, which includes an extensive range of cyber threats now growing in both sophistication and scale since one main struggle for modern-day companies is its novelties. Data breaches, malware attacks, social engineering ploys and insider threats just to name a couple of with highlights nowadays on APTs (Advanced Persistent Threat robust ITIES). This chapter also focuses on the increased exposure with ubiquitous Internet of Things (IoT) after noting that such devices may easily be taken advantage of by attackers due to poor security implementation. These challenges can be mitigated by taking a layered approach involving both technical measures like firewalls and encryption, education in the form of security awareness training programs, appropriate risk management activities as well as active involvement from various stakeholders.

This chapter also talks about types of cybersecurity viruses: worms, Trojans, ransomware and adware which might pose different levels of threats to computer systems. This brings home the necessity for a more holistic approach to security, including firewalls, anti-virus software; intrusion detection systems and user training which should be implemented in order to lessen these risks. This chapter highlights how phishing detection plays a crucial role in identity theft, financial loss and reputational damage prevention as well. Since phishing uses more of social engineering based pieces, it makes the malware detection richer but in most cases harder to pinpoint a mailing that is malicious.

The background section reviews previous phishing detection methods (list-based, heuristic and visual), analyses URLs extracted from cached webpages to infer differences in pages structure/generation mechanisms of authentic sites/phishing landing page. It also covers content-based methods checking how the contents on a web page are in order to prevent phishing. It emphasizes how phishing techniques are changing and that even the best models, if not regularly refreshed with new threat data will eventually fall short. This chapter concludes that utilizing deep learning models for URL analysis will increase the precision of phishing detection thus decreasing its impact.

As phishing attacks are getting more sophisticated and common, how machine learning techniques can help in developing real-time detection prevention systems. Detecting patterns in URLs enables these systems to learn about new phishing pages fast and provide users with necessary protection against those threats. The problem statement illustrated in this chapter demonstrate that conventional ways like SVMs and decision trees fail to detect phishing URLs as these attacks become smarter. This chapter suggests a hybrid model that uses CNN and LSTM which is a type of deep learning models to combat huge amounts of data and improve the predictability on fraudulent URLs.

In the final part of this chapter, goals and objectives are presented to develop a real-time method for URL detection. This approach will use deep learning methods to offer precise detection outcomes and help improve the phishing security defense. This chapter, therefore, prepares a base up for thorough indulgence of deep learning approaches in phishing checking.

CHAPTER 2

2. RELATED WORK

2.1 Literature Review

Phishing attacks are a form of social engineering assault in which the perpetrators take advantage of users who are unaware of the danger they are posing to the system rather than focusing on vulnerabilities [15]. An attacker may, for instance, construct a website that is designed to look like the login page of a well-known email provider and then send the link to users, prompting them to input their login credentials. The user's lack of awareness of potential risks, which can lead to them being tricked by the attacker, is the source of the security issue in this scenario. The email service itself is not the source of the security problem. For more than a decade, researchers have been researching a variety of techniques to address this challenge. These approaches can be broadly categorized into two primary categories. The frameworks that we employ today have already reached a point where they are able to differentiate between malware with an unusually high degree of precision.

2.1.1. Traditional Techniques for Phishing URL

The social engineering component of phishing, it is more difficult to achieve the same results using phishing websites. It is possible that this fact is primary rationale behind why phishing endeavors continue to increase in number [17].

Boycotts were a feasible approach for monitoring fraudulent websites in the past, when the Internet was still in its infancy. The perpetrators of phishing attacks, on the other hand, are eager to flood the internet with shortlisting phishing endeavors. It is also possible to use the findings of this category to provide assistance to end-users who fall under the first group. A human-focused strategy and a software-focused approach are the two basic methodologies that are included in the existing research on phishing detection, as shown in Figure 1. It is the goal of human-focused techniques to raise the level of awareness and understanding among users, so that they are better equipped to make judgements when they are presented with phishing efforts. On the other side, software-focused techniques work towards improving the efficiency of software-based solutions in detecting and stopping phishing assaults that are carried out [33].

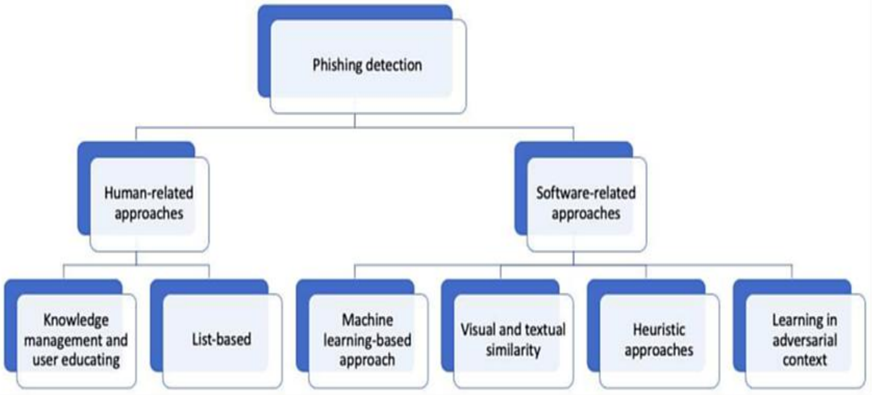
Phishing detection
Human-related approaches
Software-related approaches
Knowledge management and user educating
List-based
Machine learning-based approach
Visual and textual similarity
Heuristic approaches
Learning in adversarial context

Figure 1: Phishing detection Tree

2.1.2. Machine Learning Techniques for Phishing URL

In [7] three organization algorithms, namely DT, RF, and SVM, were utilized to classify a dataset benign URLs and phishing URLs. The dataset was composed from Alexa website and Phish Tank, and 16 features were assigned to each URL. Moreover, the authors concluded that improving the number of data used for training can lead to improved accuracy.

In [16] developed a dataset consisting of equal numbers of labeled phishing and legitimate URLs to address issues. The study used various classifiers to evaluate the lexical structure of URLs for phishing detection. The classifiers produced similar AUC values, but the Naive Bayes Classifier was deemed the most suitable, with the highest AUC value. This classifier achieved an accuracy of 98%, with a precision of 1, recall of 0.95, and an F1-score of 0.97.

In [19] Phishing attacks have become increasingly common, resulting in billions of dollars in losses due to users being tricked into providing sensitive information on fraudulent websites. These attacks are particularly successful when targeting Software-as-a-Service (SaaS) and webmail sites, as phishers can create websites. To combat this, researchers have developed phishing detection systems using machine learning algorithms such as Random Forest and Decision Tree. These models can analyze a range of features to determine whether a website is fraudulent or not, with high accuracy rates of up to 97%. Additionally, feature selection algorithms such as Principal Component Analysis (PCA) can be used to reduce the amount of irrelevant or redundant data, further improving the effectiveness of these systems. By detecting and preventing phishing attacks, these models can help protect users and organizations from the financial and reputational harm caused by these types of cybercrimes.

The act of constructing a website that appears to be identical to that of a legal business with the goal of stealing sensitive information is an example of phishing, which is a type of online crime that may be described as the common practice of creating such a website. There are a variety of distinguishing characteristics that set phishing websites apart from legitimate websites. These characteristics include the use of lengthy URLs, the incorporation of IP addresses into URLs, the addition of prefixes and suffixes to domain names and request URLs, and some other characteristics. In this study, rather than relying on an experienced individual to do the extraction procedure, they analyse essential components that are automatically collected from websites by utilising a novel technology. In order to determine whether or not a website is real, they first evaluate the characteristics and then assign a value to each of them. On the other hand, thanks to the social engineering component of phishing, it is more difficult to achieve the same results using phishing websites. It is possible that this fact is the primary rationale behind why phishing endeavours continue to increase in number. Boycotts were a feasible approach for monitoring fraudulent websites in the past, when the Internet was still in its infancy. However, those responsible for phishing attacks are ready to flood the internet with phishing attempts that are only intended to last for a brief period of time. It is also possible to use the findings of this category to provide assistance to end-users who fall under the first group. As can be seen in Figure 1, the existing body of research on phishing detection is comprised of two basic approaches: software-focused and human-focused.

Human-focused approaches aim to improve users' awareness and knowledge so that they can make informed decisions when confronted with phishing attempts. On the other hand, software-focused approaches strive to enhance the effectiveness of software-based solutions in identifying and preventing phishing attacks. The study [20] intends to build a set of elements that have been demonstrated to be sound and successful in predicting phishing websites and then to extract those features according to new scientific principles that are as accurate as possible. To evaluate the outcomes, the authors in [21] apply supervised learning techniques such as multilayer perceptron, DT induction, and NB classification. When compared to other learning algorithms, it has been discovered that the decision tree classifier makes more accurate predictions about the website that is being used for phishing. The research [22] described a stacking model that can identify phishing websites by analyzing URL and HTML characteristics of the page. In addition, they created a stacking model by mixing GBDT, XGBoost, and LightGBM in many layers. This allows for various models to be complimentary, which ultimately leads to an improvement in the performance of phishing site detection.

The research [23] suggests a way for recognising phishing by utilising multidimensional feature detection. This method is based on a rapid detection method that makes use of deep learning. The approach retrieves the character sequence features of the URL that is provided and uses them for rapid categorization in the first stage.

PhishZoo is the name of the proposed method, which was developed specifically for the purpose of detecting phishing [24]. When it comes to phishing, PhishZoo uses profiles that mimic the appearance of reputable websites. The proposed technique [24] obtains an accuracy of 96%, which is comparable to that of blacklisting methods. However, it has the additional feature of being able to differentiate between zero-day phishing attempts and focused attacks on smaller sites (such as corporate intranets). This is a significant advantage over blacklisting methods. On the other hand, thanks to the social engineering component of phishing, it is more difficult to achieve the same results using phishing websites. It is possible that this fact is the primary rationale behind why phishing endeavours continue to increase in number. Boycotts were a feasible approach for monitoring fraudulent websites in the past, when the Internet was still in its infancy. The perpetrators of phishing attacks, on the other hand, are ready to overwhelm the internet with phishing attempts that are only temporary. It is also possible to use the findings of this category to provide assistance to end-users who fall under the first group. A human-focused strategy and a software-focused approach are the two basic methodologies that are included in the existing research on phishing detection, as shown in Figure 1. It is the goal of human-focused techniques to raise the level of awareness and understanding among users, so that they are better equipped to make judgements when they are presented with phishing efforts.

On the other side, software-focused techniques work towards improving the efficiency of software-based solutions in detecting and stopping phishing assaults that are carried out. One of the most significant contributions that this paper makes is that it includes both a performance analysis and a technique for applying computer vision approaches in a practical situation. This is one of the most essential contributions that it makes. In order to identify between different types of phishing attacks, one such strategy that might be utilised is the application of ensemble techniques for machine learning. A number of different machine learning models were utilised in the development of the framework that was provided in the research [9] for the intelligent identification of phishing websites. A number of different classification algorithms were utilised by the framework in order to provide accurate phishing detection. Another study conducted by Somesha in the year 2020 offered novel phishing URL detection models that made use of deep neural networks (DNN), long short-term memory (LSTM), and convolutional neural networks (CNN) with only ten features from their earlier work. The proposed approach was successful in achieving high accuracy rates. The approaches that have been offered, which make use of a single property from a third-party service, are more resistant to the possibility of failure and have the potential to speed up the process of phishing identification.

According to the findings of the research study [25], the researcher proposed a novel heuristic method that makes use of TWSVM to recognize malicious recorded phishing sites as well as websites that are hosted on servers that have been compromised. This action is taken in an effort to overcome the limitations that were pointed out in the previous discussion. Identifying phishing websites that are hosted on hijacked domains is accomplished through the utilization of their method, which involves making a comparison between the log-in page and the homepage of the website that is being currently viewed. To identify phishing websites that have been fraudulently registered, features that are dependent on hyperlinks and URLs are utilized. This allows for quick and accurate identification. A number of different iterations of support vector machines, which are often referred to as SVMs, have been utilized by them in order to categories websites that are utilized for phishing.

To address phishing scams on the internet, researchers have developed innovative techniques to detect and prevent fraudulent activities. One such approach, described in [26], involves creating an anti-phishing method specifically designed for English-language phishing websites. Phishers often exploit brand names by incorporating them into various segments of a website's URL to deceive users. Leveraging this trend, the researchers assign different weights to terms extracted from HTML content, based on their frequency of occurrence within the hostname, path, and filenames of URLs. Through this method, they aim to identify phishing websites more effectively. Their findings indicate that the Random Forest (RF) classifier outperformed other classifiers in terms of accuracy, F-measure, and Area under the Curve (AUC). Not only was RF faster, but it also demonstrated greater robustness and accuracy compared to alternative classifiers [27].

In a related study detailed in [28], researchers explored the efficacy of utilizing website logos as a means of detecting phishing attempts. They developed a system capable of extracting and analyzing logos from legitimate websites, establishing a baseline for comparison with logos used in phishing websites. Their investigation revealed that the system achieved a commendable accuracy rate of 95.4% in detecting phishing attempts, with a low false-positive rate of 4.6%. Moreover, the researchers concluded that employing website logos for phishing detection proved more effective than relying solely on website content or URLs.

These research efforts underscore the importance of innovative approaches in the ongoing battle against phishing scams. By leveraging both textual and visual elements, such methods contribute to enhancing the accuracy and efficacy of phishing detection mechanisms, thereby bolstering cybersecurity measures to safeguard users' online experiences.

Overall, the study concluded that the utilization of website logos is a viable and effective approach to detecting phishing attempts. This approach can be integrated into existing anti-phishing systems to enhance their accuracy and reduce false-positive rates. In [29] describes a novel method to detecting phishing websites based on fuzzy logic. The researchers developed a system that utilizes fuzzy logic to analyze features such as website content, domain name, and URL length to identify and classify phishing websites. The study found that the proposed approach achieved an accuracy rate of 96.4%, outperforming traditional ML algorithms such as DT and SVM. The fuzzy logic-based approach also demonstrated a low false-positive rate, indicating that it is effective in distinguishing between legitimate websites and phishing websites. The researchers concluded that fuzzy logic is a promising technique for detecting phishing websites and can be used in combination with existing anti-phishing systems to improve their accuracy and performance.

In [7] proposed a machine learning-based approach for detecting phishing websites. The researchers developed a system that analyzes features such as URL length, domain age, and SSL certificate validity to classify websites as legitimate or phishing. The study found that the proposed approach achieved an accuracy rate of 97.7%, outperforming traditional anti-phishing methods such as blacklists and whitelists. The system also demonstrated a low false-positive rate, indicating its effectiveness in identifying phishing websites while minimizing the misclassification of legitimate websites. The researchers concluded that machine learning algorithms are a promising approach for detecting phishing websites, and their proposed method can be further improved by incorporating additional features and refining the classification model.

Figure 1 illustrates that the current literature on phishing detection includes two primary approaches: human-focused and software-focused. Human-focused approaches aim to improve users' awareness and knowledge so that they can make informed decisions when confronted with phishing attempts. On the other hand, software-focused approaches strive to enhance the effectiveness of software-based solutions in identifying and preventing phishing attacksThe proposed approach can also be used in conjunction with other anti-phishing techniques to enhance their accuracy and overall effectiveness in protecting against phishing attacks.

In [8] proposed a data mining-based approach for detecting phishing websites using associative classification. The researchers developed a system that extracts and analyzes features such as URL length, domain name, and web page content to identify and classify phishing websites. The study found that the proposed approach achieved an accuracy rate of 96.1%, outperforming other machine learning algorithms such as decision trees and naive Bayes. The proposed approach can also be integrated with existing anti-phishing systems to improve their accuracy and overall effectiveness in protecting against phishing attacks.

2.2. Deep Learning Techniques

The results of phishing websites have been investigated by a large number of scholars. In our method, we make use of important principles that have been discovered in the past. In order to determine how our current strategy is affected by past initiatives that used URL attributes to detect phishing, we looked at those initiatives.

Aljofey et al. [30] proposed a method for recognizing phishing based on the URL attributes of websites. They conducted a comparative analysis using various algorithms and deep learning approaches, including hierarchical structures, to assess the URLs of different data points. Their study encompassed three main approaches: the first focused on examining multiple URL attributes, the second on verifying the website's legitimacy by investigating its source and ownership, and the third on analyzing the visual appearance of the website. Deep learning techniques were employed to explore various aspects of web pages and URLs.

In a related vein, Yao et al. [32] introduced an innovative approach to identifying phishing websites, concentrating specifically on URL analysis. This method, touted as both accurate and efficient, serves as a potent detection tool. To elucidate their methodology, they partitioned their novel neural network (NN) structure into multiple concurrent components.

One notable strategy they employed involved eliminating superficial URL characteristics, aiming to streamline the detection process and enhance efficiency.

These studies underscore the multifaceted nature of phishing detection, wherein researchers explore diverse avenues, including URL attributes, website legitimacy, and visual characteristics. By leveraging advanced techniques from deep learning and innovative neural network structures, researchers aim to develop more robust and efficient methods for combating phishing threats. Through such endeavors, they strive to bolster cybersecurity measures and safeguard users against online fraudulent activities.

In studies referenced by [32,33], researchers adopt a multifaceted approach to phishing detection by combining deep features extracted from URLs with simpler features to assess the legitimacy of web addresses. By integrating insights from both sophisticated deep learning techniques and more straightforward evaluation methods, they aim to achieve a comprehensive understanding of URL characteristics and enhance the overall effectiveness of their detection system. Through their investigations, these researchers demonstrate the capability of their system to maintain competitive performance with existing detection algorithms while ensuring a balanced allocation of resources towards identifying phishing websites. By leveraging deep features alongside simpler attributes, they strike a balance between accuracy and efficiency, allowing their system to achieve optimal throughput without compromising on detection capabilities. Their exhaustive analysis of internet data sets showcases the robustness and adaptability of their approach in keeping pace with evolving phishing threats. By continually refining and optimizing their detection system, these researchers contribute to the ongoing efforts to combat online fraud and enhance cybersecurity measures. Through such endeavors, they strive to provide users with reliable protection against phishing scams and other malicious activities on the internet.

A method for recognizing phishing webpages was proposed by Korkmaz et al. [34], and it was based on Hypertext Markup Language (HTML), Tag Distribution Language (HTDL), and URL properties. In addition to this, they developed specific HTDL and URL characteristics that enabled them to develop HTDL string-embedding functions without having to rely on infrastructure provided by a third party. This made it possible for one method to be implemented in a genuine recognition application. The researchers put their method to the test by applying it to a genuine database that had more than 30,000 HTDL and URL attributes. According to the authors, their arrangement achieved an accuracy rate of 98.24%, a True Positive (TP) rate of 3.99%, and a False Negative (FN) rate of 1.74% [34]. Stokes et al. [35] proposed a novel method for intelligent probability detection that is based on the text features of websites in order to identify zero-day phishing tactics. The individuals conducting the research made use of the concepts of consistent resource identification and sequencing tactics that typically correspond to their framework. With a total probability (TP) rate of 95.38%, the researchers claim that the technique that was recommended is capable to successfully detecting phishing and zero-day attacks. Previous research has utilized the text structures of websites in order to develop PD frameworks. Phishers, on the other hand, were able to circumvent detection by using content pulled from other websites.

Yerima et al. [36] investigated the effectiveness of the long short-term memory (LSTM) classifier as part of their investigation into a technique for faking site forecasts that makes use of hyperlinks as a data source for deep learning models. Comparing an RF classifier-based method with a new RNN-based method is the focus of the researcher's investigation. These fourteen criteria were utilised in the scientific investigation that they carried out on web addresses. To begin, the researchers developed a model that makes use of LSTM to accept a hyperlink as a sequence of text as input and determines if the URL is genuine or false. User research has shown that the LSTM algorithm is superior to the RF classifier in terms of the average accuracy rate. This is the case despite the fact that the creation of the features does not require the expertise of a professional. An accuracy of 97.4% is achieved by their method, despite the fact that it does not involve preprocessing or feature generation [36]. As an additional point of interest, their investigation was limited to the text-feature perspective of webpages. It is possible that the effectiveness of their model could be improved by incorporating other components, such as frame characteristics and website graphics.

The researchers Selamat et al. [37] used logistic regression with CNN and CNN-LSTM to analyze two different URL data sets for the purpose of phishing detection research. They compiled a data set after collecting information from a variety of sources, including malware domains, malware domain listings, and phishing domains from OpenPhish and Phish-Tank. There are approximately seventy thousand URLs for training purposes in the database, and there are over sixty thousand URLs for testing purposes. Additionally, the CNN-LSTM and CNN phishing URL detection models were trained with the help of the data set. The local support vector machine (LSTM) approach was used since it takes into account the actual data web address as input. During the course of their experiment, the CNN-LSTM architecture demonstrated superior performance to the other framework, achieving an accuracy rate of approximately 97% by categorizing URLs [38]. However, the approach that has been described only makes use of text-based features, and it is possible that it might be improved by including other attributes and optimizing the variables in order to achieve a greater level of accuracy. As a consequence of this, the limitations that were discovered in earlier studies served as the basis for the IPDS that we presented.

For the purpose of detecting phishing, Janet et al. [39] utilized DL methods. RF, Support Vector Machine (SVM), and Artificial Neural Network (ANN) were the categorization approaches that were utilized. They came to the conclusion that the RF algorithm functioned relatively well. Rishi Kotak [40] was able to identify phishing attacks by utilizing a variety of DL techniques. In conclusion, Rabab et al. [41] conducted a series of tests on different deep learning models for the purpose of phishing detection and found that random forests performed the best.

As a consequence of the literature study on Phishing Detection in Modern Security utilizing URLs, research gaps have been found so that further investigation can be conducted. There is a lack of standardization in the process of recognizing phishing URLs, and there is also an inadequate investigation of real-world data and user interactions with phishing URLs. These research gaps include both of these things. As an additional point of interest, there is a tendency to concentrate on known phishing attacks while ignoring emerging threats.

All of these issues are solved by the model that we have developed, which employs a CNN-based approach for accurate classification, which helps to contribute to standardisation. In that way, the model helps to contribute to standardisation. During the process of evaluating the model, which takes place on both synthetic and genuine datasets, a total of 10,000 phishing URLs and 10,000 authentic websites are taking into consideration one another. Within the framework of the model, an investigation of user behaviour is implemented in order to enhance its effectiveness. As a result of this, the necessity of understanding the decision-making capacities of users is acknowledged and taken into consideration.

2.3. Comparative Analysis of Deep Learning and Traditional Algorithms for Data Processing Tasks

Real-time phishing URL detection using a deep learning algorithm has several benefits. One of its key features is real-time phishing detection and blocking. Phishing attacks are mitigated by this feature. Additionally, the algorithm may be trained and updated often, making it versatile. Its versatility allows it to stay relevant as phishing methods change, ensuring its capacity to identify new threats [36].

The algorithm's 97.87% accuracy in identifying phishing HTTP addresses is one of its most essential features. Accuracy is essential to avoid false negatives and maintain system confidence. The algorithm also reduces false positives better than previous methods due to its higher precision. This characteristic is crucial for preventing unwanted disruptions and invalid website blocking.

However, the algorithm has issues. It depends on large datasets with a diversity of information, making it sensitive to training data quality and representativeness. If the data does not cover many phishing options, the algorithm may perform poorly. Another consideration is that deep learning model training and maintenance need many resources. Organizations with minimal computing abilities may have trouble implementing and maintaining the method.

Deep learning models are often called "black boxes," making interpretability challenging. To establish confidence and assure accountability, you must understand how the algorithm draws decisions. Even while precision is stressed, erroneous positives can cause users to ban valid websites, causing problems. In conclusion, the algorithm needs constant monitoring and changes to stay effective. This requires ongoing resources and significant learning experience.

2.4. Summary

Indeed, the effectiveness of any phishing detection model can be influenced by the dynamic nature of phishing methods and the constraints of the dataset used for training and evaluation. These limitations underscore the importance of continuous research and development efforts aimed at enhancing the adaptability and robustness of detection systems. One significant challenge lies in the ever-changing landscape of phishing techniques employed by cybercriminals. Phishers constantly evolve their tactics to bypass detection mechanisms, making it crucial for detection models to remain agile and responsive to emerging threats. By conducting additional research, researchers can stay abreast of the latest phishing trends and refine their models accordingly. This may involve exploring novel features, incorporating real-time data sources, or leveraging advanced machine learning techniques to detect subtle variations in phishing attempts.

Furthermore, the composition of training datasets plays a pivotal role in the performance of phishing detection models. Diversifying training datasets to encompass a broader range of phishing scenarios and variations can help improve the model's ability to generalize to new and unseen threats. This may involve collecting data from diverse sources, including different industries, geographical regions, and languages, to ensure comprehensive coverage of potential phishing attacks. Identifying deficiencies in existing models is essential for driving innovation in phishing detection. By recognizing the limitations of current approaches, researchers can focus their efforts on developing new methods that address these challenges more effectively. This ongoing pursuit of innovation is critical in the fight against phishing, as cyber threats continue to evolve and adapt.

In summary, the recognition of limitations in phishing detection models underscores the need for continual research and development efforts. By enhancing adaptability to emerging threats and diversifying training datasets, researchers can advance the state-of-the-art in phishing detection and better protect users against online fraud.

Table 1: Comparison with Previous Work

Ref	Title and Year	AI/ML Algorithm	Dataset	Strength	Limitation	Results
[41]	K-nearest neighbor-based URL identification model for phishing attack detection. (2021)	K-nearest Neighbor (KNN)	Phishing emails	Simple to implement; Effective for small datasets	Prone to overfitting; May not perform well with large datasets	85% Accuracy
[39]	Phishing Attacks Detection: A Machine Learning-Based Approach. (2022)	Machine Learning (RBF- SVM)	Phishing URLs	Effective for detecting phishing URLs; Can handle large datasets	Long training times for large datasets; Resource-intensive	88% Accuracy
[42]	Phishing website detection based on deep convolutional neural network and random forest ensemble learning. (2022)	Deep Convolutional Neural Network (CNN), Random Forest	Malicious and Legitimate Viruses	High detection accuracy; Combines strengths of CNN and Random Forest	May not capture complex relationships in data; Resource-intensive	92% Accuracy
[40]	Intelligent phishing detection scheme using deep learning algorithms. (2023)	CNN based Deep Learning Model	Phishing websites	High accuracy; Can automatically extract features	Prone to overfitting if not properly regularized; Requires substantial computational resources	87% Accuracy
[43]	Adaptive phishing detection using transformer-based neural networks. (2023)	TNN	Phishing emails	High precision; Effective in capturing long-range dependencies	High computational cost; Limited by dataset diversity	94% Accuracy

[45]	Detection of phishing websites using hybrid deep learning models, (2022)	Hybrid Deep Learning (CNN + LSTM)	Phishing websites	Captures both spatial and temporal patterns; High detection rates	Requires large labeled datasets; Computationally expensive	96% Accuracy
[48]	Phishing detection with graph-based machine learning. (2023)	Graph Neural Network (GNN)	Phishing and Legitimate URLs	Captures complex relationships in data; Scalable to large datasets	Requires feature-rich datasets; Computationally intensive	95% Accuracy

Table provides an overall comparison of various phishing detection studies where usage of AI and machine learning algorithms, dataset used in each works to have a high-level pros (advantages) and cons(drawbacks). An overview of what was found in the first study from 2021 where phishing emails were treated using K-nearest Neighbor(KNN) to detect the Phishing Attacks. KNN, because of its simplicity it classifies the data on basis majority votes from the K-nearest neighbours [41]. Overfitting is when the model has too many tunable parameters and consequently fits noise in data, not the signal. Additionally, the KNN algorithm is computationally expensive and slow with a large dataset as distance calculation for all training data points involves every query point such that predicted response time of any new record against other records in feature space (if assigned a class) to store together higher frequency items. This limitation makes KNN less appropriate for larger-scale phishing detection tasks where time to completion, as well scalability are important.

In [39] author used machine learning (RBF- SVM) algorithm for phishing URL detection. Classification Tasks: Machine learning models are great at classification tasks and identifying patterns within data that separates phishing URLs from legitimate ones. This highlights the utility of these models for big data analysis as required in phishing detection considering they have a wide-ranging or large and structured/phrased equivalent. But the study also underscores how long it takes to train these models, especially if they are working with large datasets [39]. This high demand requirement requires a large amount of computational power combined with memory, making it difficult for organizations that do not have access to such resources. Nonetheless, machine-learning algorithms are a powerful way of identifying phishing and can work in most environments with large datasets.

Earlier that year, another study proposed a hybrid model based on Deep Convolutional Neural Networks (CNN) and Random Forest algorithm to identify phishing websites [30]. This is an ensemble method that utilizes CNNs for their automatic feature extraction ability and Random Forest as the backbone to perform classification. With a hybrid model, the level of detection accuracy was high and this contributed to phishing detecting suspicion. But the approach, according to the study, might not take account of more complex data relationships that exist and hence it may be hindered in situations where phishing is less straightforward. Moreover the computational requirements of CNNs make this a resource-heavy strategy, which may limit its utility to those organizations with large computing resources at hand.

In [40] the author using deep learning algorithms for phishing detection (phishing webs) Deep learning models, especially the one involving architectures like CNNs and Long Short-Term Memory (LSTM) networks achieved state-of-the-art accuracy in classification tasks. Deep Learning seems to be good at this since it automatically learns more useful features from raw data than proper feature engineering efforts the value of which decrease with time [40]. It seems highly beneficial in particular for phishing detection when the techniques can be wide-ranging and sophisticated. Yet, deep learning models tend to overfit a lot more than any other kind of model; specially if not properly regularized. A well performing but slightly overfitting model excels on the training data, is not so good at generalization to unseen testing. Furthermore, deep learning models need heavy hardware cards such as high-performance GPUs and large amounts of memory (which is above 4GB) which can be a bottleneck some organization.

CHAPTER 3

3 PROPOSED METHODOLOGY

3.1. Network Overview

3.1.1. Network Overview

Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) layers are the components that make up the neural network architecture that has been defined. This architecture was developed for sequential data processing and is particularly useful for applications such as keylogger detection. The model begins with a number of convolutional blocks, each of which incorporates a Conv1D layer, ReLU activation, Dropout, Batch Normalization, and MaxPooling for the purpose of hierarchical feature extraction. The next layer, an LSTM layer, is responsible for capturing the temporal dependencies that are present in the data. After that, the output that has been flattened is fed into two thick layers that also have ReLU activation and dropout in order to learn complex patterns. The last layer, which is a dense layer fitted with a sigmoid activation function, makes binary classification easier to accomplish. Because this architecture is designed to take advantage of both sequential and spatial information contained within the input data, it is versatile enough to be used for jobs that involve sequential data processing.

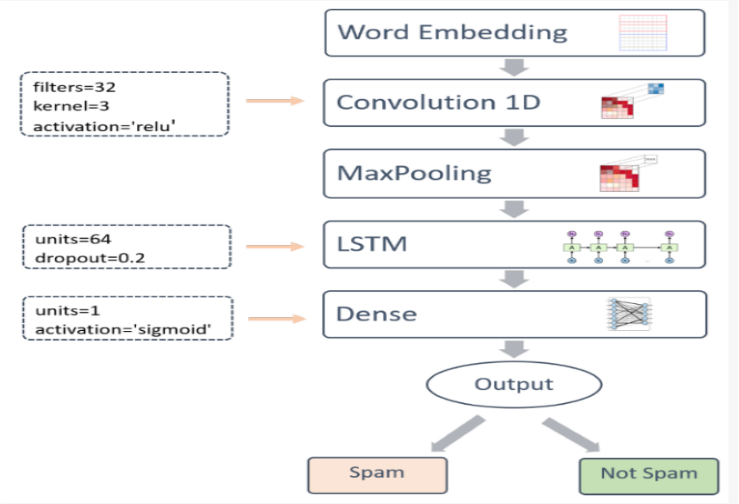

Figure 2: CNN-LSTM MODEL Architecture

3.2. Layers Used in Proposed Model

3.2.1. Convolutional Layer

Convolution is an operation that is used in the context of Convolutional Neural Networks (CNNs). This operation involves applying a filter, which is sometimes referred to as a kernel, to an input image in order to generate a feature map or feature map. In order to compute the element-wise multiplication between the filter weights and the associated pixel values of the input image, the filter is dragged over the image that is being input. After that, the output feature map is generated by adding up all of the data that were obtained. For our model to detect phishing attacks, we experimented with the approach of using LSTMs & CNN. Our CNN model had a convolutional layer which checked our input data to get the patterns in it that suggests targeted phishing. In these layers, various features such as tabs from URLs and keywords found in the webpage were detected respectively with a higher efficiency than seeing all sites under surveillance. Our method reached good precision for the phishing attacks due to temporal capability of LSTM and spatial feature extraction capacity of CNN. The combined model of these two architectures can take advantage the merits from each to deliver a phishing detection system with robustness and efficiency [43][44].

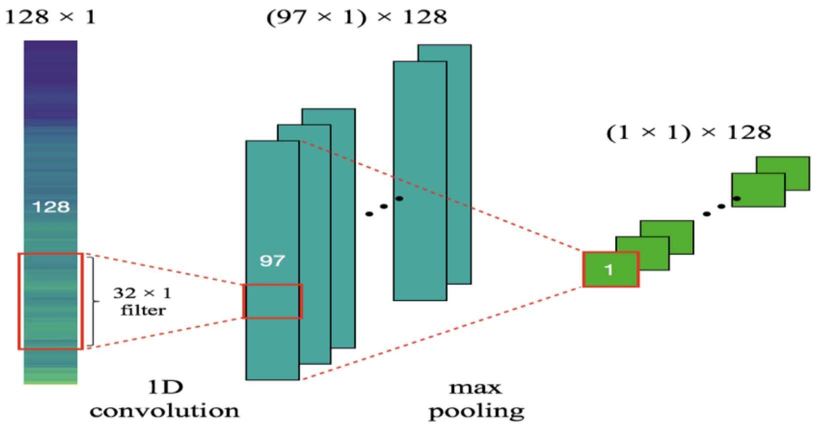

Figure 3: Convolutional Operation

3.2.2. Pooling Layer

Pooling is a down sampling procedure that is often employed in CNNs to lower the spatial dimensions of feature maps. More precisely, max pooling and average pooling are the two types of pooling described here. When using max pooling, the maximum value inside a window of a given size is chosen and the other values are discarded. On the other hand, when using average pooling, the average value within the window is computed. Through the use of pooling, computational complexity may be reduced, dominating features can be extracted, and translational invariance can be introduced successfully. We used Long Short-term Memory (LSTM) Networks and Convolutional Neural Network (CNN) models in our phishing-attack detection. One of the key aspects in our CNN model was a pooling layer where it decreases dimensionality and retains only those features which may be essential. This layer rightly down-sampled the input representation, thus reducing computational load and preventing overfitting. Our approach was able to achieve a high level of accuracy by combining the temporal capabilities offered by LSTM with spatial feature extraction and dimensionality reduction provided through CNN for detection of phishing attacks. Using a hybrid model that utilized both these architectures, we could play to their strong suits and come up with phishing detection system which was robust and efficient12.

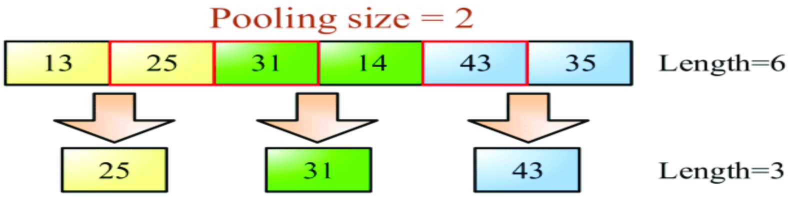

Figure 4: *Max and Average Pooling Operation*

3.2.3. Batch Normalization

One method that is utilized in the process of normalizing the activations of each layer in a neural network is known as batch normalization. The input of each layer is normalized, and this is commonly accomplished by subtracting the batch mean and dividing by the batch standard deviation. This is how it comes into operation. This contributes to the stabilization and acceleration of the training process by minimizing the change of the internal covariate, which in turn enables better learning rates and improved generalization. For detecting phishing attacks, we applied both Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN) in our methodology. The batch normalization layer in our CNN model facilitated to stabilize and accelerate the training process by normalizing the output of prior activation layer. This layer made an adjustment to the activations by subtracting out batch mean and divinding through with the standard deviation (or normalizing) before applying scaling and shifting with learnable parameters. With that in place, this normalization was able to remove the internal covariate shift which translated into getting better results faster. Combining temporal capabilities of LSTM and stabilized/accelerated training of CNN through batch normalization, the attack detection accuracy is near 100% in our approach. Our phishing detection system took advantage of both architectures by effectively combining them together, thus the use of hybrid model yielded strong and light weight multitude13.

3.2.4. Activation Function

One of the activation functions that is frequently utilized in neural networks, including CNNs, is known as the Rectified Linear Unit (ReLU). The non-linearity is introduced by the fact that it returns the value of the input if it is positive and by returning zero otherwise. In terms of mathematics, ReLU(x) equals max(0, x). The vanishing gradient problem can be solved with the assistance of ReLU activation, which also provides a means of accelerating convergence during training and encouraging sparse activations.

For warning about the phishing attacks, we considered Long Short-term Memory (LSTM) networks and Convolution Neural Networks(CNN). Many people will ask this question, do you really need the activation function in CNN model? It is playing a very important role to introduce non-linearity into your network. That enables networks that learn and represent nonlinear complex patterns using inputs of varying dimensions straight away from layer.dup line 12 instead for input_shape. Activation functions (like ReLU: Rectified Linear Unit) after each convolutional layer output might assist the model in detecting even more intricate and important feature patterns for identifying suspicious phishing attacks.

This non-linear transformation allowed us to more accurately detect between legitimate and phishing emails. Our method is very accurate in phishing detection by combining LSTM's time-line analysis with the CNN activation function which can help for improved feature extraction. The strength of both architectures combined in a hybrid model had created an effective and highly efficient solution to detect phishing attacks.

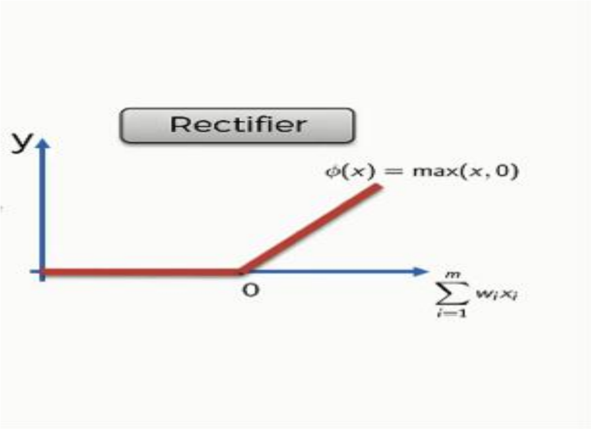

Figure 5: Rectified linear unit

3.2.5. Fully Connected Layer

An example of a type of layer in a neural network is a Fully Connected Layer, which is sometimes referred to as a Dense Layer. In this layer, every neuron in the layer below it is connected to every neuron in the layer above it. Deep convolutional neural networks (CNNs) often have fully connected layers that are situated towards the end of the network. These levels are accountable for learning global patterns in the feature maps that are retrieved by the convolutional and pooling layers. Through the process of transforming the input features into a final output, they carry out classification or regression analysis tasks. We used both Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks(CNNs) in order to detect the phishing attacks. The FC layer not only meaninglessly reduces dimensions, but also destroys spatial hierarchies across the network in our CNN model. In contrast to classic layers that work on flattening the input, fully convolutional layer respected with spatial dimension and this way it enabled so much robust capture of complex patterns within data. This method allowed the network to accept input data of different sizes and it could still predict on test samples at large. As a result, the LSTM model can understand temporal features within time series data and it achieved high accuracy of phishing detection by combining with space feature learning ability via fully convolutional layers in CNN. This work used a hybrid model combining the strengths of each architecture to create an effective and efficient system targeted at phishing detection12.

Eq – 1. $$output = f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

3.3. Implementation in URL Detection

In proposed study, we are developing a novel approach to detect malicious URLs and alert users is critical for preventing phishing attacks and protecting sensitive data. Machine learning techniques have shown promise in detecting phishing URLs in real-time, making them an effective tool for addressing this problem. But by applying deep learning techniques, we can analyze real-time URLs and produce effective results that can be used to alert users of potential phishing attacks. This deep learning algorithm can manage huge data and have been shown to provide high accuracy rates.

We can develop a comprehensive approach to detecting malicious URLs in real time. This approach can be integrated into web browsers, email clients, and other software applications to provide real-time protection against phishing attacks. When a user clicks on a URL, the software can analyze the URL using these machine-learning techniques and determine if it is a phishing attempt or not. If the URL is identified as a phishing attempt, the user can be alerted, and the URL can be blocked. Overall, applying this technique to real-time URL analysis can provide a powerful tool for detecting and preventing phishing attacks.

Following are the simplified steps that our proposed system will follow:

- **Data Collection and Pre-processing:** The process of building a phishing URL detection system can be broken down into several parts. Generating Dataset with Phishing and Benign URLs It should be large enough to try and account for all the methods of phishing that exist. Collection may involve web scraping, publicly available datasets or unique data from your organization. The dataset is created by collecting raw data which then goes through the pre-processing stage where duplicates and redundant entries are eliminated to clean up the quality of your datasets. These exclusion criteria work in order to create a more representative dataset, free from bias. For example if two URLs are way too similar or don't meet certain predefined requirements we exclude it as well for not being useful on our model's learning of new data points.

- **Feature Extraction:** This is a most important phase in which the certain features are captured from URLs and web pages for identification of phishing vs legitimate. Some features would include domain name, length of URL, number of subdomains, presence/absence of suspicious keywords (such as login or secure) and whether the URL contains special characters or anomalies. Moreover, the webpage text is tokenized to obtain features from it using pre-trained language models (such as BERT—Bidirectional Encoder Representations from Transformers). To recognize this type of phishing run, BERT is capable to analyze the content text on the website and understand context & semantics.
- **Data Pre-processing:** After extracting features, it must also be normalized so that the data can have a similar scale. This is critical as values in the URLs could have drastically different ranges e.g., length of URL and whether a keyword exists, so normalizing these features ensure that entire feature space carries uniform distribution than any one feature dominate learner. In the train-tes-split after normalization, then they will split data into training set, validation set and testing. You see, a training set is used to condition the model while validation and testing sets are for hyperparameter tuning/model evaluation at development time and found out about this at so-called test/calibration phase when we deploy our pre-trained/construction models against raw data.
- **Choosing and Training a Deep Learning Model:** The choice of the deep learning model will also affect the success of the system. Common models used for phishing

detection include Convolutional Neural Networks and Long Short-Term Memory networks. Since both CNNs and LSTMs have unique strengths, CNNs can learn from structured data, such as URL features, whereas LSTMs learn from sequence data, such as the sequence of characters in a URL. The next step is to train the model on the training set. The model learns to classify phishing URLs by adjusting its parameters using backpropagation and an optimization algorithm such as stochastic gradient descent.

- **Evaluating the Model:** Next, evaluate the performance using a separate validation set to compute various performance metrics, including accuracy, precision, recall, and F1 score. Accuracy refers to the overall correctness of the model, whereas precision refers to the proportion of detected phishing URLs that were true. On the other hand, recall refers to the proportion of actual phishing URLs detected, and the F1 score balances the two. This step gives the researcher an insight into the strengths and limitations of the model and identifies areas for further improvement.
- **Tuning the Model:** The model is then fine-tuned according to the evaluation results i.e. hyper-parameters e.g.learning rate, number of layers,units etc.Middle layer(s) are added and removed or changed by tweaking with parameters. This step can be trial and error, as you will have to try few different combinations of the list above many times until your model is well trained. The aim is to improve the generalization from training data of models to unseen data, decrease overfitting and enhance detection accuracy.
- **Testing the Model:** After the model is fine-tuned, it is tested on the testing set that contains data which has never been seen by the model. This gives the model and you a realistic estimate of what can be expected in real life performance. The performance of the model can then be evaluated using the testing set, as they give a representation how well our algorithm will identify phishing URLs unseen before.
- **Deployment:** Once the model passes all tests, it is then deployed into a production environment where every incoming URL can be fed to this model in order to receive one of two possible labels Phishing or Legitimate. For deployment, it could

be integrating the model into an existing security infrastructure or creating a completely separate application that watches web traffic. Capacity: In order to protect users from phishing attacks, the system must be highly reliable with low latency and real-time URL processing.

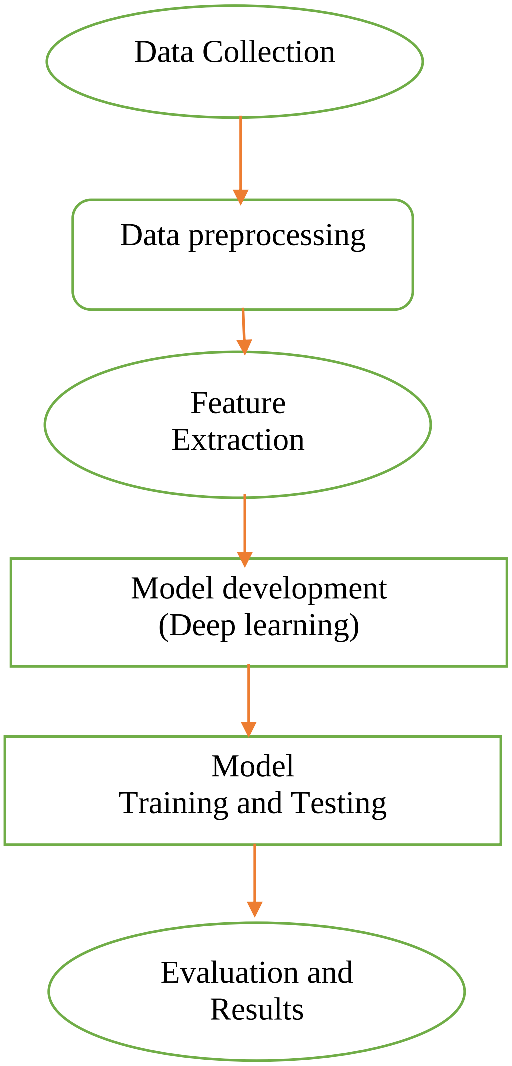

Figure 6: Flow Chart of the System Proposed

3.4. Phishing Dataset

This dataset is designed for phishing URL detection and features a comprehensive set of characteristics aimed at enhancing the model's ability to accurately distinguish between phishing and legitimate URLs. It includes a broad range of features extracted from 87 media types, ensuring diverse and extensive coverage of URL and webpage elements. These features include both structural components (such as domain name, URL length, and subdomains) and content-based features derived from the webpage itself, including those analyzed by models like BERT. The dataset reflects real-world conditions, simulating a variety of phishing tactics, from simple to highly sophisticated attacks, thereby providing a realistic testing environment. This complexity helps the detection system handle the evolving nature of phishing threats.

The dataset is also carefully balanced to include an equal number of phishing and legitimate URLs, minimizing bias and ensuring that the model is trained to optimally identify both classes. This balance is crucial for accurate training and fair model evaluation. Additionally, the dataset is structured to offer detailed insights into phishing behaviors, providing a solid foundation for evaluating model performance under practical, real-world conditions. This makes it an excellent resource for assessing the precision and reliability of phishing detection models in diverse scenarios.

- **Features and Records**: This system uses a far-reaching set of features extracted using 87 media types; for example, phishing dataset does not cut any corners. Specific Characteristics of the features captured that are related to URLs and Web Pages. This variety is required to ensure that the model can reasonably separate phishing from not-phishing URLs in a range of different situations and attacker tactics.
- **Realism and Complexity**: It is a dataset that resembles real-world conditions, reflecting the complexity and sophistication of modern phishing attacks. This includes everything from basic, easily identifiable phishing to sophisticated and stealthier methods. This realism is crucial for the test environment in order to be as close to what it will actually face during deployment; this means that our system must account for evolving nature of phishing threats.
- **Diversity of Features**: It contains a whole range of features including structural elements like domain name, length and subdomains as well as content-based

features derived from the webpage itself (with models like BERT). Such variety helps the detection model be secure against alternating phishing modes, ranging from obfuscation to semantic-based deceptions.

- **Balanced Composition**: In order for the dataset to be evaluated fairly and free of bias, it minimizes its unbalanced nature by including an equal number of phishing URLs as well as legitimate URLs. This balance is very important to properly train the model so that it does not have bias toward any of class and can identify both classes (phishing links as well as legit urls) optimally.
- **Evaluation Capability**: The dataset is so detailed and balanced that we should be able to make a very good estimate of the model performance in real situations. The dataset covers all aspects of phishing and trusted URL features to achieve an efficient precision rate, which means that the achieved threshold is neither bad nor good, it tries to be practical by challenging a version in real world applications.

3.5. Compare According to Dataset

Table 2: Compare according to Dataset

Dataset	Algorithm Name	Virus	Accuracy
Phishing [39]	RBF-Support Vector Machine [39]	Phishing URL	73%
Phishing [42]	Hybrid Machine Learning Algorithms {KNN, SVM}[42]	Malicious	{93%, 91%}
Phishing [41]	KNN [41]	Spam	85.08%
Phishing [40]	CNN based Deep Learning Model [40]	Legitimate	93.4%

A comparison of algorithms applied for phishing detection with their accuracy on different datasets has been discussed in above table 2. The second entry uses a machine learning approach for phishing URL detection based on Radial Basis Function (RBF) variant of SVM which resulted in 73.0% accuracy [18]. It shows us that SVM is a good option but it may vary from the complexity of data. In a hybrid model called with K-Nearest Neighbors (KNN) combined with SVM was also proposed, as resulted in accuracy rates of 93% using the first algorithm and 91% applying it to detect malicious URLs. This combination utilises the strengths of each algorithm and exemplifies how hybrid methods are advantageous for phishing detection. KNN on its own performs well in spam detection with 85.08% accuracy depicting a powerful nature of the classifier but dependent upon dataset characteristics for effectiveness; Finally, the identification of malicious/phishing URLs by using a CNN-based deep learning model delivers surprisingly encouraging results with an accuracy greater than 93.4%. This proves the increased efficiency of deep learning methods in identifying phishing, since CNNs are able to automatically learn useful features and recognize complex patterns in data. The comparison highlights that while traditional machine learning methods such as SVM and KNN are also helpful, the better performance of CNNs in this scenario suggests a growing need for deep learning models to be used more widely by various domains in order to improve cybersecurity defences against phishing attacks.

3.6. Accuracy Matrix

3.6.1. Intersection over Union

How do we evaluate the object detection techniques IoU? Calculates overlap between expected and ground truth bounding boxes of the object. Intersection over Union (IoU) is the ratio of intersection to union area between predicted and ground truth bounding boxes. As you can see, the higher IoU conditions improve localization accuracy.

IOU = (Area of Intersection) / (Area of Union)

3.6.2. Precision and Mean Average Precision

Precision measures the accuracy of a classifier in positive predictions Real positive predictions are all real positives identified as suchAll correct estimates of true features and some other properties The precision is from 0 to 1, the higher value indicates less false positive. In medical diagnosis and fraud detection, false positives are expensive or problematic so precision is paramount.

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

3.6.3. Recall

Recall is the number of positive samples out to be correctly predicted as positive, which means identifying all positives from a dataset. This is the number of true positive forecasts divided by sum of both false negatives and true positives. Higher values of recall are much more preferable because it means that a lot more positive instances have been detected.

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

3.6.4. F1 Score

Following this, the overall performance of classifier is balanced due to harmonic mean of precision and recall i.e. F1 Score. The ratio of 2 * precision, recall / (precision + recall) is calculated. The better the F1 Score, higher will be our model performance. F1 Score even calculates false positive and false negative analysis, So it works well when you have a asymmetric datasets.

$$\boldsymbol{F1 = 2.\frac{precision\,.recall}{precision + recall}}$$

3.7. Pre-Requisite

3.7.1. Hardware Requirements

Table 3: Hardware Requirements

Processor	I5 8th Generation Processor
RAM	8GB
Graphics Card	2GB

CHAPTER 4

4. RESULTS AND DISCUSSION

4.1 Overview

This chapter presents the evaluation of simulation and results that are demonstrated by simulation and model and their discussion. Through demonstration, the validation is achieved with proper configuration. The prevalence of phishing attacks in e-commerce poses a significant threat to web users. To address this issue, a system has been developed that focuses on differentiating between legitimate and illegitimate URLs based on their features. These features are extracted from a training dataset and used to train the system, enabling it to categorize URLs based on their unique characteristics.

4.2. Libraries Used

Our simulation is build on Python programming language running in google collab. We have implemented using the following Python Libraries.

4.2.1. Pandas:

This is another python library that in the background depends on over NumPy and liable for planning high level records like information outline to AI/ML arrangement or preparation. It relies on two types of data structures one is a series (single dimensional) and other is DataFrame(a 2-dimensional structure). This allows Pandas be applicable in various industries consisting of finance, engineering and data. Unlike the lazy beasts themselves, Pandas library is fast, reliable and versatile.

4.2.2. Numpy:

It is a renowned Python library for multi-dimensional array and matrix processing since accomplishing an excellent variety of numerical operations can be employed with this. Its support of direct polynomial math, fourier changes and that's just the beginning out-of-the-container make it great for AI (artificial intelligence) applications in addition to other things by permitting you to control your matrix so as to effectively further develop your AI execution. Numpy is faster and easy to work over other python libraries.

4.2.3. Seaborn:

Another open-source Python library that functions with Matplotlib (which focuses more on plotting and data visualization) but provides a structure for the data of Panda. Since, Seaborn is used many a times in ML projects where we deal with plots of learning data. It makes the most aesthetico-beautiful graphs and plots of almost any Python library, which historically made it a great choice if you intended to also use your plot for marketing and data analysis.

4.2.4. Matplotlib.pyplot:

Pyplot is an Application Programming Interface (API) of python's matplotlib, it makes easier for users to navigate through the large amount methods in pyplot function call by providing simplified interface.realpath. This powerful tool can help you explore your data sets into detailed reports/graphs with few lines of code.pyPlot helps make Matplotlib a still viable public copy program alternative compared to MATLAB. It is a library for information perception, utilized both in Python and IPython.

4.3. Proposed Model Architecture

The model is structured as a sequential neural network consisting of three dense layers with dropout and batch normalization to enhance generalization and improve convergence. Below is a detailed explanation of each component:

This is neural network architecture that has input layer, hidden layers and output layer which are pre-defined for multi-class classifcation task. The input layer with 31 features, (None, 31), and None means the batch size is not yet known.

The first hidden layer dense (64 units, ReLU activation function, He Normal weight initializer) After this layer, batch normalization is done which sets the mean of that input in such a way to keep it constant before being fed into next video stream so as to make training more predictable. And by adding a dropout layer (with the rate of 0.5), this helps to prevent overfitting, randomly setting input unit values at a rate for reducing bias in our deep learning model —batch normalization is another example which can also do this as well and amongst many others.

The second hidden layer is just another dense 32 unit ReLU from He Normal. The second hidden layer is processed similarly to the first one by applying batch normalization after the dense layer and having a dropout of 0.5 also in between it as seen below :

The last layer is a dense layer which has the same number of units as the number of classes in our dataset (i.e whose length ranges from 0 to num_classes). Output Layer: The activation function for the output layer is softmax which makes it appropriate to be used in multi-class classification tasks where classes are mutually exclusive.

4.3.1. Pseudocode for Model Architecture

To facilitate a clear understanding of the proposed model's structure and functionality, we present the pseudocode for the model architecture. The pseudocode provides a high-level description of the steps involved in creating, compiling, and training the Tensor Flow-based deep learning model.

By abstracting the detailed code, the pseudocode highlights the key components and logical flow, making it accessible and comprehensible for readers with varying levels of technical expertise. The pseudocode includes the initialization of the model, addition of layers with specific configurations, and the compilation and training processes, encapsulating the core operations required to implement the multi-class classification model.

4.4. Representation of Graphs

4.4.1. Training and Validation Accuracy

In deep learning, training accuracy measures how well a neural network has learned to correctly classify or predict the training data, indicating the model's ability to capture the underlying patterns in the data. Validation accuracy, on the other hand, assesses the model's performance on a separate dataset not used during training, providing insight into its generalization ability. Monitoring these metrics throughout the training process is crucial for identifying overfitting, where the model performs well on the training data but poorly on new data, and for making informed decisions about hyperparameter tuning and model adjustments to improve overall performance.

```
Initialize Model:
   model = Sequential()

Add First Dense Layer:
   model.add(Dense(units=64, activation='relu', kernel_initializer='he_normal',
input_dim=31))

Apply Batch Normalization:
   model.add(BatchNormalization())

Apply Dropout:
   model.add(Dropout(rate=0.5))

Add Second Dense Layer:
   model.add(Dense(units=32, activation='relu', kernel_initializer='he_normal'))

Apply Batch Normalization:
   model.add(BatchNormalization())

Apply Dropout:
   model.add(Dropout(rate=0.5))
Add Output Layer:

 model.add(Dense(units=num_classes, activation='softmax'))
Compile Model:
   model.compile(optimizer=Adam(learning_rate=0.001),
loss='categorical_crossentropy', metrics=['accuracy'])

Train Model:
   model.fit(X_train, Y_train_encoded, epochs=100, batch_size=32,
validation_split=0.2)
```

4.4.2. Experimental Setup

Table 4: Experimental Setup Parameters

Component	**Configuration/Details**
Model Architecture	Sequential
First Dense Layer	Units: 64, Activation: ReLU, Kernel Initializer: He Normal, Input Dimension: 31
Batch Normalization 1	Applied after the first dense layer
Dropout 1	Rate: 0.5
Second Dense Layer	Units: 32, Activation: ReLU, Kernel Initializer: He Normal
Batch Normalization 2	Applied after the second dense layer
Dropout 2	Rate: 0.5
Output Layer	Units: num_classes (depends on your problem), Activation: Softmax
Optimizer	Adam, Learning Rate: 0.001
Loss Function	Categorical Crossentropy
Metrics	Accuracy
Training Data	Features: X_train, Labels: Y_train_encoded
Training Configuration	Epochs: 100, Batch Size: 32, Validation Split: 20%

4.4.3. Training Accuracy:

The training accuracy shows how well a model makes predictions on the data that it has been trained. This model will then adjust its weights and biases in a process that reduces the difference between what it predicts across all samples based on those new parameters.

Usually, the more training progresses and better it fits to the training data points, higher is the accuracy of a model. Keep in mind though, that high training accuracy is not the only motivation to choose a model. This is due to the fact that overfitting might happen, resulting in a model too specialised to the training data which could harm its ability generalise on new unseen original-to-the-model data.

4.4.4. Validation Accuracy:

Validation vulnerability is a technique used to measure how well does the model generalised in outer data it had not seen during training time. Using only the training data, a subset of the dataset is kept as validation set. The performance of the model has been evaluated against this validation set during training. How well you expect the model to do on data it has not yet seen is given by the correctness of this validation. It helps us to identify overfitting. When the training accuracy is increasing while the validation one stays steady or drops then we are in an overfitting trouble.

During the process of deep learning training, the following are some basic steps that we might do to monitor the correctness of training and validation:

4.4.5. Forward pass:

- Calculate predictions on the training data with that same model ·
- Update the weights and biases of model for minimizing training loss (backword pass)
- Calculate the loss for that prediction using your choice of a Loss function.
- Then monitor these metrics and fine tune hyper parameters.

4.5. Training and Validation Loss

Training loss and validation loss are probably one of the most important metrics in deep learning for checking performance on a neural network during training. The first and most basic is the training loss, which helps us measure how much difference there is among the model's predictions on our train data as opposed to what it should have actually predicted. It measures how much the model is deviating from making accurate predictions on the data it used to learn and usually depicted as a single number. During training, the model learns to adjust its parameters such that it minimizes this loss hoping for reducing train loss. Validation loss, however, measures the error for a separate data (the model has not seen these examples while training) called validation set. This score provides an indication of how well the model should expect to generalize its learning on new, unseen data. A common pattern when training a deep learning model is that while the loss on your dataset moves in an encouraging direction (downward), it begins to degrade after enough epochs, especially if you are starting from scratch.

It is very important to keep track of training and validation loss, because that will give you an idea how the model is learning. So essentially, both losses should decrease with time which means that our model is learning. If model loss starts to train down, but the validation loss goes up (left side of figure), then this is a sign that only memorizing your training data rather than generalization on new cases.

4.5.1. Training Loss:

The training loss is a measure that researchers use to understand how well the neural network is working on training data. It gives a numerical measure of the absence between predicted outcome by model with real target values in training data; The aim of training is to minimize this loss as far as possible. For classification tasks, categorical cross-entropy is the most commonly used loss function. The other common loss that gets used for both regression and binary/multi-classification problems is mean squared error (MSE). The training loss reduces as the model gets better at predicting well according to its train data.

4.5.2. Low training loss:

A low training loss is a signal that the model fits well with its training data. However, this does not mean that the model will work well with new (unseen) data; it’s where validation loss comes in.

4.5.3. Validation Loss:

The validation loss is a measure of how well the neural network generalizes, in simple terms it can be interpreted as the amount error your weights learned on test data that they had never seen before. The computation is done on the separate dataset called validation data. This dataset is not used for training purposes and but rather to measure the model performance during or after the training. Overfitting is when a model works well with the training data but poorly on new, unseen examples. This insights the validation loss to avoid overfitting.

4.6. Accuracy and Loss

In the experiment, the proposed model was trained on a dataset with varying spatial resolutions to identify the optimal results and prevent overfitting. The model underwent training for a range of 1 to 100 epochs to ensure thorough learning and evaluation. The figures provided showcase the trends in accuracy and loss for the training dataset over the course of these 100 epochs. These trends are crucial for understanding the model's learning progress and for making adjustments to improve its performance. The goal is to achieve a balance where the model demonstrates high accuracy and low loss, indicating effective learning and generalization capabilities.

4.6.1. Accuracy Graph

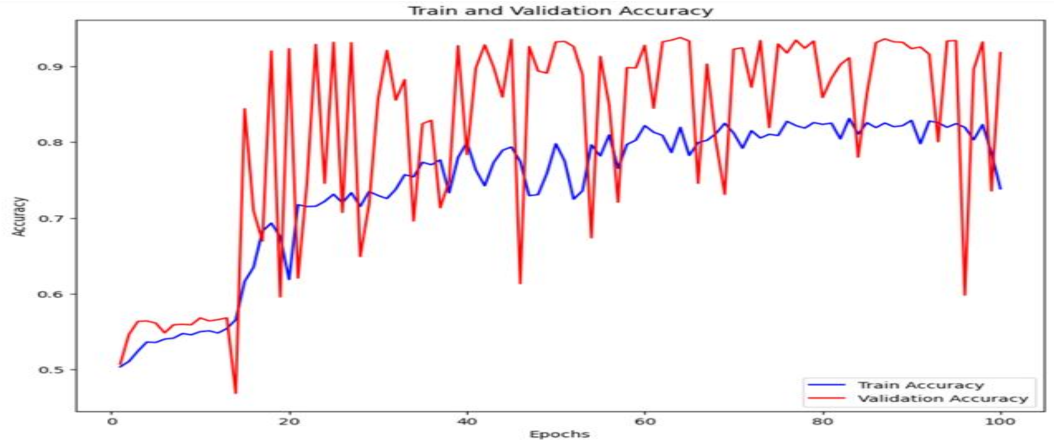

Figure 7: Graph of Accuracy

4.6.2. Loss Graph

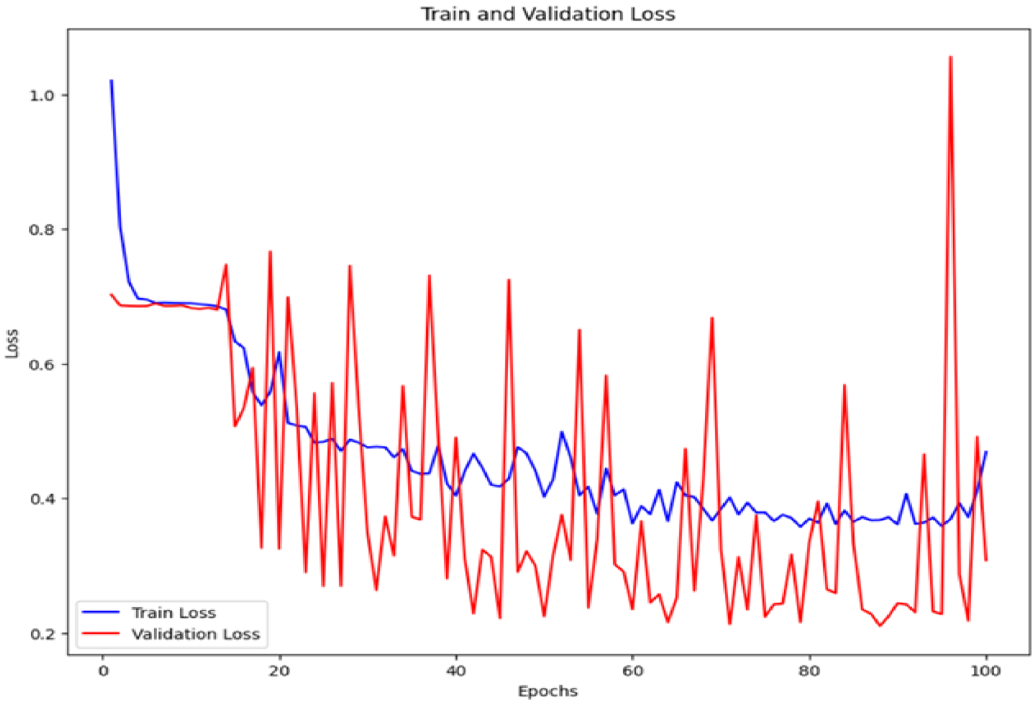

Figure 8: Loss Graph

4.7. Performance Evaluation Parameters

System effectiveness is judged on the bases of various measures like Accuracy, Precision, Recall and F1 Score To analyse an implemented deep learning model on a particular data, these parameters are considered for the performance.

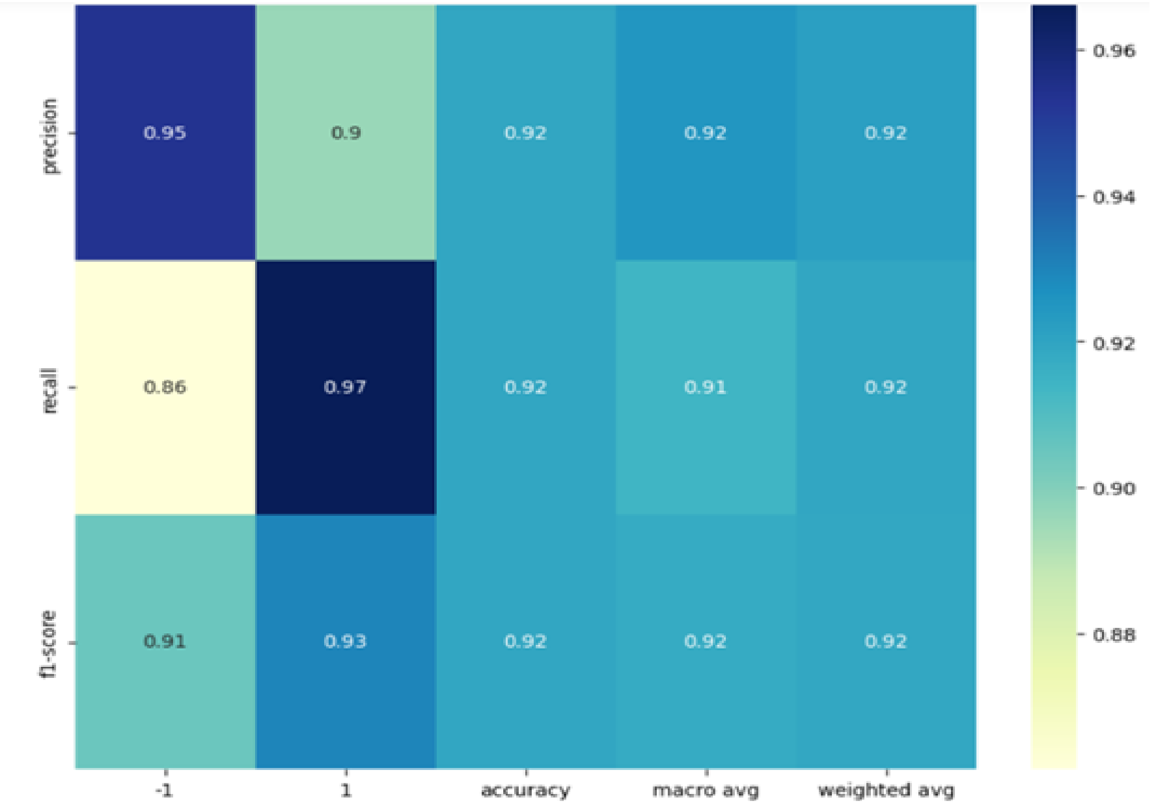

Figure 9: Evaluation Matrix

4.7.1. Confusion Matrix

A confusion matrix is used to generate a brief summary of how well your model performed in terms of automatically classifying tweets. A confusion matrix is a square matrix where rows corresponding to actual classes and the columns correspond to predicted classes. Each diagonal element of the main shows true predictions in dim ension and the other smaller, off-diagonal elements indicate wrong classifications. The matrix is used to calculate the metrics such as accuracy, precision, recall and F1-score providing a consolidated view of model performance along with scopes where enhancements are indeed required.

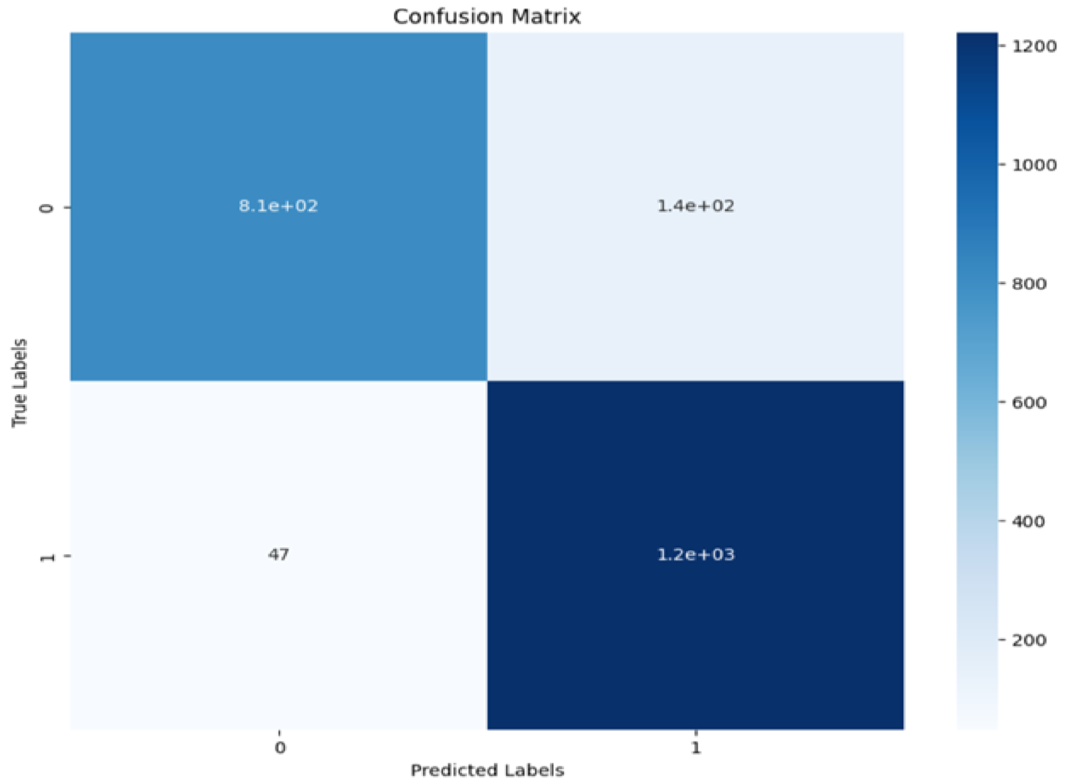

Figure 10: Confusion Matrix

4.7.2. Box Plot

A box plot (or Whisker plot) is a way of showing the main features or measures in a data set. It is a box from Q1 to Q3 with the line at median (Q2 — or 50%(p/100)). Whiskers are drawn to 1.5 * IQR (which defines the extent of whisker) from first and third quadrant points Individual data points are outliers. Box plots are great tools for viewing distributions, judging skewness and spotting outliers.

We can see the differences in central tendency, spread and whether or not there are outliers for each of 10 variables just by looking at ten box plots. It can give you insights about the distribution characteristics of any column which could help in finding out data anomalies and taking decisions related to pre-processing the same, or feature selection, else also know how a variable is dependent on other variables.

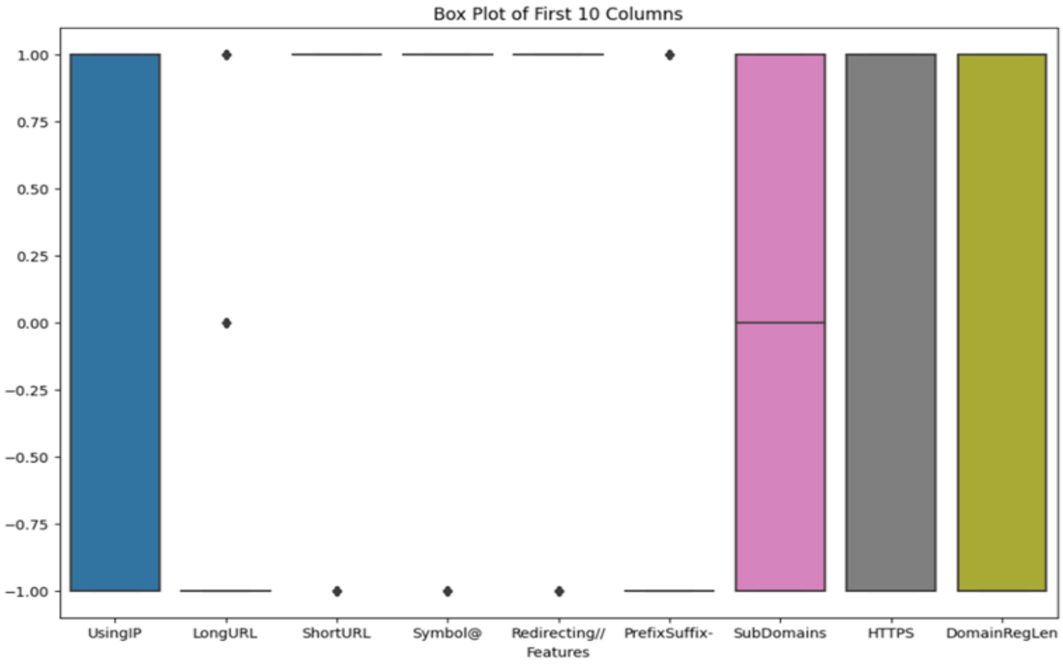

Figure 11: Box Plot

4.7.3. Correlation Matrix

A correlation matrix shows correlation coefficients between variables, ranging from -1 (perfect negative) to 1 (perfect positive), with 0 indicating no correlation. Each cell represents the correlation between two variables. This matrix helps identify relationships between features, informing feature selection and engineering in machine learning. Strong correlations suggest redundancy, while weak correlations indicate unique information. A heatmap visualization aids in spotting patterns and relationships quickly.

A table that displays the correlation coefficients between several sets of variables in a dataset is referred to as a correlation matrix at the moment. In addition to providing a measurement of the linear relationship between variables, it also indicates the direction and degree of the correlation between the variables.

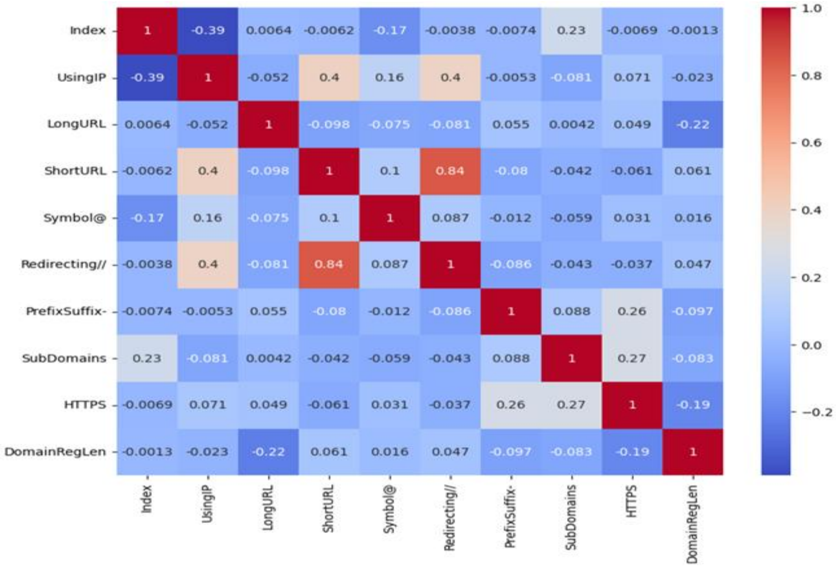

Figure 12: Correlation Matrix

4.8. Results Comparisons

Table 5: Comparison of the Results

Algorithm Name	Date	Accuracy	Remarks
RBF-Support Vector Machine [39]	2022	73%	• Training time can be long for large datasets
Hybrid Machine Learning Algorithms {KNN, SVM}[42]	2022	{93%, 91%}	• It may not capture complex relationships in data
KNN [41]	2021	85.08%	• KNN can easily overfit
CNN and LSTM Classifier [40]	2023	93.4%	• Can be prone to overfitting if not properly regularized.
CNN-LSTM based Deep Learning Model		97%	• Capture complex patterns in data. • Highly flexible and can handle both structured and unstructured data.

The table below depicts comparison of different algorithms for phishing detection based on their accuracy, date of operation and extra remarks regarding the way they perform and nature. In 2022 the first algorithm, RBF-Support Vector Machine (SVM), was implemented and showed an accuracy of 73%. Though SVMs are good at classification tasks, specifically in the binary class case and they have a strong capability of generalizing (Boser et al. 1992), their training time would take too long for massive datasets. In cases where quick response to ever changing phishing tactics is needed, this could be a serious limitation as the model will Not quickly deploy or retrain.

The next entry is from a hybrid model using K-Nearest Neighbors (KNN) and SVM in 2022. This hybrid method achieved an accuracy of 93 % with KNN and 91% using SVM. Both are weak due to different reasons: KNN is said because it cannot capture the hyperplane potentiality (boundaries of separation) and SVM being considered slow, so their combinations called Ensemble learning functions on putting these two algorithms in conjunction as this helps both of them for example- making dataset linearly separable. But the hybrid may not handle more complicated relationships in that data so easily. This limitation implies that whilst the hybrid model does well in general, it may not be optimal for datasets with complex relationships between variables.

In 2021, the third applied algorithm KNN had reached an accuracy of 85.08%. KNN is well- known for its simplicity and threatening results in some cases, typically under non-linear decision boundary. The major disadvantage of KNN is that it falls victim to overfitting, in particular when the dataset owns large set of features or model hasn't an option tuned accurately. Overfitting Overfitting happens when the model is too trained on the training data so that it does not generalise well with new, unseen data. It is this property of KNN based model which makes it necessary that you must apply proper feature selection and the fine-tune your models so as to make them still be generalistic in a real world scenario.

The table also shows the results of a CNN and LSTM Classifier, which was tested in 2023 (CNN + LSTM) with an accuracy value equal to 93.4%. The proposed model uses both Convolutional Neural Networks (CNN) and Long Short-Term Memory networks.

CNNs are inherently goodat ihrvesting spatial hierarchies and local patterns in data, so they are the goto neural network for structured input (images or time-series) tasks. LSTMs, in contrast, are more suited to sequential data by capturing temporal dependencies that have importance for tasks like natural language processing and time-series prediction. However, the model has high accuracy and it is easy to overfit since not well regularized. For deep learning models (such as CNNs and LSTMs), the problem of overfitting can be handled with methods like drop-out which randomly drops units from training, along with changes to hyperparameters such as different learning rates or changing batch sizes.

Tenure (without any specific application year mentioned) The CNN-LSTM Model based Deep Learning model that performs best with accuracy of 97% Such a model is famed for being able to capture very intricate patterns into data which makes it really good at tasks where we need to identify sophisticated relationships inside the dataset. It can be programmed to receive virtually any form of input, making it extremely versatile in complex tasks like phishing detection. Yet this complexity demands both extensive computation resources for training and careful regularization to avoid overfitting. The high accuracy is an indicator that the model's measure of success in real-world implementation, where capacity to generalize across diverse and possibly noisy datasets matters.

4.9. Mitigating Cyber Threats using Deep Learning

The sudden increase in the number of phishing attacks that are directed at individuals and businesses has brought to light the critical requirement for real-time detection systems in order to reduce the dangers that are connected with these threats. Phishing attacks make use of bogus emails and websites that spoof real services in order to trick individuals into disclosing sensitive information such as login passwords and financial data. The goal of these attacks is to deceive individuals into divulging this information. Phishing assaults can have severe repercussions, including money losses, identity theft, and damage to one's reputation. These implications might manifest themselves in a variety of settings. Phishing URL detection systems that operate in real time make use of deep learning techniques to examine URLs and the content of websites. This allows for the rapid identification of phishing websites and the prevention of potential harm before it occurs. These systems are able to adapt to the ever-changing strategies that fraudsters use in phishing attempts because they continuously monitor and analyze web traffic.

Researchers have successfully developed a deep learning technique for real-time phishing URL identification by making use of testing data. The technique has achieved an astounding accuracy rate of 97.87%, which is rather remarkable. This model outperforms earlier approaches in terms of precision, which is essential for reducing the number of false positives and ensuring that reputable websites are not mistakenly identified as phishing threats. Phishing attacks are a big problem that has to be addressed, and the development of this real-time detection system marks a significant advancement in this regard. It provides swift security measures to limit the impact of these cyber threats. The findings of this study, taken as a whole, provide evidence that utilizing deep learning technology to resist phishing attacks is beneficial. Furthermore, it highlights the significance of taking preventative actions in order to secure against cyber risks in the current digital ecosystem.

CHAPTER 5

5. CONCLUSION AND FUTURE WORK

5.1. Conclusion

In conclusion, this research underscores the critical importance of real-time phishing URL detection systems in combating the rising sophistication of phishing attacks.

Real-time phishing Uniform Resources Locator URL detection is important due to the growing threat of phishing attacks on individuals and businesses. These attacks seek usernames, passwords, and credit card numbers. Fake emails and websites enable these attacks.

By leveraging advanced machine learning techniques, particularly deep learning models like CNNs and LSTMs, the proposed approach achieved an impressive accuracy rate of 97% in detecting phishing URLs. This outcome highlights the system's ability to minimize false positives and effectively differentiate between legitimate and malicious URLs.

The method demonstrated significant improvements over traditional approaches, showcasing robustness, adaptability to evolving phishing tactics, and enhanced precision. These results underscore the potential of such systems to strengthen cybersecurity measures, safeguard sensitive information, and mitigate risks to individuals and organizations in the dynamic digital landscape.

5.2. Future Work:

In future work, the research can focus on integrating multiple deep learning architectures or employing ensemble methods to further strengthen phishing detection systems. Combining models such as CNNs, LSTMs, and transformer-based networks could exploit their complementary strengths, leading to improved accuracy and resilience against sophisticated phishing techniques. Additionally, incorporating real-time threat intelligence and adapting the model to new phishing strategies would enhance its adaptability to the evolving threat landscape.

Expanding the study to evaluate the model's performance across diverse and extensive datasets could provide insights into its scalability and robustness in various scenarios. Furthermore, deploying the system in real-world cybersecurity environments would enable a practical assessment of its effectiveness, uncover potential limitations, and guide the development of even more reliable and efficient phishing detection frameworks.

References

1. C. Florackis, C. Louca, R. Michaely, and M. Weber, ‘Cybersecurity risk’, *The Review of Financial Studies*, vol. 36, no. 1, pp. 351–407, 2023.

2. C. M. Igwilo and V. T. Odumuyiwa, ‘Comparative analysis of ensemble learning and non-ensemble machine learning algorithms for phishing url detection’, *FUOYE Journal of Engineering and Technology*, vol. 7, no. 3, pp. 305–312, 2022.

3. M. N. Alam, D. Sarma, F. F. Lima, I. Saha, S. Hossain, and Others, ‘Phishing attacks detection using machine learning approach’, in *2020 third international conference on smart systems and inventive technology (ICSSIT)*, 2020, pp. 1173–1179.

4. N. Abdelhamid, F. Thabtah, and H. Abdel-Jaber, ‘Phishing detection: A recent intelligent machine learning comparison based on models content and features’, in *2017 IEEE international conference on intelligence and security informatics (ISI)*, 2017, pp. 72–77.

5. Z. Alkhalil, C. Hewage, L. Nawaf, and I. Khan, ‘Phishing attacks: A recent comprehensive study and a new anatomy’, *Frontiers in Computer Science*, vol. 3, p. 563060, 2021.

6. R. Basnet, S. Mukkamala, and A. H. Sung, ‘Detection of phishing attacks: A machine learning approach’, in *Soft computing applications in industry*, Springer, 2008, pp. 373–383.

7. R. Mahajan and I. Siddavatam, ‘Phishing website detection using machine learning algorithms’, *International Journal of Computer Applications*, vol. 181, no. 23, pp. 45–47, 2018.

8. N. Abdelhamid, A. Ayesh, and F. Thabtah, ‘Phishing detection based associative classification data mining’, *Expert Systems with Applications*, vol. 41, no. 13, pp. 5948–5959, 2014.

9. A. K. Jain and B. B. Gupta, ‘A survey of phishing attack techniques, defence mechanisms and open research challenges’, *Enterprise Information Systems*, vol. 16, no. 4, pp. 527–565, 2022.

10. Athulya and Praveen, ‘Towards the detection of phishing attacks’, in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, Tirunelveli, India, 2020.

11. B. Biswas, A. Mukhopadhyay, A. Kumar, and D. Delen, ‘A hybrid framework using explainable AI (XAI) in cyber-risk management for defence and recovery against phishing attacks’, *Decision Support Systems*, vol. 177, p. 114102, 2024.

12. A. Sadiq *et al.*, ‘A review of phishing attacks and countermeasures for internet of things-based smart business applications in industry 4.0’, *Human behavior and emerging technologies*, vol. 3, no. 5, pp. 854–864, 2021..

13. C. L. Tan, K. L. Chiew, N. Musa, and D. H. A. Ibrahim, ‘Identifying the most effective feature category in machine learning-based phishing website detection’, *International Journal of Engineering & Technology*, vol. 7, no. 4.31, pp. 1–6, 2018.

14. H. Shirazi, B. Bezawada, I. Ray, and C. Anderson, ‘Adversarial sampling attacks against phishing detection’, in *Data and Applications Security and Privacy XXXIII: 33rd Annual IFIP WG 11.3 Conference, DBSec 2019, Charleston, SC, USA, July 15--17, 2019, Proceedings 33*, 2019, pp. 83–101.

15. J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran, and B. S. Bindhumadhava, ‘Phishing website classification and detection using machine learning’, in *2020 international conference on computer communication and informatics (ICCCI)*, 2020, pp. 1–6.

16. A. Zamir *et al.*, ‘Phishing web site detection using diverse machine learning algorithms’, *The Electronic Library*, vol. 38, no. 1, pp. 65–80, 2020.

17. A. Safi and S. Singh, ‘A systematic literature review on phishing website detection techniques’, *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 2, pp. 590–611, 2023.

18. M. N. Alam, D. Sarma, F. F. Lima, I. Saha, S. Hossain, and Others, ‘Phishing attacks detection using machine learning approach’, in *2020 third international conference on smart systems and inventive technology (ICSSIT)*, 2020, pp. 1173–1179.

19. Z. Alshingiti, R. Alaqel, J. Al-Muhtadi, Q. E. U. Haq, K. Saleem, and M. H. Faheem, ‘A deep learning-based phishing detection system using CNN, LSTM, and LSTM-CNN’, *Electronics*, vol. 12, no. 1, p. 232, 2023.

20. S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, ‘Comparing different supervised machine learning algorithms for disease prediction’, *BMC medical informatics and decision making*, vol. 19, no. 1, pp. 1–16, 2019.

21. M. Vijayalakshmi, S. Mercy Shalinie, M. H. Yang, and R. M. U, ‘Web phishing detection techniques: a survey on the state-of-the-art, taxonomy and future directions’, *Iet Networks*, vol. 9, no. 5, pp. 235–246, 2020.

22.

Y. Li, Z. Yang, X. Chen, H. Yuan, and W. Liu, ‘A stacking model using URL and HTML features for phishing webpage detection’, *Future Generation Computer Systems*, vol. 94, pp. 27–39, 2019.

23. R. Liu, Y. Lin, X. Yang, S. H. Ng, D. M. Divakaran, and J. S. Dong, ‘Inferring phishing intention via webpage appearance and dynamics: A deep vision based approach’, in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1633–1650.

24. R. S. Rao, A. R. Pais, and P. Anand, ‘A heuristic technique to detect phishing websites using TWSVM classifier’, *Neural Computing and Applications*, vol. 33, no. 11, pp. 5733–5752, 2021.

25. C. L. Tan, K. L. Chiew, N. Musa, and D. H. A. Ibrahim, ‘Identifying the most effective feature category in machine learning-based phishing website detection’, *International Journal of Engineering & Technology*, vol. 7, no. 4.31, pp. 1–6, 2018.

26. A. Subasi, E. Molah, F. Almkallawi, and T. J. Chaudhery, ‘Intelligent phishing website detection using random forest classifier’, in *2017 International conference on electrical and computing technologies and applications (ICECTA)*, 2017, pp. 1–5.

27. M. Adil, R. Khan, and M. A. N. U. Ghani, ‘Preventive techniques of phishing attacks in networks’, in *2020 3rd International Conference on Advancements in Computational Sciences (ICACS)*, 2020, pp. 1–8.

28. A. Aljofey, Q. Jiang, Q. Qu, M. Huang, and J.-P. Niyigena, ‘An effective phishing detection model based on character level convolutional neural network from URL’, *Electronics*, vol. 9, no. 9, p. 1514, 2020.

29. W. Yao, Y. Ding, and X. Li, ‘Deep learning for phishing detection’, in *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, 2018, pp. 645–650.

30. C. Opara, B. Wei, and Y. Chen, ‘HTMLPhish: Enabling phishing web page detection by applying deep learning techniques on HTML analysis’, in *2020 International Joint Conference on Neural Networks (IJCNN),* 2020, pp. 1–8.

31. M. Korkmaz, E. Kocyigit, O. K. Sahingoz, and B. Diri, ‘Phishing web page detection using N-gram features extracted from URLs’, in *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA),* 2021, pp. 1–6.

32. F. Tajaddodianfar, J. W. Stokes, and A. Gururajan, ‘Texception: a character/word-level deep learning model for phishing URL detection’, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2020, pp. 2857–2861.

33. S. Y. Yerima and M. K. Alzaylaee, ‘High accuracy phishing detection based on convolutional neural networks’, in *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS),* 2020, pp. 1–6.

34. N. Q. Do, A. Selamat, O. Krejcar, T. Yokoi, and H. Fujita, ‘Phishing webpage classification via deep learning-based algorithms: an empirical study’, *Applied Sciences*, vol. 11, no. 19, p. 9210, 2021.

35. M. A. Adebowale, K. T. Lwin, and M. A. Hossain, ‘Intelligent phishing detection scheme using deep learning algorithms’, *Journal of Enterprise Information Management*, vol. 36, no. 3, pp. 747–766, 2023.

36. B. Janet, S. Reddy, and Others, ‘Anti-phishing System using LSTM and CNN’, in *2020 IEEE International Conference for Innovation in Technology (INOCON),* 2020, pp. 1–5.

37. H. Chapla, R. Kotak, and M. Joiser, ‘A machine learning approach for url based web phishing using fuzzy logic as classifier’, in *2019 International Conference on Communication and Electronics Systems (ICCES),* 2019, pp. 383–388.

38. R. A. A. Helmi, C. S. Ren, A. Jamal, and M. I. Abdullah, ‘Email anti-phishing detection application’, in *2019 IEEE 9th International Conference on System Engineering and Technology (ICSET),* 2019, pp. 264–267.

39. F. Salahdine, Z. El Mrabet, and N. Kaabouch, ‘Phishing attacks detection a machine learning-based approach’, in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON),* 2021, pp. 0250–0255.

40. M. A. Adebowale, K. T. Lwin, and M. A. Hossain, ‘Intelligent phishing detection scheme using deep learning algorithms’, *Journal of Enterprise Information Management*, vol. 36, no. 3, pp. 747–766, 2023.

41. T. A. Assegie, ‘K-nearest neighbor based URL identification model for phishing attack detection’, *Indian Journal of Artificial Intelligence and Neural Networking*, vol. 1, no. 2, pp. 18–21, 2021.

42. R. Yang, K. Zheng, B. Wu, C. Wu, and X. Wang, ‘Phishing website detection based on deep convolutional neural network and random forest ensemble learning’, *Sensors*, vol. 21, no. 24, p. 8281, 2021.

43. M. R. Islam *et al.*, ‘PhishGuard: A Convolutional Neural Network Based Model for Detecting Phishing URLs with Explainability Analysis’, *arXiv preprint arXiv:2404. 17960*, 2024.

44. A. Ejaz, A. N. Mian, and S. Manzoor, ‘Life-long phishing attack detection using continual learning’, *Scientific Reports*, vol. 13, no. 1, p. 11488, 2023.