THESIS

MACHINE LEARNING-BASED PHISHING DETECTION USING URL FEATURES: A COMPREHENSIVE REVIEW

Submitted by

Asif Uz Zaman Asif

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall 2023

Master's Committee:

Advisor: Indrakshi Ray
Co-Advisor: Hossein Shirazi

Indrajit Ray
Haonan Wang

ABSTRACT

MACHINE LEARNING-BASED PHISHING DETECTION USING URL FEATURES: A COMPREHENSIVE REVIEW

In a social engineering attack known as phishing, a perpetrator sends a false message to a victim while posing as a trusted representative in an effort to collect private information such as login passwords and financial information for personal gain. To successfully carry out a phishing attack, fraudulent websites, emails, and messages that are counterfeit are utilized to trick the victim. Machine learning appears to be a promising technique for phishing detection. Typically, website content and Unified Resource Locator (URL) based features are used. However, gathering website content features requires visiting malicious sites, and preparing the data is labor-intensive. Towards this end, researchers are investigating if URL-only information can be used for phishing detection. This approach is lightweight and can be installed at the client's end, they do not require data collection from malicious sites and can identify zero-day attacks. We conduct a systematic literature review on URL-based phishing detection. We selected recent papers (2018 –) or if they had a high citation count (50+ in Google Scholar) that appeared in top conferences and journals in cybersecurity. This survey will provide researchers and practitioners with information on the current state of research on URL-based website phishing attack detection methodologies. The results of this study show that, despite the lack of a centralized dataset, this is beneficial because it prevents attackers from seeing the features that classifiers employ. However, the approach is time-consuming for researchers. Furthermore, for algorithms, both machine learning and deep learning algorithms can be utilized since they have very good classification accuracy, and in this work, we found that Random Forest and Long Short-Term Memory are good choices of algorithms. Using task-specific lexical characteristics rather than concentrating on the number of features is essential

for this work because feature selection will impact how accurately algorithms will detect phishing URLs.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Indrakshi Ray and my co-advisor, Hossein Shirazi for their continuous support for guiding me with this research. Their extensive knowledge and experience have been instrumental in shaping my academic growth.

This work was supported in part by funding from NSF under Award Numbers CNS 1715458, DMS 2123761, CNS 1822118, NIST, ARL, Statnett, AMI, NewPush, and Cyber Risk Research.

DEDICATION

I would like to dedicate this thesis to my mother, who is my biggest cheerleader.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

A phishing attack is a social engineering attack that is created by the attacker to deceive the victim and attempt to obtain sensitive data with the ultimate goal of stealing the victim's valued possessions. Phishing is an issue that has persisted in the field of computer science since the mid 90's [4], and there is currently no effective method to avoid it. With the growing number of internet users, it appears that we as a community are always attempting to solve this problem and playing catch-up with the perpetrators. One of the reasons for the persistence of phishing is the lack of online literacy among users, which makes it difficult to combat the problem. The other reason is that adversaries are continuously coming up with new ways to trick people into providing personal information on counterfeit websites.

Every year, billions of dollars are lost as a result of phishing catastrophes. Everything is moving online as technology advances, and as a result, phishing attacks have become a very prominent problem in the field of computer security. According to reports provided by the Anti-Phishing Working Group ($APWG$), more than 1.3M phishing attacks were recorded in the fourth quarter of 2022. This quarter's phishing activity was the worst that the $APWG$ has ever recorded, and it was also the first time that the quarterly total exceeded one million [5].

1.1 Phishing Websites

Even if phishing attacks are labeled and executed differently, there is one thing that all phishing attacks have in common: the phisher attempts to redirect users to a phishing site using a malicious URL. URL manipulation is often the first stage in building phishing websites. Consequently, attackers have been working on various means through which a malicious URL can be represented and evade detection. Since the representation of URL keeps on changing, even professionals cannot correctly identify phishing URLs. Before launching an attack, phishers spend time researching their victim and discovering vulnerabilities, ensuring that the assault is successful. In these forms of social engineering attacks, the attacker carefully plants everything, including distractions and a victim's emotional wrath, for the victim to not focus on the attack that is transpiring. A malicious URL created by an attacker to launch a phishing attack poses a significant risk to the targeted victim; therefore, everyone who uses the internet must understand what factors contribute to a malicious URL. The Internet is also used heavily by people without a technological background. Consequently, an automated approach is required to detect phishing URLs. In the past, people used signature-based and rule-based approaches to detect phishing websites. However, these approaches are ineffective against zero-day attacks which are referred to as vulnerabilities that are exploited as soon as they are discovered or even before anyone is aware of them.

1.2 URL-Based Phishing Detection

Machine learning researchers have used URL-based features and content-based features (website images, HTML, and JavaScript code) to distinguish phishing from genuine websites. In this survey, we focussed only on *URL*-based features. A number of reasons motivated this choice. First, machine learning algorithms focusing on lexical characteristics of URL are lightweight and more efficient than those using both content-based and URL-based features. Second, this approach can thwart phishing attacks at the very initial stage when a user stumbles into a potentially harmful URL or phishing campaign. Third, the use of *URL* only features does not require one to visit malicious websites to download content-based features. Visiting malicious websites may cause malware to be loaded which may lead to future attacks. Fourth, *URL*-based classifiers can be installed on clients' mobile devices as they are lightweight – the clients' browsing habits are abstracted from the servers – making them more privacy-preserving.

1.3 Survey methodology

In this survey, we produced a comprehensive review of the research on URL-based phishing detectors using machine learning. We looked into the feature extraction procedure, the datasets, the algorithms, the experimental design, and the results for each work. We looked at the crucial steps in creating a phishing detector, and after analyzing several different approaches, we gave our conclusions regarding the features that may be used, the ideal algorithms, the dataset's current state, and some recommendations. We used two criteria for the paper selection process in this survey. First, we looked into the articles on URL-based phishing detection that has been published in the past five years (2018 onwards) in journals having an impact factor of 2.0 or higher and in conferences from Tier (1, 2, and 3)[1]. We also examined papers having at least 50 citations in Google Scholar. We found 26 papers satisfying our criteria which are given in Table 1.1.

Table 1.1: Selected Papers Details: Publication [Pub], Journal [J], Conference Proceeding [CP], Workshop Proceeding [WP], Quartile [Q]

Ref	Year	Title	Pub.	Impact Factor	Cite
[6]	2023	GramBeddings: A New Neural Network for URL Based Identification of Phishing Web Pages Through N-gram Embeddings	J-Q1	5.105	1
[7]	2023	A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN	J-Q2	2.690	-
[8]	2022	HDP-CNN: Highway deep pyramid convolution neural network combining word-level and character-level representations for phishing website detection	J-Q1	5.105	3
[9]	2022	PDGAN: Phishing Detection With Generative Adversarial Networks	J-Q1	3.367	7
[10]	2022	Website Phishing Detection Using Machine Learning Classification Algorithms	CP	-	-
[11]	2021	Towards Lightweight URL-Based Phishing Detection	J-Q2	3.638	18
[12]	2021	An Explainable Multi-Modal Hierarchical Attention Model for Developing Phishing Threat Intelligence	J-Q1	6.791	8
[13]	2021	Detecting phishing websites using machine learning technique	J-Q1	3.752	17
[14]	2021	URL-based Phishing Websites Detection via Machine Learning	CP	-	-
[15]	2021	Lightweight URL-based phishing detection using natural language processing transformers for mobile devices	J-Q2	0.883	15
[16]	2020	Robust URL Phishing Detection Based on Deep Learning	J-Q3	0.858	11
[17]	2020	A Character-Level BiGRU-Attention for Phishing Classification	CP	-	7
[18]	2020	Visualizing and interpreting rnn models in url-based phishing detection	CP	-	15
[19]	2020	Accurate and fast URL phishing detector: A convolutional neural network approach	J-Q1	5.493	122
[3]	2020	Building robust phishing detection system: an empirical analysis	WP	-	6
[2]	2020	Bypassing detection of URL-based phishing attacks using generative adversarial deep neural networks	CP	-	49
[20]	2020	An evasion attack against ml-based phishing url detectors	J	-	8
[21]	2020	An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network from URL	J-Q2	2.690	55
[22]	2020	Comparison of Classification Algorithms for Detection of Phishing Websites	J-Q2	2.688	15
[23]	2019	PDRCNN: Precise phishing detection with recurrent convolutional neural networks	J-Q2	1.968	52
[1]	2019	Everything Is in the Name – A URL Based Approach for Phishing Detection	CP	-	27
[24]	2019	URL-based Phishing Detection using the Entropy of Non-Alphanumeric Characters	CP	-	14
[25]	2019	Intelligent web-phishing detection and protection scheme using integrated features of Images, frames and text	J-Q1	8.665	104
[26]	2019	Machine learning based phishing detection from URLs	J-Q1	8.665	416
[27]	2018	PHISH-SAFE: URL features-based phishing detection system using machine learning	CP	-	87
[28]	2018	Evaluating deep learning approaches to characterize and classify malicious URL's	J-Q2	1.737	84

[1] We used the following sources for conference rankings: https://people.engr.tamu.edu/guofei/sec_conf_stat.htm

1.4 Our Contribution

In this research, we looked into the feature extraction procedure for both machine learning and deep learning models, the available datasets and popular data sources, the widespread algorithms for both machine learning and deep learning approaches, the experimental design, and the results for each of those works. We looked at each of the crucial steps in creating a phishing detector, and after analyzing several different approaches, we gave our conclusions regarding the features that may be used, the ideal algorithms, the dataset's current state, and some recommendations.

The rest of the paper is organized as follows. Chapter 2 contains the background and the impact assessment for phishing attacks as well as the anatomy of URL. The different feature extraction techniques used by researchers are then illustrated in Chapter 3. Most automated systems are now largely built using machine learning-based algorithms, hence Chapter 4 of the paper discusses the recent algorithms that are being used. The numerous data sources that are used by researchers were covered in Chapter 5. Chapter 6 contains the experimental results. Chapter 7 presents an overview of the survey findings. Chapter 8 contains the conclusion and some recommendations.

Chapter 2

Background

When the internet was just getting started in the early 1990s, phishing first appeared. The term "phishing" is a play on the word "fishing," and it refers to the practice of attracting and catching unwitting victims. Hackers trying to obtain accounts from America Online (AOL) were the ones responsible for the earliest phishing instances. Customers were asked to confirm their accounts by inputting sensitive information such as usernames, passwords, and credit card details into emails that seemed to be from AOL. Scary threats threatening account suspension, if the required information was not provided, were regularly included in these emails, which were crafted to appear genuine. Since then, phishing has continued to threaten cyber security, and it has taken on many forms intending to gather sensitive information from a victim [4].

2.1 Types of Phishing

These scams can be found in several different formats, including emails, Short Message Service (SMS) messages, and social media posts (using URL), to name a few. With the development of technology, the method of phishing attacks has changed. The phishing attacks evolved from SMS or Multimedia Messaging Service (MMS), but today, most are sent via social media and emails. Phishing attacks might be directed at a particular user or a group of users, depending on the attacker's intentions and the resources available. A phishing attack usually occurs when an attacker poses as a representative from a reputable organization and requests that the victim responds to their message immediately by clicking on the provided link [29]. This link then directs the user to a fraudulent website where they are asked to enter their sensitive credentials, allowing the attacker to steal the victim's personal information. Attacks involving malicious links are referred to as smishing when they are sent through SMS or MMS and email phishing when they are sent over email. Vishing, on the other hand, is another type of attack where the attackers use Voice over Internet Protocol ($VoIP$) to coordinate these operations as well as phishers in this situation hide the location of the call while transmitting the audio messages to give the impression that a real conversation is taking place [30].

Phishers also target large corporations as technology is highly expensive since it requires a significant amount of time and research to develop. Competition between businesses is quintessential and as a result, phishers target organizations to steal information, which they subsequently sell or hold for ransom. There are two types of attacks carried out here, and they are targeted depending on the psychology of the organization's employees. The first type of attack is a spear attack, which is directed at a specific individual or organization rather than a large group of individuals [31]. The attackers gather information about the company by using social networking sites like LinkedIn and once prepared they send malicious emails to the organization, and the attackers usually commit a large amount of time and effort to this attack because the payoff is usually greater. The second type of attack is whaling, in which senior workers, such as the Chief Executive officer (CEO) and Chief Financial Officer (CFO), are targeted because they are in a position to offer critical information about a firm [32] [33] [34]. Because this is such a huge attack, phishers take their time and launch the attack when it is least expected, while also attempting to appear as real as possible.

Angler phishing, a new form of phishing attack, emerged as social media usage increased. Social media can be a way for a user to vent out and phishers usually target those users by pretending to be customer service and offering a better service by luring the victims and thus taking personal information from the user. Social media is now a hub for phishers as they gather information and deceive a user by using brand names to trick the victims by showing that they are going to solve users' problems [35].

2.2 Impact and Damage Assessment of Phishing Attacks

The threat of phishing attacks looms big in an increasingly connected world, affecting individuals, organizations, and economies globally. To comprehend the complexities of this issue, we must first investigate the global damages caused by phishing attacks; only then will we be able to devise appropriate ways to combat these phishing threats and safeguard our digital lives. As a result, one of the purposes of this research is to provide an overview of the current status of phishing attacks worldwide.

Phishing is a global threat, and according to a survey by IBM and Ponemon Institute, phishing attacks account for 16% of breaches [36]. Furthermore, according to $APWG$, there were over one million recorded phishing attacks, with 27.7% of these attacks aimed at the financial sector. According to the data, the average Business Email Compromise (BEC) attack sought to steal $132,559. To visualize the trajectory of phishing attacks, we gathered data from the last ten years, beginning in 2013 APWG. The status of phishing attacks is addressed in this organization's quarterly reports. Figure 2.1 shows that phishing attacks have surged about 500% since 2019 [37]. This can be attributed to the global pandemic, COVID-19, since everyone throughout the world was forced to relocate their daily livelihood and their work on the internet, creating an opportunity for adversaries to create new URLs for phishing attacks.

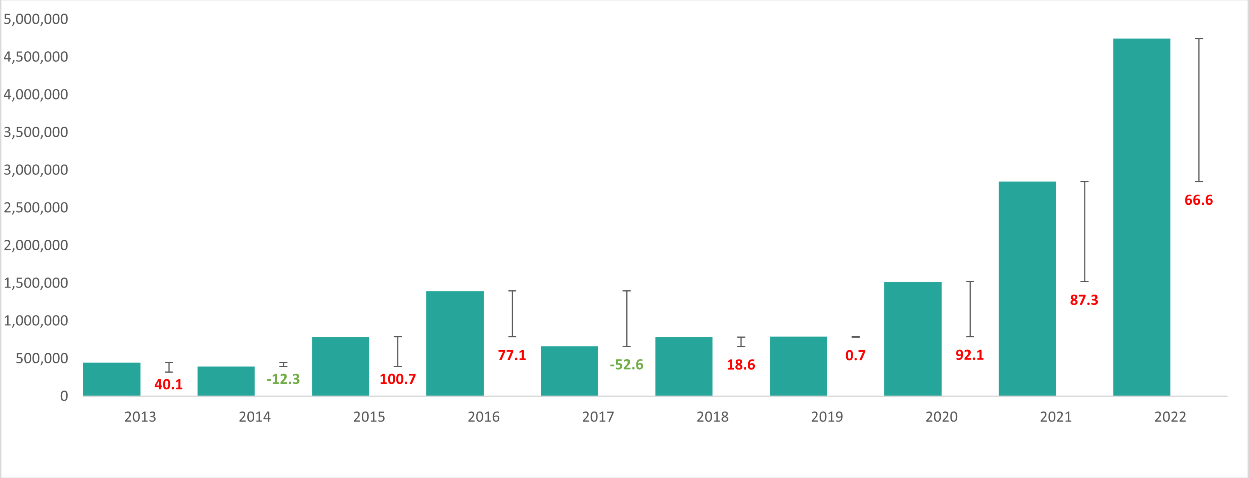

Figure 2.1: Unique Phishing URLs

When we examined the financial damage caused by phishing attacks, we discovered that the actual loss is greater than the stated figure because many businesses do not report phishing attacks to maintain their reputation. However, the reported quantity provides us with an estimate of the current status of phishing attacks and the accompanying losses.

2.2.1 Global Status of Phishing attacks

A startling trend appeared in the United Kingdom's cybersecurity landscape between 2022 - 2023, revealing that phishing attacks had become the dominant risk across numerous sectors. During this time, 83% of UK charities were victims of cyber-attacks. The situation was not much different in the business world, as 79% of organizations reported falling victim to phishing attacks among which 31% of the attacks targeted corporations using impersonation. Furthermore, those aged 25 to 44 were shown to be the most vulnerable to phishing attacks in the UK, underlining the critical need for increased cybersecurity awareness and defenses across the age spectrum. Throughout 2021, phishing attacks remained large over European corporations, accounting for a sizable 42% of all reported cyber threats. The continued difficulty for European firms and organizations to build their defenses against these subtle attacks emphasized the significance of effective cybersecurity measures. The Asia-Pacific region saw a spectacular increase in cybercrime, with recorded incidences increasing by an alarming 168% between 2020 -2021. This frightening trend included a wide range of cyber risks, including the ever-present phishing attacks. Among these digital risks, phishing ranked as the most prominent threat, firmly consolidating its dominance in Asian enterprises' security landscape in 2021. Phishing accounted for 43% of all cyberattacks on the region [38]. Australia suffered a significant financial blow as a result of the BEC attacks, with losses totaling more than AUD 98 million. These astonishing figures were reinforced by a statewide discovery in which Australian firms collectively self-reported losses related to cybersecurity breaches totaling an astounding AUD 33 billion [39]. With numerous types of phishing attempts targeting certain businesses, the threat landscape continued to evolve. Vishing targeted industries such as social media, webmail, and cloud services, as well as telecommunications. Spear phishing, on the other hand, targeted the financial, IT, and healthcare industries. Smishing constituted a substantial threat to social media platforms, webmail and cloud services, job search platforms, telecommunications, and transportation services. Finally, Whaling, a highly targeted kind of phishing, targeted government institutions, the IT and manufacturing industries, and the banking sector [40]. Looking at the situation of phishing attacks in Africa, we discovered that the

Table 2.1: Complaints Count, 2020-2022

	Crime Type	**2022**	**2021**	**2020**
Victim Count	Phishing	300,497	323,972	241,342
Victim Loss, $	Phishing	52,089,159	44,213,707	54,241,075

financial impact of phishing attacks on the economy is not officially disclosed [39]. In the evolving landscape of cybersecurity in South America in 2021, phishing attacks emerged as a dominant threat. This treacherous strategy was used in 47% of the overall cyber attacks conducted at organizations in the region [38]. In North America, particularly in the United States, the cybersecurity situation is stark and frightening. BEC emerged as a significant danger in 2022, inflicting astonishing losses of $2.7 billion, according to Federal Bureau of Investigation (FBI) statistics. This amount is especially stunning when compared to the more generally known issue of credit card theft, which caused $264 million in losses during the same time period [41]. Notably, phishing schemes were used in 47% of attacks against organizations in North America in 2021 [38]. Additionally, phishing attacks have been among the top five crimes since 2019 according to a report from the FBI [41]. The number of victims and losses that have been documented during the past three years are given in Table 2.1. The United States stands out as the top target for cybercriminals using these deceptive tactics. This trend is fueled in part by the proactive reporting of phishing attacks by a large number of organizations in the United States. It can be seen that in the United States, states with greater per capita income levels are preferred targets for phishing scams. Nevada is one of the states most severely affected by these phishing scams [42].

2.2.2 Phishing impact on Organization and Individual

Phishing attacks can have a wide range of negative implications for organizations. For starters, they frequently cause direct financial losses because cybercriminals use misleading tactics to drain payments or compromise key financial information. Aside from the immediate financial cost, these attacks can cause significant damage to an organization's reputation, eroding confidence and credibility among stakeholders. As a result, a loss of trust can lead to a decrease in customer confidence and, in certain situations, the loss of loyal customers. Furthermore, phishing incidents can interrupt operations, producing delays and productivity setbacks. Organizations' overall worth may suffer as a result of the cumulative effect, and regulatory fines may exacerbate the financial strain. Individuals are heavily impacted by phishing attempts in a variety of ways. Identity theft is one of the most distressing effects. When personal information, such as social security numbers or bank information, is compromised, it can result in significant financial losses and a lengthy battle to reclaim one's identity. Aside from the financial burden, the psychological impact is significant, impacting mental well-being. Victims frequently report increased worry, anxiety, and a loss of trust in both online conversations and their own judgment. Furthermore, productivity suffers as people deal with the fallout from phishing assaults, devoting significant time and energy to resolving difficulties, changing passwords, and protecting their online presence [43].

2.2.3 Time to contain Phishing Attack

According to a study done by IBM and the Ponemon Institute, the consequences of phishing attempts are not only financially devastating but also time-consuming to resolve. According to their research, it takes an average of 219 days, or almost 7 months, for firms to even identify that a phishing effort has occurred. Furthermore, once found, it takes an average of 76 days, or almost two and a half months, to completely eliminate the threat [36], which is illustrated in Figure 2.2

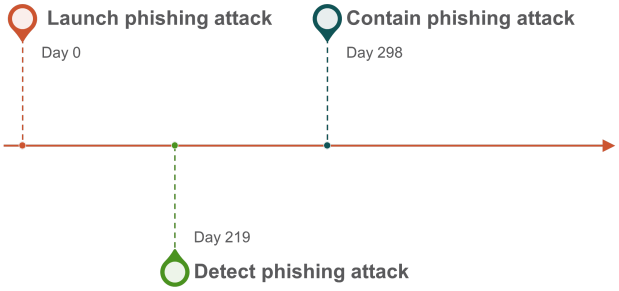

Figure 2.2: Timeline to contain phishing

Hence, in this study, we focused on URL based on phishing strategies so that phishing attacks can be contained in real-time and at an early stage.

2.3 URL Anatomy

The anatomy of the URL is critical for understanding how attackers manipulate it for launching phishing attacks. Figure 2.3 shows the structure of a legitimate URL and the different components that are present within the URL. An attacker may manipulate any segment of the URL to create a malicious link that can be used to launch a phishing attack.

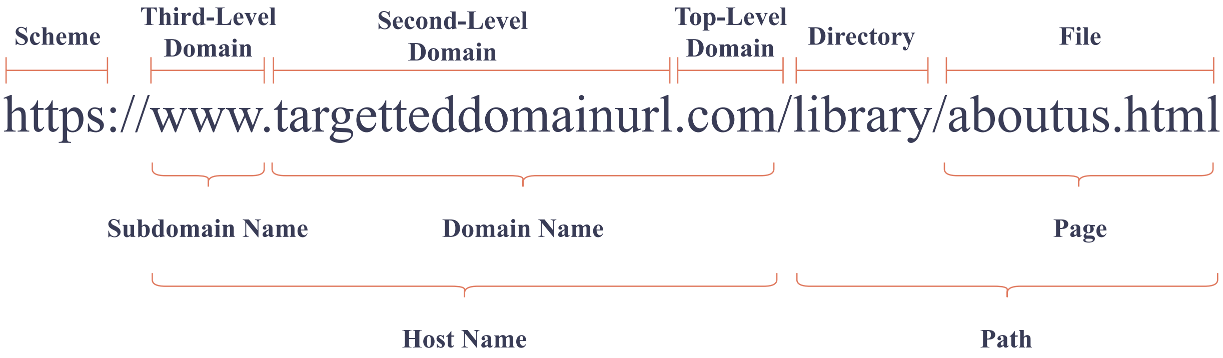

Figure 2.3: URL anatomy breakdown into individual components

The URL of a website is made up of four major components: scheme, host, path, and query string. The scheme specifies the protocol used by the URL, with the two most common being Hypertext Transfer Protocol ($HTTP$) and Hypertext Transfer Protocol Secure ($HTTPS$). Although $HTTPS$ is more secure, attackers are now utilizing it to make the URL appear secure. The host typically refers to the targeted server where the resources of an application are located, and it can be followed by the port number that an application uses to communicate with the server. The host is further subdivided into three parts: a subdomain, a second-level domain, and a top-level domain. A subdomain is a specific page on a website. The name of the website being accessed is the second-level domain. The top-level domain indicates the type of entity the website is registered as on the internet, with '.com' being the most commonly used. The paths are then used to identify the specific resource that a web client is attempting to access. Finally, a URL may contain a query string, which is a string of information that a resource can use to obtain specific information from a server. This query part is optional, and not all URLs will have it.

Understanding the different components of a URL is important because this is usually the starting point of a phishing attack. An attacker will try to use social engineering to trick a victim so that the malicious URL goes undetected by the victim and also by the detection algorithm. Before launching a phishing attack, the attacker will always take their time and try out multiple combinations to manipulate the URL so that the phishing link does not raise suspicion. To accomplish this goal, the attacker will employ various obfuscation techniques. In this case, the attacker may obfuscate the hostname with the IP address, and the phished website name is placed in the path, for example, http://159.203.6.191/servicepaypal/. They can do something similar by obfuscating the host with another domain name, such as http://a0243562.xsph.ru/servicePayPal/C/, where the hostname contains a valid-looking domain but the phishing website is located in the path. The attacker can also obfuscate with a long domain name, such as https://paypalhelpservice.simdif.com/, in which the organization being phished appears in the URL's subdomain. Furthermore, an attacker can obfuscate a domain name that is unknown or misspelled, such as http://paypa1.com, which is misspelled and unrelated to the actual domain.

All of the techniques mentioned above will attempt to redirect a victim to a malicious site and will entice the victim to provide sensitive information to the attacker, which the attacker will use to exploit the victim. The name of the company PayPal is incorporated in some form in the malicious URL in all of these cases, and a user may not always be able to tell the difference between phishing and a legitimate site. The attacker launches these attacks by creating a sense of urgency for the victim; the victim may get stressed out and become vulnerable to judgemental errors. Thus, in order to prevent URL-based website phishing attacks, an automated approach is required.

Chapter 3

Feature Extraction

Manual feature extraction is required for URL-based website phishing attack detection when using machine learning; this is generally known as using hand-crafted features. However, when a deep learning approach is employed, the feature extraction procedure is done automatically and does not require domain expertise.

Researchers have often used URL lexical features alongside domain features to create a better ML model. Table 3.1 provides a list of features used by the algorithms.

3.1 URL Lexical features

Information that is directly connected to a website's URL components is referred to as URL lexical features. URL-based characteristics include lexical features that keep track of the attributes of the URL, such as its length, domain, and subdomain. Popular lexical elements of URLs include the use of the $HTTPS$ protocol, special characters and their counts (for a dot, a hyphen, and at symbol), numerical characters, and IP addresses.

3.2 Domain Features

Information about the domain on which a website is hosted is included in the domain features. The age of the domain and free hosting is generally included in this feature set as it is a crucial signal for distinguishing between a legitimate website and a phishing website. Typically, a newly hosted website serves as a warning sign for a phishing site.

Table 3.1: Combination of features used in the literature

Ref.	[18]	[1]	[24]	[3]	[27]	[25]	[26]	[11]	[2]	[20]	[12]	[13]	[19]	[28]	[21]	[22]	[8]	[23]	[17]	[16]	[9]	[14]	[7]	[10]	[6]	[15]
Automatic Features													✓	✓							✓					
Hand-Crafted Features	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓		✓				✓	✓	✓		✓
URL Lexical Features	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓						✓	✓	✓	✓	✓
Domain Features	✓	✓	✓		✓	✓					✓				✓								✓			
Word Features	✓	✓	✓		✓		✓	✓		✓							✓									
Character Features																	✓	✓	✓	✓						
Search Index Features						✓																				
Total Features	**17**	**-**	**46**	**51**	**14**	**35**	**104**	**12**	**42**	**93980**	**17**	**-**	**-**	**-**	**95+**	**87**	**-**	**9**	**-**	**-**	**-**	**111**	**30**	**30**	**-**	**48**

3.3 Word Features

Word features prevent a typical user from becoming suspicious. Attackers utilize words like secure, support, safe, and authentic within the URL itself to make it appear real. To make the URL appear legitimate, they also include well-known brand names, such as PayPal, Amazon, and Facebook, inside the URL.

3.4 Character Features

Phishing sites often use suspicious characters. The length of the URL, the usage of uncommon letters or symbols, and misspelled words are a few examples of character-based indicators that are frequently used to identify phishing websites.

3.5 Search Index Based Features

This feature contains details such as the website's page ranking, Google index, and website traffic information. These features are crucial for gathering data about the website and can be used to distinguish between legitimate and fraudulent websites. The average lifespan of a phishing website is quite short, and it typically produces no statistics.

Chapter 4

Algorithms

Researchers frequently compare their suggested models in their experiments with other algorithms. Table 4.1 contains a list of all the algorithms that we commonly encountered while conducting this survey. Both machine learning and deep learning methods were used for classifications to construct a defensive system.

4.1 Classification using Machine Learning

4.1.1 Logistic Regression (LR)

LR is a common statistical machine-learning method for binary classification problems or for predicting an outcome with two possible values and this is specifically required for phishing detection because URL might either be legitimate or fraudulent [1,2,10,12,20,21,23]. LR can process a lot of URLs as it is a computationally efficient technique and can handle big datasets with high-dimensional feature spaces [10, 21]. In order to select the most crucial aspects for phishing URL detection, feature selection can be done using LR models. In addition to increasing the model's effectiveness, this can decrease the input space's dimensionality and works with word-based features [1], character-based features [21] and bi-gram-based features which is, contiguous pairs of words [23]. Additionally, when given a balanced dataset, LR can learn the decision boundary that best discriminates between positive and negative samples without favoring either class [12, 20]. However, in order to train the model, LR needs labeled data. This can be a problem in phishing detection because acquiring labeled data can be challenging [2].

4.1.2 Decision Tree (DT)

DT is a type of supervised machine learning method for classification and regression tasks. It works by iteratively segmenting the input data into subsets based on the values of the input attributes in order to discriminate between the various classes or forecast the target variable. DT

is commonly used by researchers for phishing detection problems [2, 10, 11, 14, 15, 20, 26, 28]. Phishing URLs frequently exhibit traits that set them apart from real URLs. DT algorithm learns to distinguish between legal and phishing URLs using these properties as input to the features needed to train the algorithm. For detecting URL-based phishing, DT is advantageous because it is a highly interpretable model that makes it possible for human specialists to determine the reasoning behind a choice. Given the large potential for feature density in URLs, the feature space is highly dimensional. Without suffering dimension problems, DT can handle this type of data [10]. Moreover, DT that use lexical features can produce a better result, it creates a set of rules based on lexical properties that are simple for human specialists to comprehend [20,26,28]. When working with massive datasets, decision trees offer outcomes with good performance [14, 28]. However, overfitting is common in DT, especially in small or significantly unbalanced datasets. A model may as a result perform well on training data but badly on the newly collected information [11].

4.1.3 Random Forest (RF)

RF is another machine learning method used for classification, regression, and feature selection tasks [2, 3, 10, 11, 15, 16, 19–22, 24, 26, 28]. Because RF can manage large and complex datasets and has the capability to deal with noisy data it is well-adapted for URL-based phishing detection [28]. To make predictions, the ensemble learning method of RF, combines data from various decision trees, reducing the possibility of overfitting while improving the model's generalization capabilities [10, 20–22, 26]. A measure of feature importance can also be provided by RF, which means that this algorithm can be used to understand the key features that contribute to phishing detection, improving the algorithm's overall accuracy [3, 10, 24]. RF is better for the real-time detection of phishing URLs because it is computationally efficient and can be trained on big datasets rapidly as it requires minimal parameter tuning [24]. We also observed that using the lexical features of the URL, RF can produce good performance accuracy [26]. However, the RF algorithm may not work well with imbalanced datasets but it can be observed that on a balanced dataset, it gives better performance [19], [22], [16]. Another disadvantage of using RF is that the model produces better results at the cost of both training and prediction time [11].

4.1.4 Naive Bayes (NB)

NB algorithm is a probabilistic algorithm used in machine learning for classification purposes which is based on Bayes' theorem, to identify URL-based website phishing [10–12, 15, 21–24, 26–28]. NB can manage high-dimensional data, which means the algorithm can handle a large number of features in the URL [21]. NB is susceptible to the model's feature selection, though the model's accuracy may suffer if essential features are excluded [23]. It can therefore work better on small feature sets with important features [10]. Additionally, it was discovered that applying only word-based features to NB does not yield better results [26]. The NB algorithm also has the benefit of learning the underlying patterns in the data with a small quantity of labeled training data, given how difficult it can be to acquire labeled data, this is especially helpful for URL-based phishing detection [22, 28]. When there are an uneven amount of samples in each class, NB may not perform well. This could lead to a model that is biased in support of the dominant class [27]. On a balanced sample, however, this algorithm performance improves [24].

4.1.5 Gradient Boosting (GB)

GB is a machine learning technique that creates a sequence of decision trees, each of which aims to fix the flaws of the one previous to it. The combined forecasts of all the trees result in the final prediction [10, 15, 20–22]. Since GB can be used to train models on huge datasets, it is especially suitable for large-scale phishing attack detection [22]. Additionally, the balanced dataset makes sure that the accuracy of the model is not biased towards one class over another and forces the model to equally understand the underlying patterns of the data for each class. As a result, the model becomes more accurate and generalizable [15, 20, 22]. Moreover, GB is effective when more attributes are considered [20, 21] as well as on character-based features [10, 22]. The model may be less accurate or may not perform well on new, untested data if the training data is biased or insufficient. However, on a balanced dataset, the algorithm performs better [15]. To ensure that the model is able to extract the most informative features from the data, GB necessitates thorough feature engineering. When dealing with complicated and diverse information like URLs, this can be a time-consuming and difficult operation [20, 21].

4.1.6 Adaptive Boosting ($AdaBoost$)

$AdaBoost$ is a machine learning algorithm that is a member of the ensemble learning technique family. This approach for supervised learning can be applied to classification and regression tasks and is also used for URL-based website phishing detection [10, 22, 26, 28]. As an ensemble approach it combines several weak learners to provide a final prediction, $AdaBoost$ is a powerful algorithm that can be a viable choice for URL-based phishing detection [10]. $AdaBoost$ can predict outcomes more precisely when it has access to a larger training dataset. The algorithm can produce predictions that are more accurate by better capturing the underlying relationships and patterns in the data [22, 28]. However, $AdaBoost$ may not be the best option for datasets with a lot of irrelevant or redundant features because it does not directly do the feature selection. This may lead to longer training times and poor results [26].

4.1.7 K-Nearest Neighbour ($K-NN$)

$K-NN$ is an algorithm where a prediction is made based on the labels of the k data points that are closest to an input data point in the training set. In the context of URL-based phishing detection, this means that the algorithm may compare a new URL to a list of known phishing and legitimate URLs and find the ones that are most similar to the new URL and thus are used for URL-based website phishing detection [2, 10, 15, 16, 20, 22, 25, 26]. High-dimensional feature vectors, such as those found in URLs, might be challenging to process. However, the $K-NN$ technique can efficiently detect similarities across URLs and is well-suited to high-dimensional data [22]. Even with imbalanced datasets, where the proportion of samples in one class is significantly higher than the other, the $K-NN$ approach can perform well [15, 20]. Additionally, $K-NN$ works well with word-based features [20, 25, 26]. In $K-NN$ when producing predictions, an algorithm that has a bigger value of k will take into account more neighbors and improve performance [16]. However, the number of nearest neighbors taken into account or the distance measure utilized can have an impact on how well the $K-NN$ method performs. These hyper-parameters may need a lot of effort to be tuned [10]. The $K-NN$ method is susceptible to adversarial attacks, in which a perpetrator creates URLs on purpose to avoid being detected by the system [2, 20].

4.1.8 Support Vector Machine (SVM)

SVM, a form of supervised learning algorithm used in classification and regression analysis, was commonly used by researchers [2, 3, 10–12, 15, 16, 19, 20, 22, 24, 25, 27]. SVM is good for detecting URL-based website phishing because it can handle high-dimensional data and identify intricate connections between features [22]. Numerous characteristics, including the lexical features of the URL, and the existence of specific keywords, can be used to detect phishing when analyzing URLs. These characteristics can be used by SVM to recognize trends in phishing URLs and separate them from real URLs. It can be observed that only using the lexical features of the URL does not yield good results [20]. However, hybrid features like a combination of text, image, and web page content work better for SVM [25]. Hence to achieve optimum performance, SVM requires fine-tuning of several parameters, SVMs additionally can require a lot of computational power, especially when working with big datasets [11]. This may slow down training and prediction times and necessitate the use of powerful hardware [16]. Moreover, the ratio of legitimate URLs to phishing URLs is very uneven, which can result in unbalanced data that will degrade the performance of SVM [24]. However, on a balanced dataset, SVM performs better [19]. Additionally, if there is a lack of training data, SVM's accuracy is likely to decline [27].

4.2 Classification using Deep Learning

4.2.1 Neural Network (NN)

NN uses complex patterns and correlations between input features can be learned. By finding patterns that are suggestive of phishing attempts, NN can learn to differentiate between legitimate and phishing URLs in the context of URL-based phishing detection [14, 15, 21]. The ability of NN to acquire intricate patterns and connections between the characters in a sequence makes them effective for character-based characteristics [21]. Additionally, because the algorithm can learn from the data and produce predictions for each class with nearly equal importance, neural networks can perform well on balanced datasets [15]. However, to perform well, NN needs a lot of high-quality training data. Especially in rapidly changing phishing contexts, collecting and identifying a sufficiently large and diverse array of URLs might be difficult [14].

4.2.2 Multi-Layer Perceptron (MLP)

MLP is another type of NN that has been found to be successful in URL-based phishing detection [11,12,22]. A class imbalance may significantly affect several other algorithms, however, because MLPs employ numerous hidden layers and may thus identify more complex patterns in the data, they are less prone to this problem [11, 12]. However, it can be computationally expensive to train MLPs, especially for larger datasets or more intricate network designs. Long training periods may result from this, which may slow down the deployment of phishing detection systems [22].

4.2.3 Convolutional Neural Network (CNN)

CNN is a class of neural networks that are frequently employed in computer vision, but recently it has emerged to be a great tool for phishing detection [6–9, 16, 19, 21, 23, 28]. When labeled training data is limited, CNNs can benefit from pre-trained models and transfer learning to enhance performance in detecting phishing URLs. CNN is capable of handling variations in the input data, including changes to the URL's length and the existence of unexpected letters or symbols. This is because the pooling layers can downsample the feature maps to lessen the influence of variances, while the convolutional filters used in CNN can recognize patterns in various regions of the CNN [8,21]. Without manual feature engineering, CNN can automatically extract high-level features from the data that comes in. This is because the filters in the convolutional layers are trained to identify the most important data patterns [7, 16, 21, 23]. Additionally, the

CNN performs well on a balanced dataset [6, 19]. It is possible to train more sophisticated CNN architectures that can recognize subtler patterns and correlations in the data with a larger dataset which can increase the model's capacity to correctly categorize new phishing samples [9, 16, 28]. However, if a CNN model fits the training data too closely and cannot generalize to new, untested data, the problem of overfitting arises. This can be prevented by using batch normalization and dropout techniques [9]. Additionally, CNNs can require a lot of processing power, particularly when employing deep structures with numerous layers, therefore, this can need a lot of computing power and hardware resources [7, 16, 23].

4.2.4 Recurrent Neural Network (RNN)

RNN are a type of neural network that excels at processing sequential data such as text or time series data. Because URLs may be represented as a sequence of characters, and because RNNs can learn to recognize patterns and characteristics in this sequence, they can be utilized for URL-based phishing detection [23, 28]. Each character or characteristic in a URL is built sequentially, depending on the ones that came before. These sequential relationships can be observed by RNNs, which can then utilize to forecast whether a URL is genuine or phishing. RNN performance on balanced datasets depends on the particular task at hand as well as the network's architecture. For tasks requiring capturing long-term dependencies and temporal correlations between the input features, RNNs are especially well-suited [23]. To properly learn to recognize patterns in sequential data, such as URLs, RNNs need a lot of training data. This implies that RNNs may not be used efficiently for phishing detection for enterprises with limited access to training data [28]. RNNs can be challenging to understand, particularly when working with massive data sets. RNNs can only be as effective as the training set that they are given. The RNN may struggle to accurately identify new and emerging dangers if the training data is not representative of all the threats that an organization might encounter [28].

4.2.5 Long Short-Term Memory ($LSTM$)

$LSTM$ is a specific type of RNN that was developed to address the issue of vanishing gradients that RNN frequently encounter and thus this algorithm is used by researchers for phishing detection [7–9, 12, 17–19, 28]. The long-term dependencies and sequential patterns in URLs can be captured by $LSTM$, making it a good choice for URL-based website phishing detection. In order to detect tiny variations and patterns in phishing URLs that could otherwise go undetected, $LSTM$ networks are particularly good at identifying sequential data and hence is a good choice for URL-based website phishing attack [9, 12]. $LSTM$s can function well even when trained on minimal amounts of data [28]. These models are perfect for dealing with imbalanced datasets because they can find long-term correlations in the data. For identifying trends in the minority class, these dependencies can be very important [8, 17]. Additionally, $LSTM$ performs poorly for small datasets [7] but performs well on large datasets [18]. However, particularly when using vast data sets, training $LSTM$ models can be computationally and memory-intensive [19]. Overfitting is a possibility with $LSTM$ models, especially when working with limited data. When a model develops a proficiency at recognizing trends in training data but is unable to generalize that skill to fresh, untried data, overfitting occurs. This issue can be solved by using dropout in $LSTM$ [9]. $LSTM$ is complex in nature but the number of parameters needed for an $LSTM$ model can be decreased by using pre-trained word embeddings like Word2Vec [18].

4.2.6 Bidirectional Long Short-Term Memory ($BiLSTM$)

$BiLSTM$ is a form of machine learning-based RNN architecture that is used to detect URL-based website phishing attacks [6, 17, 18, 23]. $BiLSTM$ is a form of neural network design that is effective at detecting data's sequential patterns. The capacity of $BiLSTM$ algorithms is to examine the complete URL string in both ways, i.e., from the beginning to the end and from the end to the beginning, which makes them particularly useful for URL-based phishing detection [6, 18, 23]. Positive instances are often more scarce in imbalanced datasets than negative examples. $BiLSTM$ may simultaneously learn from both phishing and legitimate instances, which may aid in improving its ability to distinguish between the two classes [17, 18]. It can be costly computationally to train $BiLSTM$ networks, especially if the input sequences are large and complex. The algorithm's capacity to scale for very big datasets may be constrained by this [17, 18].

4.2.7 Gated Recurrent Units (GRU)

GRU is a sort of recurrent neural network that has been found to be useful for URL-based phishing detection [17, 18]. GRUs are more memory-efficient and require fewer parameters than other recurrent neural network types. They are thus well suited for use in contexts with limited resources, such as those seen in cloud-based systems or on mobile devices [18]. Additionally, on imbalance datasets, GRUs can perform well [17, 18].

4.2.8 Bidirectional Gated Recurrent Units ($BiGRU$)

$BiGRU$ is a GRU version that captures sequential dependencies in both forward and backward directions. $BiGRU$ is useful for detecting URL-based phishing [17, 18]. There are two layers in $BiGRU$, one of which moves the input sequence forward and the other which moves it backward. This gives the network the ability to record dependencies that happen both before and after a certain input feature, which is helpful for identifying intricate patterns in URLs. Additionally, on imbalance datasets, $BiGRU$s can perform well [17, 18]. However, $BiGRU$ can be computationally demanding to train and may need a lot of resources, such as processing speed and memory. This can make developing and delivering the model difficult, especially on devices with limited resources.

Table 4.1: Frequently used algorithms for phishing detection

No	Algorithms	References
1	Tree Based (RF, DT, $AdaBoost$, GB)	[2,3,10,11,14–16,19–22,24,26,28]
2	SVM	[2,3,10–12,15,16,19,20,22,24,25,27]
3	NB	[10–12,15,21–24,26–28]
4	LR	[1,2,10,12,20,21,23]
5	$K-NN$	[2,10,15,16,20,22,25,26]
6	Sequential Recurrent Networks (RNN, $LSTM$, $BiLSTM$, GRU, $BiGRU$)	[6–9,12,17–19,23,23,28]
7	Neural Network (NN, MLP)	[11,12,14,15,21,22]
8	CNN	[6–9,16,19,21,23,28]

Chapter 5

Dataset

The availability and quality of data are crucial for the performance of machine learning-based phishing detection algorithms. In order to detect phishing attacks, algorithms need to be trained on a large and diverse dataset of both phishing and legitimate URLs of websites. This dataset is used to learn the features that differentiate phishing websites from legitimate ones. Some of the most important features include the structure of the website, the content, and the URLs. The more diverse and representative the data is, the more accurate the algorithms will be at detecting phishing attacks. This is because the algorithms are able to learn from a variety of examples and generalize their understanding of what constitutes a phishing attack. Moreover, it is also important to keep the data up-to-date to reflect the latest trends and techniques used by attackers. As phishing techniques are constantly evolving, algorithms need to be trained on new data to stay effective.

Crawling data from a repository is one of the many ways used by researchers to get data for experiments. To collect phishing URLs, most of the researchers have used *OpenPhish.com* and *PhishTank.com*, but there are different resources for legitimate websites. We also notice that the majority of studies focus on skewed data because the number of phishing sites is often much higher in number compared to that of legitimate sites as URLs based website phishing attacks are prevalent. Several studies, on the other hand, also use balanced datasets to avoid dataset bias. In this section, we will explore various data sources that are available for both phishing and legitimate websites.

5.1 Data Sources for Phishing URLs

Phishing data sources are collections of URLs that have been reported as phishing sites. They are maintained by organizations and companies to protect users from phishing attacks. The data sources are used by anti-virus software, browser extensions, and other security tools to identify and block phishing websites. Data from these sources can be also used to train a machine-learning model to detect new samples of phishing websites.

Some of the key features of phishing databases include user submissions which allow users to submit URLs that they believe are phishing sites, a verification process, to ensure that they are indeed phishing, integration with security tools, and regular updates. Two main data sources of phishing websites are PhishTank.com and OpenPhish.com.

PhishTank.com is a community-based repository where contributors work to sanitize data and information pertaining to online phishing. The data is available in CSV or XML formats. In addition, an Application programming interface (API) is also available for research purposes [44].

OpenPhish.com is a live repository of phishing URLs and is a fully automated self-contained phishing intelligence platform. OpenPhish obtains its data from a variety of sources, including security researchers, government agencies, and other organizations. The website uses a combination of automated and manual verification methods to ensure that the URLs in its database are indeed phishing sites. In addition to providing a database of phishing URLs, OpenPhish also offers a variety of services, including integration with security tools, such as anti-virus software and browser extensions, to provide real-time protection against phishing attacks [45].

While these two websites are designed to provide datasets of reported phishing websites, there are differences in their behavior. OpenPhish obtains its data from a variety of sources, including security researchers, government agencies, and other organizations. PhishTank, on the other hand, relies mainly on user submissions to build its database. Both websites use a verification process to ensure that the URLs in their databases are indeed phishing sites. However, the specifics of the verification process can vary between the two websites. For example, PhishTank uses a community-based verification process where users can vote on whether a reported URL is a phishing site, while OpenPhish uses a combination of automated and manual verification methods. OpenPhish and PhishTank both can be integrated with security tools, such as anti-virus software and browser extensions, to provide real-time protection against phishing attacks. However, the specifics of the integration process can vary between the two websites.

Besides the well-known data sources indicated above, researchers also use websites like **MalwareUrl** [46], **MalwareDomain** [47], and **MalwareDomainList** [48] to collect malicious URLs. These community-driven tools collaborate to combat cyber threats.

5.2 Data Sources for Legitimate URLs

Researchers have used different resources to collect a set of legitimate URLs including compiling a list of popular websites, using web crawling sources, and online directories.

Common Crawl is a large-scale web crawl that covers billions of web pages. The corpus is made up of petabytes of data that have been collected since 2008. It includes raw web page data, extracted metadata, and text extractions. This repository's material is maintained in Web ARChive ($WARC$) format, which contains URL-related data. Common Crawl makes this corpus accessible for collaborative research and analysis [49].

DMOZ.org, also known as the Open Directory Project, was a large, open directory of the web, organized into a hierarchy of categories. It was created with the goal of organizing a vast amount of information on the web and making it more easily accessible to users. The DMOZ directory was created and maintained by a volunteer editor community, who reviewed and categorized websites into various topics, such as arts, business, health, sports, and many more. The directory was widely used by search engines, such as Google, as a source of information to help improve the relevancy of search results. DMOZ was one of the largest and most comprehensive directories on the web, with millions of websites listed and organized into thousands of categories. However, the project was discontinued in 2017 due to a decline in editor participation and the increasing dominance of search engines, such as Google, in providing web search and navigation services. Despite its discontinuation, the legacy of the DMOZ project continues to influence the development of the web and the organization of online information. The directory remains a valuable resource for researchers, web developers, and users looking to explore the web and find information on specific topics [50].

In addition to the widely used services of Common Crawl and DMOZ, other services have been used by other researchers. **Yandex.XML** as a search engine provides API to submit queries and receive answers in XML format [51]. Additionally, authentic URLs can be collected using the archive form provided by **Alexa Web Crawl**. Alexa Internet began contributing its crawl data to the Internet Archive in 1996 [52]. Finally, **Link Klipper** is a Google Chrome browse add-on that enables users to export all of the links from a webpage into a CSV file [53].

5.3 Datasets for Phishing Detection

In addition to data sources of phishing and legitimate URLs, there existing ready-to-use datasets. **ISCXURL2016** [54] is a dataset that includes both authentic and phishing URLs. There are 35,300 benign URLs in this dataset that was gathered from the top Alexa websites using the Heritrix web crawler. This dataset also contains 12,000 URLs from the WEBSPAM-UK2007 dataset, 10,000 URLs from OpenPhish, 11,500 URLs from DNS-BH, and 45,450 URLs from Defacement URLs; a total of more than 78,000 URLs.

MillerSmiles Archives is a collection of phishing emails compiled by security researcher Paul Miller. It contains a large number of phishing emails organized by types, such as banking or payment phishing. It was a valuable resource for security researchers for building and testing phishing detection algorithms and provided insight into the tactics, techniques, and procedures used by phishers. The archives have not been updated since 2013 and the domain name millersmiles.co.uk is inactive. Despite that, it is still valuable for those studying phishing history and algorithms. Phishing URLs are included in these emails and can be obtained from the archives [55].

Other researchers also collected phishing and legitimate URLs to compose ML datasets. **Phishstorm** is a dataset that contains both legitimate and phishing URLs. 48,009 legitimate URLs and 48,009 phishing URLs are included in this dataset's total of 96,018 URLs [56].

Ebbu2017 dataset comprises 36,400 valid URLs and 37,175 phishing URLs. The legitimate URLs were collected from Yandex.XML and the phishing data was collected from PhishTank [57].

UCI-15 dataset defined 30 different attributes for phishing URLs and extracted values of those attributes for each phishing URL. Data were collected mainly from PhishTank, MillerSmiles, and from Google search operator and the total number of instances in this dataset is 2456 [58].

The dataset **UCI-16**, which contains 1353 examples of both legitimate and phishing URLs, is also used by researchers. It comprises 10 distinct features. Phishing URL data are gathered from PhishTank and legitimate URLs as collected from Yahoo and using a crawler [59].

Finally, **MDP-2018** dataset, which was downloaded between January and May 2015 and May and June 2017, has 48 features that were taken from 5000 legitimate URLs and 5000 phishing URLs. This dataset includes details on both legal and fraudulent URLs. Sources of fraudulent websites include PhishTank, OpenPhish, and legitimate websites like Alexa and Common Crawl [60].

Table 5.1 provides a detailed overview of the datasets, including the sources, number of phishing and legitimate URL data, and the total number of legitimate and phishing samples used.

Table 5.1: Dataset sources and the size of the data used for experiments in the literature

Ref	Dataset				
	Dataset source		**Dataset size**		**Total samples**
	Legitimate	**Phishing**	**Legitimate**	**Phishing**	
[18]	Common Crawl	PhishTank	800k	759k	1,500k
[1]	DMOZ	PhishTank	55k	55k	100k
[24]	DMOZ	PhishTank	100k	15k	115k
[3]	Alexa	PhishTank	110k	32k	142k
[27]	Yahoo directory, DMOZ	PhishTank	2k	32k	34k
[25]	Google Search Operator	PhishTank, MillerSmiles	6k	6.8k	12.8k
[26]	Yandex.XML	PhishTank	36k	37k	73k
[11]	Kaggle [61]	PhishTank	40k	60k	100k
[2]	DMOZ	PhishTank, MillerSmiles	54k	52.8k	106.8k
[20]	DMOZ, Alexa, Phish-storm	PhishTank, OpenPhish, Phish-storm	96k	96k	192k
[12]	DMOZ	PhishTank	4k	4k	8k
[13]	Alexa	PhishTank	7k	6k	13k
[19]	Common Crawl	PhishTank	10.6k	10.6k	21.2k
[28]	Alexa, DOMZ	PhishTank, OpenPhish, MalwareURL, MalwareDomain, MalwareDomainList	79k	62k	141k
[21]	Alexa, Yandex, Common Crawl	PhishTank, OpenPhish, MalwareDomain	278k	278k	556k
[22]	PhishTank, MillerSmiles, OpenPhish	PhishTank, MillerSmiles, OpenPhish	10.4k	11.9k	22.3k
[8]	Alexa	PhishTank	343k	70k	413k
[23]	Alexa	PhishTank	245k	245k	490k
[17]	Common Crawl	PhishTank	800k	759k	1559k
[16]	Common Crawl	PhishTank	1140k	1167k	2307k
[9]	Common Crawl	PhishTank	2220k	2353k	4573k
[14]	Alexa	PhishTank	85k	60k	145k
[7]	Alexa	PhishTank	10k	9.7k	19.7k
[10]	Kaggle (Source not mentioned)	Kaggle (Source not mentioned)	-	-	11k
[6]	Custom Crawler developed	PhishTank, OpenPhish	400k	400k	800k
[15]	Alexa, Common Crawl	PhishTank, OpenPhish	25.96k	25.96k	51.9k

Chapter 6

Experiments

In this section, we looked at performance metrics for identifying and evaluating the best-performing algorithms that we have seen from our review, details are provided in Table 6.1. In addition, we conducted original experiments using commonly used algorithms, which are listed in Table 4.1. In the following section, several detection strategies will be discussed and the metrics utilized will be briefly explained, along with corresponding equations.

6.1 Evaluation Metrics

We use N to represent the number of legitimate/phishing websites, with P denoting phishing and L denoting legitimate. The following metrics are used to assess the performance of the algorithms.

Precision is the proportion of phishing attacks $(N_{P\rightarrow P})$ classified correctly as phishing attacks to the total number of attacks detected $(N_{L\rightarrow P} + N_{P\rightarrow P})$.

$$Precision = \frac{N_P \rightarrow P}{N_{L\rightarrow P} + N_P \rightarrow P} \tag{6.1}$$

Recall is the proportion of phishing attacks $(N_{P\rightarrow P})$ classified correctly to total phishing attacks $(N_{P\rightarrow P} + N_{P\rightarrow L})$.

$$Recall = \frac{N_P \rightarrow P}{N_{P\rightarrow P} + N_P \rightarrow L} \tag{6.2}$$

Accuracy is the proportion of phishing and legitimate sites that have been correctly classified $(N_{L\rightarrow L} + N_{P\rightarrow P})$ to the total number of sites $(N_{L\rightarrow L} + N_{L\rightarrow P} + N_{P\rightarrow P} + N_{P\rightarrow L})$.

$$Accuracy = \frac{N_{L\rightarrow L} + N_{P\rightarrow P}}{N_{L\rightarrow L} + N_{L\rightarrow P} + N_{P\rightarrow P} + N_{P\rightarrow L}} \tag{6.3}$$

F1-Score is a widely used evaluation metric that combines the model's recall and precision into a single score for binary classification models.

$$F1 - score = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{6.4}$$

Table 6.1: Performance evaluation by researchers with metrics: [Acc]uracy, [P]recision, [Rec]all, [F1]-Score. Studies [1–3] used other metrics.

Ref	**Best Performing Algorithm**	**P**	**Rec**	**Acc**	**F1**
[14]	DT			97.40	96.30
[22]	Gradient Tree Boosting (GTB)			97.42	
[10]	eXtreme Gradient Boosting ($XGBoost$)	95.78	96.77	96.71	96.27
[24]	RF	94.00	94.00	94.05	93.20
[26]	RF	97.00		97.98	
[11]	RF	97.40		99.29	98.22
[27]	SVM			91.28	
[19]	CNN	99.57	100.00	99.80	99.78
[7]	CNN	99.00	99.20	99.20	99.20
[16]	CNN	96.53	95.09	95.78	95.81
[23]	CNN	97.33	93.78	95.60	95.52
[8]	CNN			98.30	94.95
[21]	CNN	92.35	98.09	99.02	95.13
[28]	$LSTM$	99.88	99.82	99.97	99.85
[9]	GRU	98.00		97.56	
[18]	$BiGRU$	99.40	99.50	99.50	99.40
[17]	$BiGRU$	99.64	99.43	95.55	99.54
[15]	Transformer			96+	
[13]	LURL			97.40	
[20]	EXPOSE			97+	
[6]	GramBedding	97.59	98.26	98.27	99.73
[25]	Adaptive Neuro-Fuzzy Inference System ($ANFIS$)			98.30	
[12]	Multi-Modal Hierarchical Attention Model ($MMHAM$)	97.84	96.66	97.26	97.24

6.2 Experimental Setup and Results

We want to evaluate both machine learning and deep learning algorithms' performance on a dataset in order to gain further insight into the conclusions made in the literature. We evaluated commonly used algorithms, as listed in Table 4.1, to achieve this.

We utilized a dataset that included 34 URL-based features and contained both benign and phishing URLs. The distribution of this dataset is relatively balanced, with 4,998 cases categorized as benign and 4,934 as phishing. To evaluate the models, we segregated the data in our experimental setup, putting 70% of it into the training set and 20% aside for the test set and the remaining 10% as our validation set. The Table 6.2 provides a snapshot of the label distribution.

Table 6.2: Label distribution for experiment

	Phishing	**Legitimate**	**Total**
Train	3552	3598	**7150**
Validation	395	400	**795**
Test	987	1000	**1987**

We performed each of our experiments ten times with varied splits of train, validation, and test data, and then averaged the results to acquire our final results from the experiments. The results of our experiments are shown in Table 6.3. Our goal in this study was to evaluate the effectiveness of deep learning and machine learning models. Additionally, we tried to find out if unidirectional and bidirectional deep learning models performed differently.

Table 6.3: Performance evaluation with metrics: [Acc]uracy, [P]recision, [Rec]all, [F1]-Score

No.	**Algorithm**	**P**	**Rec**	**Acc**	**F1**
1	RF	94.80	94.79	94.79	94.79
2	DT	93.22	93.16	93.16	93.16
3	$AdaBoost$	93.09	93.12	93.12	93.12
4	GB	93.34	93.32	93.32	93.32
5	SVM	86.86	85.40	85.40	85.03
6	NB	75.49	51.63	51.63	37.16
7	LR	92.61	92.59	92.59	92.59
8	$K-NN$	92.33	92.32	92.33	92.33
9	NN	98.14	98.13	98.13	98.13
10	CNN	99.48	99.48	99.48	99.48
11	RNN	99.10	99.09	99.09	99.09
12	$LSTM$	99.12	99.10	99.10	99.10
13	$BiLSTM$	98.82	98.79	98.79	98.79
14	GRU	99.34	99.33	99.33	99.33
15	$BiGRU$	99.22	99.21	99.21	99.21

The results of our studies reveal that RF outperforms machine learning-based models in terms of precision, recall, accuracy, and F1-score, with values of 94.80$\%$, 94.79$\%$, 94.79$\%$, and 94.79$\%$, respectively. For machine learning algorithms, we observed that NB provided the lowest accuracy during our experiments, and we believe this is because this algorithm was unable to obtain effective features for matching the patterns with the URLs, which is consistent with what we observed in the literature. For deep learning-based algorithms, we have seen that all of the models perform better than machine learning algorithms, and CNN surpasses others in terms of precision, recall, accuracy, and F1-score with values of 99.48$\%$, 99.48$\%$, 99.48$\%$, and 99.48$\%$, respectively. As a result, deep learning-based models are desirable options for constructing URL-based phishing detectors. In addition, we compared the performance of unidirectional and bidirectional models. Our findings suggest that uni-directional models outperform bi-directional ones. This is consistent with the findings of our survey; in the literature, we have shown that $LSTM$, a uni-directional model, produces better results than bi-directional models.

Chapter 7

Survey Findings

Several observations were made about the automated URL-based website phishing detection strategies employing machine learning algorithms while this study was being conducted. These observations are contained in the sections that follow for the features, algorithms, and dataset.

7.1 Feature

The feature selection process, which can make or break the detector, has a significant impact on the performance of an automated website phishing detector. The specific features must be chosen before the classification process can begin for both machine learning and deep learning approaches. However, if a deep learning-based approach is used, the feature extraction process can be done automatically because these algorithms are capable of identifying the key characteristics on their own; as a result, deep learning features can also be used if researchers are attempting to come up with new sets of features. For a URL-based website phishing attack detector to operate well, a combination of features directly connected to the URL is required. For instance, combining Domain Name System (DNS), domain, and lexical elements of the URL will improve the detector's accuracy. There is one thing to keep in mind, though, and that is to avoid using too many features for classification as this could lead to bias and over-fitting, both of which would impair the detector's effectiveness.

7.2 Algorithm

While conducting this survey, it was discovered that researchers were using a variety of algorithms from the fields of machine learning and deep learning to combat the problem of phishing. Researchers initially employed heuristic-based approaches to tackle these issues, but as machine learning models advanced, this strategy was swiftly supplanted. The manual feature extraction was a vital component of the machine learning-based method because it influenced how well the algorithms worked. Deep learning-based approaches, however, are currently quite popular because the models can now automatically infer the semantics of the URL, eliminating the need for manual extraction. Although the essence of these works has been simplified, the underlying architecture is still a conundrum. As a result of this survey, we can see that developing a URL-based detector using deep learning-based algorithms yields better results. Additionally, someone who has little prior domain expertise about what features to choose for categorization purposes may benefit from a deep learning method because this can be done automatically.

Based on the classification accuracy of this algorithm in this domain that we have observed from the literature, it can be suggested that RF algorithms in the area of machine learning perform the best with an accuracy of 99.29% with DT being another good machine learning algorithm that comes in second place with an accuracy of 97.40%. $LSTM$ is an algorithm that is the best choice (accuracy 99.96%) in the field of deep learning additionally CNN is the second-best-performing algorithm with accuracy of 99.79% for the deep learning-based approach. Someone who has little prior domain expertise about what features to choose for categorization purposes may benefit from a deep learning method because this can be done automatically.

To confirm our understanding, we conducted experiments and observed results that were consistent with past literature. RF succeeded effectively in the domain of machine learning algorithms. Nonetheless, when we used deep learning techniques, we got a different result. In this context, CNN outperformed other models on the dataset used in our study. Furthermore, both our survey findings and testing results confirm that deep learning models are the best choice for building URL-based phishing detection systems.

7.3 Dataset

The datasets utilized were not from a single source, and each researcher used a separate dataset to develop their system. As a result, the lack of a shared dataset can be a concern because one dataset may contain certain phishing site data while the other does not. Furthermore, because phishing URL databases are not open-source, many academics do not use them. This is advantageous because attackers may acquire publicly accessible datasets and use them to extract key attributes and tailor their assaults accordingly. The drawback of that is that it might be laborious and time-consuming for a researcher to create a dataset.

Chapter 8

Conclusion and Future Work

In this systematic survey, we discussed URL-based phishing detection approaches. We were particularly interested in the features, algorithms, and dataset for URL-based detection techniques in this work. Numerous components are working together to provide an effective detector for URL-based website phishing attacks and this effort is driven by the fact that phishing is a critical cyber security concern that requires a faster and more effective solution to prevent phishers from stealing a user's sensitive information.

We aimed to assess the fairness of both machine learning and deep learning models in our study by doing tests on the same test set for all algorithms, which we had not seen in other literature's experiments, which are listed in Table 6.1. Furthermore, when we conducted our own experiments, we discovered that deep learning-based models outperform machine learning models significantly, as shown in Table 6.3 because deep learning models were able to capture complex patterns in the URLs and representations from the data without the need for extensive feature engineering. Additionally, we observed that the difference between the uni-directional and bi-directional models is not statistically significant in our experiments. However, bi-directional models are more complex and can perform better with fine-tuning.

It was observed that the lexical analyzers are effective tools for detecting URL-based phishing since they can detect phishing on the fly (real-time detection), and they can also correctly identify newly constructed phish. However, more effort needs to be put into making the detector more robust because attackers are always coming up with new ways to use phishing attacks to get past defenses already in place. One approach to do this is to use adversarial phishing samples to train the model, and these samples can be produced using an Generative Adversarial Network (GAN).

One significant problem that needs to be resolved right away was not included in the survey's scope, but it needs addressing nonetheless. Google Sites is increasingly used to create websites, and fraudsters use it to build phishing websites and conduct phishing attacks. The problem, in this case, is that because sites created with Google Sites disclose less information in the URL, the approaches covered in this survey may not be adequate to thwart phishing attempts made using Google Sites.

Furthermore, while doing our research, we discovered two major problems in the field of machine learning-based detectors. These difficulties stem from dataset imbalances and the high computing demands faced by complicated deep learning models. In light of these findings, we want to steer our future research efforts on few-shot learning models. These models have the advantage of being less computationally intensive because of their ability to operate well with less training data.

Bibliography

[1] Harshal Tupsamudre, Ajeet Kumar Singh, and Sachin Lodha. Everything is in the name–a url based approach for phishing detection. In *International symposium on cyber security cryptography and machine learning*, pages 231–248. Springer, 2019.

[2] Ahmed AlEroud and George Karabatis. Bypassing detection of url-based phishing attacks using generative adversarial deep neural networks. In *Proceedings of the Sixth International Workshop on Security and Privacy Analytics*, pages 53–60, 2020.

[3] Jehyun Lee, Pingxiao Ye, Ruofan Liu, Dinil Mon Divakaran, and Mun Choon Chan. Building robust phishing detection system: an empirical analysis. *NDSS MADWeb*, 2020.

[4] KnowBe4. History of phishing. https://www.phishing.org/history-of-phishing. (Accessed on 06/24/2022).

[5] APWG. Phishing activity trends report. https://apwg.org/trendsreports/, April 2021. (Accessed on 14/11/2021).

[6] Ahmet Selman Bozkir, Firat Coskun Dalgic, and Murat Aydos. Grambeddings: A new neural network for url based identification of phishing web pages through n-gram embeddings. *Computers & Security*, 124:102964, 2023.

[7] Zainab Alshingiti, Rabeah Alaqel, Jalal Al-Muhtadi, Qazi Emad Ul Haq, Kashif Saleem, and Muhammad Hamza Faheem. A deep learning-based phishing detection system using cnn, lstm, and lstm-cnn. *Electronics*, 12(1):232, 2023.

[8] Faan Zheng, Qiao Yan, Victor CM Leung, F Richard Yu, and Zhong Ming. Hdp-cnn: Highway deep pyramid convolution neural network combining word-level and character-level representations for phishing website detection. *Computers & Security*, 114:102584, 2022.

[9] Saad Al-Ahmadi, Afrah Alotaibi, and Omar Alsaleh. Pdgan: Phishing detection with generative adversarial networks. *IEEE Access*, 10:42459–42468, 2022.

[10] Mukta Mithra Raj and J Angel Arul Jothi. Website phishing detection using machine learning classification algorithms. In *Applied Informatics: 5th International Conference, ICAI 2022, Arequipa, Peru, October 27–29, 2022, Proceedings*, pages 219–233. Springer, 2022.

[11] Andrei Butnaru, Alexios Mylonas, and Nikolaos Pitropakis. Towards lightweight url-based phishing detection. *Future Internet*, 13(6):154, 2021.

[12] Yidong Chai, Yonghang Zhou, Weifeng Li, and Yuanchun Jiang. An explainable multi-modal hierarchical attention model for developing phishing threat intelligence. *IEEE Transactions on Dependable and Secure Computing*, 2021.

[13] Ashit Kumar Dutta. Detecting phishing websites using machine learning technique. *PloS one*, 16(10):e0258361, 2021.

[14] Qasem Abu Al-Haija and Ahmad Al Badawi. Url-based phishing websites detection via machine learning. In *2021 International Conference on Data Analytics for Business and Industry (ICDABI)*, pages 644–649. IEEE, 2021.

[15] Katherine Haynes, Hossein Shirazi, and Indrakshi Ray. Lightweight url-based phishing detection using natural language processing transformers for mobile devices. *Procedia Computer Science*, 191:127–134, 2021.

[16] Abdullah Al-Alyan and Saad Al-Ahmadi. Robust url phishing detection based on deep learning. *KSII Transactions on Internet and Information Systems (TIIS)*, 14(7):2752–2768, 2020.

[17] Lijuan Yuan, Zhiyong Zeng, Yikang Lu, Xiaofeng Ou, and Tao Feng. A character-level bigru-attention for phishing classification. In *Information and Communications Security: 21st International Conference, ICICS 2019, Beijing, China, December 15–17, 2019, Revised Selected Papers 21*, pages 746–762. Springer, 2020.

[18] Tao Feng and Chuan Yue. Visualizing and interpreting rnn models in url-based phishing detection. In *Proceedings of the 25th ACM Symposium on Access Control Models and Technologies*, pages 13–24, 2020.

[19] Wei Wei, Qiao Ke, Jakub Nowak, Marcin Korytkowski, Rafał Scherer, and Marcin Woźniak. Accurate and fast url phishing detector: a convolutional neural network approach. *Computer Networks*, 178:107275, 2020.

[20] Bushra Sabir, M Ali Babar, and Raj Gaire. An evasion attack against ml-based phishing url detectors. *arXiv preprint arXiv:2005.08454*, 2020.

[21] Ali Aljofey, Qingshan Jiang, Qiang Qu, Mingqing Huang, and Jean-Pierre Niyigena. An effective phishing detection model based on character level convolutional neural network from url. *Electronics*, 9(9):1514, 2020.

[22] Paulius Vaitkevicius and Virginijus Marcinkevicius. Comparison of classification algorithms for detection of phishing websites. *Informatica*, 31(1):143–160, 2020.

[23] Weiping Wang, Feng Zhang, Xi Luo, and Shigeng Zhang. Pdrcnn: Precise phishing detection with recurrent convolutional neural networks. *Security and Communication Networks*, 2019:1–15, 2019.

[24] Eint Sandi Aung and Hayato Yamana. Url-based phishing detection using the entropy of non-alphanumeric characters. In *Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services*, pages 385–392, 2019.

[25] Moruf A Adebowale, Khin T Lwin, Erika Sanchez, and M Alamgir Hossain. Intelligent web-phishing detection and protection scheme using integrated features of images, frames and text. *Expert Systems with Applications*, 115:300–313, 2019.

[26] Ozgur Koray Sahingoz, Ebubekir Buber, Onder Demir, and Banu Diri. Machine learning based phishing detection from urls. *Expert Systems with Applications*, 117:345–357, 2019.

[27] Ankit Kumar Jain and BB Gupta. Phish-safe: Url features-based phishing detection system using machine learning. In *Cyber Security*, pages 467–474. Springer, 2018.

[28] R Vinayakumar, KP Soman, and Prabaharan Poornachandran. Evaluating deep learning approaches to characterize and classify malicious url's. *Journal of Intelligent & Fuzzy Systems*, 34(3):1333–1343, 2018.

[29] Rana Alabdan. Phishing attacks survey: types, vectors, and technical approaches. *Future Internet*, 12(10):168, 2020.

[30] Edwin D Frauenstein and Stephen V Flowerday. Social network phishing: Becoming habituated to clicks and ignorant to threats? In *2016 Information Security for South Africa (ISSA)*, pages 98–105. IEEE, 2016.

[31] Deanna D. Caputo, Shari Lawrence Pfleeger, Jesse D. Freeman, and M. Eric Johnson. Going spear phishing: Exploring embedded training and awareness. *IEEE Security Privacy*, 12(1):28–38, 2014.

[32] Jason Hong. The state of phishing attacks. 55(1), 2012.

[33] Tenzin Dakpa and Peter Augustine. Study of phishing attacks and preventions. *International Journal of Computer Applications*, 163:5–8, 04 2017.

[34] Akarshita Shankar, Ramesh Shetty, and B Nath. A review on phishing attacks. *International Journal of Applied Engineering Research*, 14(9):2171–2175, 2019.

[35] Louise O'Hagan. Angler phishing: Criminality in social media. In *5th European Conference on Social Media ECSM 2018*, page 190, 2018.

[36] IBM and Ponemon Institute. Cost of a data breach report 2023. https://www.ibm.com/downloads/cas/E3G5JMBP. (Accessed on 9/9/23).

[37] APWG. Phishing activity trends report. https://docs.apwg.org/reports/apwg_trends_report_q4_2022.pdf?_gl=1*1pdz40k*_ga*MzI5NTU2NDUwLjE2ODUzMTQ3MTM.*_ga_55RF0RHXSR*MTY5NDY2ODAwMC43LjAuMTY5NDY2ODAwMC4wLjAuMA..&

_ga=2.46847847.1333104756.1694668004-329556450.1685314713, May 2022. (Accessed on 9/9/2023).

[38] Charles Griffiths. The latest 2023 phishing statistics. https://aag-it.com/the-latest-phishing-statistics/, September 2023. (Accessed on 09/09/2023).

[39] knowbe4. Report: 2023 phishing by industry benchmarking. https://info.knowbe4.com/phishing-by-industry-benchmarking-report. (Accessed on 09/09/2023).

[40] Tim Youm. The digital economy growth, trends in phishing attacks and the industries commonly targeted. https://www.linkedin.com/pulse/digital-economy-growth-trends-phishing-attacks-industries-tim-youm/, July 2021. (Accessed on 09/09/2023).

[41] FBI Springfield. Internet crime complaint center releases 2022 statistics. https://www.fbi.gov/contact-us/field-offices/springfield/news/internet-crime-complaint-center-releases-2022-statistics, March 22. (Accessed on 5/27/23).

[42] Cassie Bottorff Kelly Main. Phishing statistics by state in 2023. https://www.forbes.com/advisor/business/phishing-statistics/, June 2023. (Accessed on 09/09/2023).

[43] CYBSAFE. The ripple effect: How one phishing attack can cause disaster across your organization. https://www.cybsafe.com/blog/how-can-phishing-affect-a-business/, July 2023. (Accessed on 09/09/2023).

[44] PhishTank. Join the fight against phishing. https://phishtank.com/.

[45] OpenPhish. Phishing intelligence. https://openphish.com/.

[46] MalwareURL. Fighting malware and cyber criminality. http://www.malwareurl.com/. (Accessed on 03/04/2023).

[47] RiskAnalytics. Not all threat intel is created equal. https://riskanalytics.com//. (Accessed on 03/04/2023).

[48] Malware Domain List. Malware domain list. https://www.malwaredomainlist.com/. (Accessed on 03/04/2023).

[49] Common crawl. Common crawl. https://commoncrawl.org/.

[50] Curlie. https://curlie.org/.

[51] Yandex. Yandex. https://yandex.com/dev/.

[52] ARossi. Alexa crawls. https://archive.org/details/alexacrawls?tab=about.

[53] Codebox. Link klipper - extract all links. https://chrome.google.com/webstore/detail/link-klipper-extract-all/fahollcgofmpnehocdgofnhkkchiekoo?hl=en.

[54] UNB. https://www.unb.ca/cic/datasets/url-2016.html.

[55] MillerSmiles.co.uk. Phishing scams and spoof emails at millersmiles.co.uk. http://www.millersmiles.co.uk/.

[56] Sameul Marchal. Phishstorm - phishing / legitimate url dataset. https://research.aalto.fi/fi/datasets/phishstorm-phishing-legitimate-url-dataset, 2014.

[57] Ebubekirbbr. Pdd/input at master · ebubekirbbr/pdd. https://github.com/ebubekirbbr/pdd/tree/master/input, 2019.

[58] Rami Mustafa A Mohammad. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/phishing+websites, 2015.

[59] Neda Abdelhamid. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/Website+Phishing, 2016.

[60] Choon Lin Tan. Phishing Dataset for Machine Learning: Feature Evaluation. https://data.mendeley.com/datasets/h3cgnj8hft/1, 2018.

[61] Siddharth Kumar. Malicious and benign urls. https://www.kaggle.com/datasets/siddharthkumar25/malicious-and-benign-urls, May 2019.

Appendix A

Novelty, Approach and Limitation

Ref	Novelty	Approach	Limitation
[18]	Utilizing novel visualization strategies to depict the inside RNN process	Investigate Recurrent Neural Networks (RNNs) for phishing attack detection using only the lexical features of URLs. Four RNN models are trained in the paper using a dataset of 1.5 million URLs, and they successfully achieve high detection without the need for manual feature identification by experts.	In this study, the batch size and dropout rate were not taken into consideration. Furthermore, only a small number of factors were adjusted during the fine-tuning phase.
[1]	Enhancing the effectiveness of URL-based phishing attack detection methods by investigating alternate feature extraction methods	Using word segmentation, n-grams, and a list of well-known words that are highly suggestive of phishing attacks, create a phishing detection system that is highly indicative of phishing attacks	The suggested method can identify some sorts of URLs, such as brand in subdomain, brand in domain, and brand in path, however it cannot identify domains made up of unrelated or incorrectly composed domains
[24]	Suggests a brand-new feature for URL-based phishing detection termed the entropy of non-alphanumeric (NAN) characters. This feature takes into account how often NAN characters appear in URLs, which is thought to affect how well URL-based detection works.	The authors suggest that phishing detection be built around URLs. They contend that URL attributes, particularly the distribution of non-alphanumeric (NAN) characters, can have a significant impact on the effectiveness of URL-based detection	This method has a False Positive Rate (FPR) that is over 20%, which is quite high. As a result, a sizable percentage of trustworthy URLs might be mistakenly identified as phishing URLs, raising false alerts.

[3]	The methodology's peculiarity is that it produces numerous randomized models from a single feature vector and a predetermined fixed training set	Address the susceptibility of adversarial assaults on classifiers based on machine learning. The method tries to strengthen the detection system's resilience to such evasion attacks by adding controlled noises to the feature set during training	The effectiveness of the proposed approach against various adversarial attacks or how well it can withstand sophisticated evasion strategies are not specifically discussed in the paper
[27]	Provides a solution that makes use of the capabilities of machine learning algorithms to automatically understand patterns and characteristics of phishing URLs	The study leverages PHISH-SAFE, a machine learning-based anti-phishing system, which concentrates on using URL features. 14 features retrieved from the URL are employed by the system, which is trained using Support Vector Machine and Nave Bayes classifiers, to identify whether a website is a phishing or non-phishing site	To increase the accuracy of the suggested phishing detection system, extra features can be included. Other machine-learning methods can also be applied to boost the efficiency of the suggested system
[25]	The first study to take into account the most effective text, image, and frame feature-based phishing detection technique	The methodology used entails creating a strong phishing detection and protection system using an Adaptive Neuro-Fuzzy Inference System (ANFIS) algorithm and incorporating features from text, graphics, and frames of real and fraudulent websites	The proposed approach's possible implementation and deployment issues, such as computing complexity, scalability, and the practical viability of incorporating it into a real-world web browser plug-in for real-time phishing detection, are not discussed in detail in the paper

[26]	Proposed a real-time anti-phishing system that makes use of seven different categorization methods and features based on natural language processing (NLP). The system is not dependent on outside services because it can recognize new web pages and is language agnostic	Seven distinct machine learning algorithms were used to implement a phishing detection system. Another feature set included word vectors, hybrid features, and features based on natural language processing (NLP)	The entire dataset was not trained for the experiment due to a training time limitation
[11]	The authors provide a novel method for phishing identification that only makes use of features taken directly from the URL. As opposed to conventional methods, which rely on other qualities like content or behavior, which might not always be dependable or accessible	The authors train their phishing detection model using supervised machine learning methods. In this method, labeled data is used to train a model that can distinguish between authentic and phishing URLs using known patterns	Although the use of feature extraction from URLs for phishing detection is mentioned in the study, neither the investigation of unique features nor their impact on the performance of the model are specifically mentioned

[2]	The work's original idea is to employ Generative Adversarial Networks (GANs) to create URL-based phishing samples that can deceive both basic and complex machine learning-based phishing detection techniques, including black box models	Developed an evasion attack against the MLPU system's characteristics that are trained on distance. Using the Generative Adversarial Network (GAN), they produced malicious URLs. Binarized URL feature vectors were provided as input to the generator. Using constant feedback from the discriminator, it eventually learned how to produce adversarial examples	The research focuses on Generative Adversarial Networks-based evasion attacks. (GANs). The proposed method has not, however, been put to the test against graph-based phishing detection methods
[20]	Through an evasion attack, the authors discover security flaws in the MLPU systems under consideration	Presented the systems with a foundation for an evasive attack.They mimicked an attack on 41 MLPU systems by reproducing them	Confined the testing to standard machine learning and deep neural network models exclusively
[12]	In order to create useful phishing website security operations centers, a multi multi-modal hierarchical attention model (MMHAM) was proposed	To identify fraud indications from the three main modalities of website material—URLs, textual content, and visual design— a multi-modal hierarchical attention model (MMHAM) was proposed that uses deep representation-based techniques	The study did not examine the phishing websites' semantic linguistic trends. This suggests that the study might not have adequately captured the subtleties and underlying trends in the textual content of phishing websites, which may have an impact on how thorough the phishing detection method is proposed

[13]	In order to identify phishing URLs, this work proposes a machine learning-based solution employing a recurrent neural network	The suggested method relies on RNN that accepts URL as input. The author evaluates the suggested technique using a dataset of 7900 malicious and 5800 legitimate sites	One of the study's shortcomings is that it solely concentrated on categorizing websites into predetermined groups based on predetermined features, which might not always adequately reflect the constantly growing phishing attacks
[19]	According to the paper, the suggested method is suited for mobile devices, making it possible to use it on those devices without noticeably degrading performance	The paper proposes a technique that just considers the text in URLs. Due to the fact that it does not involve additional processing of traffic statistics or web content, this approach enables quicker analysis and identification of harmful URLs. The method uses CNNs to accurately and efficiently analyze the URL text and identify dangerous URLs	One-hot character-level vectors are used as CNN inputs, and the method exclusively uses the text in the URL for analysis. The variety of properties that can be derived from the URL text may be constrained as a result, and significant information that may be contained in other URL components like the domain name structure or query parameters may be lost. As a result, the identification of phishing websites could produce false positives or false negatives

[28]	The strategy used in this paper is distinct from conventional techniques that rely on word- or token-level modeling and model URLs at the character level. Character-level modeling is more resilient to variations and obfuscations in harmful URLs and enables for the capture of fine-grained patterns in URLs	Recurrent neural networks (RNN), identity-recurrent neural networks (I-RNN), long short-term memory (LSTM), convolutional neural networks (CNN), and convolutional neural network-long short-term memory (CNN-LSTM) architectures are among the deep learning architectures that are evaluated in this paper for the detection of malicious URLs. This thorough assessment aids in determining the right architecture for the job	The lack of a thorough investigation of how deep learning techniques function in real-time circumstances is one of the paper's limitations
[15]	In order to combat the changing threat scenario of a phishing attack and get around the computational cost associated with deep learning-based algorithms, pre-trained transformers utilized for URL-based phishing detection on mobile devices	As a starting point, 15 ANN models were created by applying Artificial Neural Networks (ANNs) to URL-based and HTML-based website properties. With the maximum accuracy of 86.2%, it was found that utilizing deep ANNs on URL-based characteristics alone performs badly. This shows that even with deep ANNs, URL-based features alone are insufficient for effective phishing detection.	Despite their potential, BERT and ELECTRA now perform somewhat worse than some of the best models, according to the study

[6]	The method introduces the use of N-gram embeddings that are instantly calculated without the need for any pre-training phase	GramBeddings, the proposed method, introduces new improvements like on-the-fly computation of N-gram embeddings, removal of word/subword level information, smart N-gram selection with attention mechanism, a publicly available dataset, real-time inference, language-agnostic prediction, and elimination of third-party services/hand-crafted features for phishing detection	The available datasets lack diversity in terms of domain and TLD, data leakage, and repetitive domain names
[10]	Provide a workable method based on URL characteristics for identifying rogue websites, which can greatly reduce the likelihood of phishing attempts	Investigating the use of eight current machine learning classification techniques for phishing detection, including Extreme Gradient Boosting (XGBoost), Random Forest (RF), Adaboost, Decision Trees (DT), K-nearest neighbours (KNN), Support Vector Machines (SVM), Logistic Regression, and Nave Bayes (NB)	The study assesses the suggested strategy using a particular dataset; it makes no indication of whether it has also been tested using a variety of datasets from various sources.

[7]	The progressive aspects of the work are the deep learning-based strategies for phishing website detection that are suggested. Long short-term memory (LSTM) and convolutional neural network (CNN) models, as well as a combination of LSTM-CNN techniques, are specifically used.	By training on large datasets that include examples of both legitimate and phishing websites, the suggested techniques take advantage of deep learning to assess and identify phishing websites	Deep learning methods can demand a lot of processing power and training time, therefore, the suggested system's scalability and usefulness may be hampered by the lengthy training procedure, particularly in real-time or resource-constrained applications
[14]	This particular use of neural networks and decision trees have special benefits in terms of accuracy and interpretability	The suggested method entails examining website URLs, utilizing machine learning methods like decision trees and neural networks, assessing performance using classification accuracy, and using a recent phishing dataset for evaluation	The offered methods may work well, but they may not fully leverage into the potential of more sophisticated deep learning models or other sophisticated machine learning methods that might raise the system's accuracy
[9]	This method avoids the need to crawl the website or rely on third-party services, in contrast to standard phishing detection strategies that rely on webpage content elements. Potentially overcoming some of the drawbacks of content-based techniques, this innovative emphasis on URL-based identification can deliver more accurate and dependable findings	In this paper, the Long Short-Term Memory (LSTM) network serves as the generator in this paper's GAN architecture, while the Convolutional Neural Network (CNN) serves as the discriminator. The CNN uses the synthetic phishing URLs created by the LSTM to identify whether they are phishing or not	The paper's main drawback is that it doesn't fully analyze how complex the PDGAN model is in comparison to other models

[16]	The suggested technique relies exclusively on the URL and does not take into account fixed elements like URL length. This method is appealing because it is simple to implement in constrained systems like firewalls, which might not have access to other contextual data.	In order to do the URL identification task, the researchers employed a CNN model which receives the URL as input rather than relying on parameters like URL length	Data from a specified time period (2006–2018) were utilized for training and testing the suggested solution, therefore they might not accurately reflect the state of phishing attacks now. Selection bias may potentially affect the data that was gathered
[17]	The proposed model, the BiGRU-Attention model, which combines bi-directional gated recurrent units (BiGRU) with an attention mechanism for online phishing detection, is what makes the study innovative	In the proposed work, a better model for online phishing detection will be developed using a bi-directional gated recurrent unit (BiGRU) with an attention mechanism. In order to determine a score utilizing the attention mechanism, the model takes into account the characters in the URL that come before and after a specific character.	Cross-validation is absent from the experiment, which can lead to overfitting.

[23]	The novelty of the paper lies in the proposed approach called PDRCNN (Phishing Website Detection using URL-based CNN). PDRCNN encodes the information of a URL into a two-dimensional tensor and feeds it into a novel deep learning neural network for classification, unlike existing anti-phishing approaches that call for crawling webpage content or utilizing third-party services	The authors first extract global characteristics from the created tensor using a bidirectional LSTM (Long Short-Term Memory) network, providing all string information to each character in the URL. The relevant elements of the URL are then captured, the retrieved features are compressed into a fixed-length vector space, and a CNN (Convolutional Neural Network) is utilized to automatically determine which characters play important roles in phishing identification	PDRCNN does not take into account if the URL-corresponding website is active or whether there is a problem.
[8]	The novel HDP-CNN that the authors suggest combines character-level and word-level embedding data from the URL string sequences.	The method starts with the input of URL string sequences and conducts character-level and word-level embedding independently. After that, it connects the character-level and word-level embedding representations of the URL using the Highway network and extracts local features of various sizes from the region embedding layer. The deep pyramid structure network is then passed over to them to capture the global representation of the URL	To evaluate the method's robustness, additional study with balanced datasets or real-world data is required

[22]	A thorough comparison offers insightful information about how various algorithms function in the context of phishing detection, which can help with understanding the advantages and disadvantages of various strategies	For their comparison, the authors chose eight popular classification algorithms. The Scikit Learn module, a well-known machine learning package, was used to configure these algorithms in Python. These algorithms were chosen because they are often used in the literature and have the ability to identify phishing attempts	The authors noted that manual expert review was used to adjust hyper-parameters. Given that the choice of hyper-parameters can change based on the dataset and issue domain, this could induce bias or limits in the selection of the best hyper-parameter values
[21]	The suggested model does not involve retrieval of the target website content or reliance on external services, in contrast to existing anti-phishing strategies that do. This makes it a quick and self-contained solution for phishing detection	Convolutional neural networks (CNN) at the character level are used in the proposed model to detect phishing attacks based on the website's URL. It is a self-contained and effective method for detecting phishing since it eliminates the need to retrieve content from the target website or rely on outside services	The training period for the suggested model is stated as being fairly lengthy, which may be a possible disadvantage in terms of effectiveness and scalability