IEEE ComSoc IEEE Open Journal of the Communications Society

Received 17 August 2024; revised 4 September 2024; accepted 11 September 2024. Date of publication 17 September 2024;
date of current version 31 March 2025.

Digital Object Identifier 10.1109/OJCOMS.2024.3462503

Across the Spectrum In-Depth Review AI-Based Models for Phishing Detection

SHAKEEL AHMAD[1], MUHAMMAD ZAMAN[1] (Member, IEEE), AHMAD SAMI AL-SHAMAYLEH[2], RAHIEL AHMAD[1], SHAFI'I MUHAMMAD ABDULHAMID[3], ISMAIL ERGEN[4], AND ADNAN AKHUNZADA[5] (Senior Member, IEEE)

[1]Faculty of Computer Science, University of Lahore, Lahore 40100, Pakistan
[2]Department of Data Science and Artificial Intelligence, Faculty of Information Technology, Al-Ahliyya Amman University, Amman 19328, Jordan
[3]Department of Information Technology, Science and Technology Division, Community College of Qatar, Doha, Qatar
[4]Department of Fine Art, Design and Architecture, Faculty of Digital Game Design, Istinye University, 34396 Istanbul, Türkiye
[5]College of Computing and IT, Department of Data and Cybersecurity, University of Doha for Science and Technology, Doha, Qatar

CORRESPONDING AUTHORS: M. ZAMAN AND S. M. ABDULHAMID (e-mail: muhammad.zaman@cs.uol.edu.pk; shafii.abdulhamid@ccq.edu.qa)

ABSTRACT Advancement of the Internet has increased security risks associated with data protection and online shopping. Several techniques compromise Internet security, including hacking, SQL injection, phishing attacks, and DNS tunneling. Phishing attacks are particularly significant among Web phishing techniques. In a phishing attack, the attacker creates a fake website that closely resembles a legitimate one to deceive users into providing sensitive information. These attacks can be detected using both traditional and modern AI-based models. However, even with state-of-the-art methods, accurately classifying newly emerged links as phishing or legitimate remains a challenge. This study conducts a comparative analysis of more than 130 articles published between 2020 and 2024, identifying challenges and gaps in the literature and comparing the findings of various authors. The novelty of this research lies in providing a roadmap for researchers, practitioners, and cybersecurity experts to navigate the landscape of machine learning (ML) and deep learning (DL) models for phishing detection. The study reviews traditional phishing detection methods, ML and DL models, phishing datasets, and the step-by-step phishing process. It highlights limitations, research gaps, weaknesses, and potential improvements. Accuracy measures are used to compare model performance. In conclusion, this research provides a comprehensive survey of website phishing detection using AI models, offering a new roadmap for future studies.

INDEX TERMS Anomaly detection, blocklists, cyber-attack mitigation, cybersecurity, deep learning (DL), machine learning (ML), phishing detection, threat intelligence, Web phishing detection, whitelists.

I. INTRODUCTION

WEB PHISHING is a cyber attack method in which attackers disguise themselves as trustworthy entities to extract sensitive personal information from individuals, such as usernames, passwords, and credit card details [1]. This is typically done through emails, instant messaging, or Web pages that mimic real services. Generally, phishing aims to gain access to personal and financial information for fraudulent activities or sales on the dark Web [2]. Understanding the basics of Web phishing is crucial, as it forms the foundation for developing effective detection and prevention strategies [3].

Phishing attacks have evolved significantly over the years, becoming more sophisticated and complex to detect. Initially, phishing was relatively simple in design, involving general messages sent to multiple recipients at once, hoping that a small fraction would fall for it [4]. Such messages often included poor grammar and other obvious signs of fraud [5]. Today's phishing techniques are highly targeted and personalized, using even minute details about the victim obtained from social media or other sources [6]. This approach, known as spear phishing, increases the chances of success due to its more convincing deceit [7].

Detection of Web phishing involves identifying and mitigating malicious activities before they can spread [8]. Traditional detection mechanisms are highly dependent on blacklists, which are databases containing known phishing URLs that are blocked in Web browsers and security software [9]. The effectiveness of blacklists is often limited, as they can only protect against threats that have been identified in the past—making them essentially reactive. This limitation has spurred the development of more advanced detection techniques designed to track down newer and emerging Web phishing threats in real-time [10].

Machine learning has become a critical tool in the fight against phishing on the Web. By analyzing large volumes of data, machine learning algorithms can identify patterns and anomalies associated with phishing attacks [11]. These algorithms can learn characteristics related to URL structure, domain age, website content, and email metadata to detect phishing attempts with high precision [12]. Thus, using machine learning not only enhances detection capabilities but also reduces false positives, thereby avoiding the misidentification of legitimate websites and emails as malware [13]. However, traditional and AI models often fail to detect newly emerged phishing links. Therefore, systematic literature reviews are essential for researchers to identify study gaps, evaluate the performance of existing models, and discuss current datasets [14].

The significant contributions of this study include the examination of methods to prevent phishing attacks, covering attack types, phishing processes, user behaviors, and prevention measures. It discusses both traditional and modern detection methods, analyzes existing solutions, and addresses their limitations. The key contributions include:

1) Discussing all possible phishing attack modes, techniques, and the effects of attacks.
2) Exploring attack processes, typologies, and anti-phishing solutions.
3) Presenting a comparative analysis of traditional as well as machine learning (ML) and deep learning (DL) based models.
4) Providing taxonomic classification of anti-phishing techniques.
5) Measuring the performance of models using accuracy to evaluate significant models for phishing detection.
6) Finally, presenting a plethora of promising future research directions.

The remainder of this paper is organized as follows: Section II discusses the literature review. Section III covers the research methodology. Section IV discusses different phishing attack methods along with the step-by-step phishing attack process. Section V presents how phishing works and different phishing techniques. Section VI explains different phishing detection methods based on webpage screenshots, while Section VII describes the different phishing detection datasets and their comparative analysis. Similarly, Section VIII presents different anti-phishing methods with their comparative analysis. Section IX presents model recommendations for phishing detection, highlighting which model is the best fit for phishing detection. Section X highlights open challenges and discussions related to phishing detection and research papers. In the last section, the conclusion is presented.

II. LITERATURE REVIEW

Internet use has become integral to people's daily lives, making it difficult to envision a world without it. According to the Global Digital Population Survey Report (GDP) [15], published in 2023, approximately 5.3 billion people use the Internet worldwide. Of these, 62% use social media. In the report [16], it is stated that 94.6% of these users have accessed the Internet through smartphones. This connectivity has revolutionized life, including information exchange, online shopping, communication, and professional tasks. At the beginning of 2019, when the pandemic began, there were significant changes in traditional offline services. These services transformed from offline to online platforms, particularly in industries such as catering and retail.

In this digital era, individuals frequently share sensitive online data, such as login credentials, personal information, and credit card details. Unfortunately, cybercriminals exploit various illicit methods to acquire this information and subsequently engage in unauthorized activities on the Internet. Network security concerns have been present since the inception of the Internet, evolving in tandem with its development. In [17], the author proposed that the rapid evolution of network attack techniques poses significant challenges to cybersecurity. There are several categories of cybersecurity issues, classified based on attack methods and forms, including denial-of-service attacks (DoS), man-in-the-middle (MitM) attacks, SQL injection (SQL-Inj), zero-day (ZD) exploits, DNS tunneling, phishing, and malware. In [18], the author explained that the dynamic landscape of the Internet and its vulnerabilities necessitate ongoing efforts to enhance cybersecurity measures and protect users from potential threats.

In [19], the author explained that phishing attacks require tactful skills, including re-engineering, networking, coding, databases, and deep knowledge of protocols and how information is stolen from these protocols. In [20], the author explained how the attacker designs the phishing page linked to the database; the Web form looks like the original and shares a link with the user using social media, SMS Gateway, and email. This sharing contains alarming messages and warning text, including misleading images, to attract and induce the user to click the link. Phishing attacks have caused economic losses in the last 30 years. In [21], the author discusses the history of phishing, stating that phishing attacks increased dramatically during the 2019 pandemic. In that period, governments worldwide issued financial assistance to their citizens and started collecting sensitive data that contained bank accounts, credit card history, debit card history, and personal details to disburse funds.

Similarly, the attacker launched the same campaign to obtain data online from citizens. According to phishing attack statistics published in 2022, approximately 36% of data breaches are caused by phishing attacks, and 83% of citizens in the U.S. experience phishing attacks [3]. This ratio increased from 80% to 345% from 2020 to 2021. Another report published in 2022 by a U.S.-based organization [2] states that the number of phishing attacks doubled in 2022 compared to the 2019 pandemic due to the high success rate.

In [2], the author proposed several methods to prevent phishing attacks, including technical staff education and training on daily email responses, SMS, WhatsApp, and social media material sharing. In [22], it is represented that the objective is to survey recently published anti-phishing methods. Identifying a phishing website is a challenging task during the process of obtaining user information. Researchers have proposed several methods to identify phishing websites before the invention of artificial intelligence, including traditional methods such as whitelisting universal resource locators (URLs) and blocklisting URLs [23]. In the whitelist URLs list, several URLs were considered legitimate, while others were considered phishing. Similarly, blocklists contain all shortened URLs, unnecessary strings, long lengths, unstructured formats, and ambiguous domains. Whitelist URLs and blacklist URLs are shared with the general public to avoid visiting such URLs [24].

This approach prevents the user from phishing attacks; however, it is not as effective because of the higher computational cost of algorithm matching with a single string by string in a real-time environment. However, this method could not identify shortened, modified, and long-string phishing URLs [25]. Another ancient method is known as the rule-based phishing detection method. In this method, rules are defined for Web surfing. This type of detection requires expert knowledge of cybersecurity policies and Web filtering. According to this method, the user must know how to implement rules and analyze the URLs, either phishing or legit [26]. After the introduction of ML and DL models, phishing detection and identification became more efficient, but there are drawbacks to traditional ML models. In these models, feature extraction must be performed manually to identify phishing pages. This means that humans must write the rules that if such a string, word, or signature does not validate, that URL will be marked as phishing [27].

According to the World Wide Web Consortium (W3C) [28], a URL must contain elements such as the protocol, subdomain, domain, port number, database path, query parameters, and data that need to be retrieved. Ancient rules such as whitelists and blacklists, while checking the URL registered domain, if the domain exists, the system passes the URL as legitimate; otherwise, it blocks it [24]. For the authenticity of the URLs, the system needs to verify information such as domain expiry and registered date from a third party. Once the authentication rules are published, the attacker learns them and works according to the authentication laws to bypass the system. Thus, ancient methods have not been successful in controlling phishing attacks. Several models were used for anti-phishing after the ML and DL models. The phishing detection mechanism uses labeled data to classify phishing and legitimate websites. Different state-of-the-art models have been used for Web phishing detection and identification in ML and DL [24], [29], [30], [31], [32], [33], [34], [35], [36]. The fundamental use of these models is to identify and classify phishing links correctly. Therefore, the models are differentiated based on their accuracy and computation time. The higher accuracy and lower time computation models are considered the best models for detection.

Phishing attacks have increased dramatically due to the increased number of members on social media and online businesses. Therefore, cyber risks and threats have increased, needing to be addressed appropriately [37]. The complex nature of hyperlinks makes it difficult for the human eye to recognize original and fake links. Therefore, cybersecurity experts are paying more attention to the detection of counterfeit URLs. Phishers use advanced techniques and methods after learning the modern methods of ML and DL [27].

Several research papers have been published on phishing detection methods. In [38], the authors have analyzed different phishing solutions based on different parameters. The authors discussed lists of phishing techniques used on other devices and provided countermeasures against phishing attacks in four major categories: AI-Anti-Phishing models, Classical methods based on different scenarios, and lists-based. The authors concluded that the appropriate feature selection method gives a higher output for better results and that the model shows the highest accuracy compared to other AI models. However, the authors did not investigate other ML and DL methods proposed in [39], [40], [41], [42], [43], such as SVM, LSTM, NB, and other modern DL models, which can detect with accuracy rates from 99.00% to 99.62%.

In [44], the authors explained the two main types of phishing attacks, including social engineering and the use of malware. The authors also discussed feature extraction techniques based on some rules. However, they did not discuss the challenges associated with feature extraction, limitations, which feature extraction technique is suitable for which ML DL model, and how accurately required features were extracted as given in these [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45] papers.

In [46], authors provide phishing attacks, and their solutions are categorized into three main types: URL-based, content-based, and hybrid approaches. After carefully studying and comparing the proposed methods, they concluded that a hybrid approach is best for the detection and prevention of real-time phishing. However, they did not discuss the challenges associated with implementing the model and dataset described in these [47], [48] papers.

In [49], the authors explained different anti-phishing methods with phishing techniques. They discussed nine

different datasets used for phishing detection methods. They also wrote about 18 different AI-based models and compared their results. They discussed various challenges and limitations of the models, such as precision and over-fitting. However, they did not discuss the methods to reduce the model over-fitting and improve accuracy, as discussed in these papers [49], [50].

The authors in [51] discussed different email phishing detection techniques, including email spoofing using modern ML methods and natural language processing (NLP). NLP and ML are used for feature extraction and to detect malicious email content. In [52], the author explained that the analysis is based on URL parts, page contents, and Web page coding to find whether any tag or part of the code is modified or redirected elsewhere. Then, all models' performance will be compared to find the best one. In [53], the authors did not clearly explain the feature detection methods. There are many ways to extract features, such as manual selection methods and applying ML and DL models. However, there is a problem with the ML and DL models because the analyst has to manually select features that will be useful for the current dataset only. In case the data set changes, the feature selection technique fails. Furthermore, the authors did not review the modern methods for feature selection.

The author reviews modern AI-based phishing detection models in [54]. This paper divided the detection methods into four categories: ML DL-based, scenario-based, hybrid approach, and list-based. The author fails to provide a more detailed review of AI models that can be used confidently for phishing detection. This paper lacks data processing and feature extraction techniques for phishing detection datasets. In [55], they proposed a novel approach for detecting phishing websites by combining Support Vector Machines (SVMs) with nature-inspired optimization algorithms. SVMs are robust classifiers that aim to find an optimal hyperplane to separate data points into different classes. By integrating these algorithms, the researchers achieved promising results in identifying phishing URLs. The author and colleagues developed an anti-phishing browser that leverages the Random Forest algorithm and a rule-based extraction framework.

In [56], the authors proposed an RF method for the detection of Web phishing. The study is based on rule-based detection. The features extracted for modeling are based on the RF model. In [57], the author gives us a view of the different phishing techniques. This study is based on the hybrid approach for phishing detection, and features are extracted using XGBoost and Gradient Boost. The author used the same models for noise removal from the dataset. The author and collaborators explore the interaction of swarm intelligence and deep learning for phishing detection. Swarm intelligence draws inspiration from collective behavior observed in natural systems (e.g., ant colonies, bird flocks). Combining these principles with deep learning allows their I-BBA model to be observed [58].

In [59], the authors introduce a novel approach that combines support vector machines (SVMs) with nature-inspired optimization algorithms. SVMs are robust classifiers that aim to find an optimal hyperplane to separate data points into different classes. By integrating these algorithms, the researchers achieved promising results in identifying phishing websites. In [60], the study investigates techniques for detecting spoofed websites, which often mimic legitimate sites to deceive users. The authors explore machine learning models, feature engineering, and anomaly detection methods. Their work enhances the accuracy of identifying fraudulent Web pages. Their study involves URL feature extraction, behavioral analysis, and model training. Taking into account the lexical and content-based features, they contribute to the development of robust detection mechanisms [61]. Another study [62] explores a comprehensive approach to Web phishing detection. The authors combine Web crawling techniques, cloud infrastructure, and deep learning frameworks. By analyzing Web content, network traffic, and behavioral patterns, their model provides robust protection against phishing attacks.

The overview paper [63] by Scholar and the team critically examines existing methods for detecting phishing sites. They discuss zero-day attacks, adversarial evasion, and real-time detection challenges. In this work, the authors investigate the effectiveness of combining multiple machine-learning models for phishing classification. They achieve improved performance by leveraging ensemble techniques such as stacking or blending. Their study emphasizes the benefits of model fusion in security applications.

In [64], researchers and colleagues propose an ensemble model designed explicitly for detecting phishing intrusions from URLs. Their approach combines decision trees, random forests, and gradient boosting. By considering diverse classifiers, they enhance the robustness of their detection system.

The research article [65] opens up a new dimension to the world: the DRL-BWO algorithm, optimized by Black Widow Optimization, for UAV networks. In addition, DRL incorporates an enhanced reinforcement learning-based DBN for the detection of intrusions in UAV networks. The BWO algorithm is applied to the parameter optimization of the DRL approach. It enhances the performance of intrusion detection in UAV networks, securing communication over the UAV.

A. RESEARCH GAP IN LITERATURE REVIEW

During the literature review, we identified several research gaps that highlight areas where further investigation is needed. A complete review of these research gaps is provided in Table 1.

III. RESEARCH METHODOLOGY

The literature review section of this article is based on various research papers, including surveys, reviews, and

TABLE 1. The research gap review table summarizes the methods employed and the limitations identified in various studies focused on phishing detection. It highlights gaps such as the need for model adaptability, handling modern attack techniques, effective feature selection, dataset diversity, and real-world applicability across different approaches.

Sr. No	Authors	Methods	Limitation
1	Abad et al. [22]	SVM, RF, DT, KNN, RNN, NB, Optimizer, DRLSH, BPLSH	There is a lack of explanation on model adaptability with multiple doors, handling of modern attack techniques, and real-world dataset implementation.
2	Anupam et al. [28]	SVM, BAT, WA	In this paper, the author did not address the issues of imbalanced data, feature engineering, or new phishing techniques.
3	Shahrivari et al. [29]	DT, LR, KNN, ANN, RF, Ad	Lack of feature selection/extraction methods, lack of exploration of deep learning techniques, and inadequate real-world scenario applicability.
4	ALSARIERA et al. [31]	ABET, BET, RoFBET, LBET, ANN, SVM, RF, KNN, SVM, DT	There is a need to explore diverse datasets beyond Mendeley, scalability assessment for larger datasets, hyperparameter impact analysis, and adaptation to evolving threats.
5	Lokesh & BoreGowda [35]	RF, KNN, DT, L-SVC	Lack of specific feature discussion, detailed algorithm comparison, real-world dataset robustness, and adaptation to emerging phishing techniques.
6	Butt et al. [38], [39], [65]	LSTM, SVM, NB, ISHO, WARM, Firefly, BAT	Lack of comparative analysis, absence of discussion on URL feature selection, and limited applicability to specific datasets.
7	Zamir et al. [44]	[RF], [NN], NB, KNN	Lack of integration methods for various data sources repetition of existing models from previous research.
8	Jovanovic et al. [57]	XGBoost, WI, MOFA	In this study, the author discussed only selected features and did not discuss handling blank images and shortened URLs.
9	KARIM et al. [60]	DNN, RF, NB, Na	Absence of generalization to evolving phishing methods, inadequate feature extraction for large datasets, and limited real-world scenario assessment.
10	Zieni et al. [61]	list-based, similarity-based, and machine learning-based	Inadequate handling of imbalanced datasets, lack of feature details for classification, and focus on controlled experiments over real-world scenarios.
11	Shaukat et al. [62]	SVM, RF, MP, XGBoost	Limited to three datasets, lacks a universal feature extraction method, and inadequate analysis of ML and DL models.
12	Korkmaz et al. [66]	XGBoost, RF, LR, KNN, SVM, DT, NN, NB	Missing exploration of mitigation strategies for detected phishing attacks.
13	Adebowal et al. [67], [68]	LSTM, CNN, IPDS Classifier	Uncertainty on adaptation to new phishing methods, lack of automated feature selection for large datasets, and inadequate real-world scenario assessment.
14	Maci et al. [69]	DL, DRL, MDL, ICMDP	Unclear performance with increasing features and large datasets, lack of exploration on additional features' impact, and absence of real-time scenario evaluation.

original research articles. These articles were selected from high-impact journals published between 2020 and 2024 as shown in Figure 1 and Figure 2. The selection process began with a broad query as shown in Figure 3 using the general keyword "Web Phishing Detection," which generated a list of numerous relevant papers.

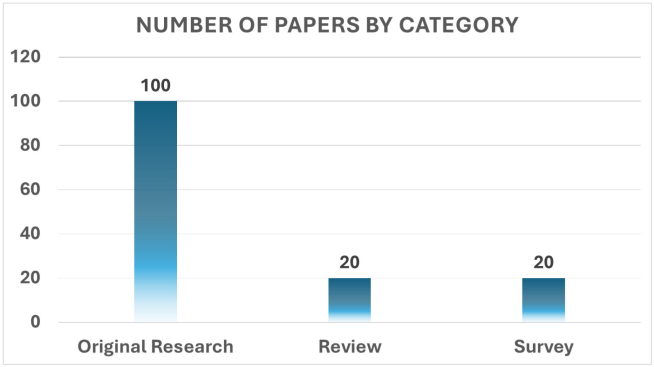

FIGURE 1. Illustrates the distribution of papers across different categories: original papers, review papers, and survey papers.

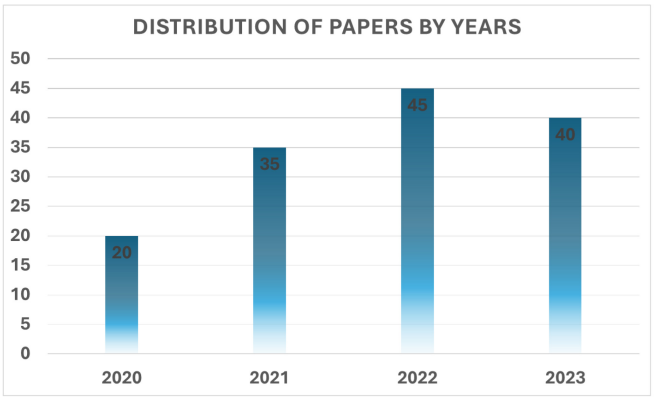

FIGURE 2. Illustrates that the contents are sourced from the most recent publications from the years 2020 to 2023.

We filtered the papers to focus on those more specific to the domain. Initially, the query returned approximately 3,500 articles. After narrowing the selection to more specific documents, we identified about 500 articles relevant to the query. However, many of these documents were written before 2020. By refining the query, we ultimately selected the most recently published papers from 2020 to 2024. These articles include 100 original research articles, 30 survey articles, and 20 review articles.

IV. PHISHING ATTACK METHODS, STEP-BY-STEP PROCESS, EFFECTS AND TECHNIQUES

Phishing attack refers to various techniques to share malicious link share with end users. There are several methods described below.

A. PHISHING ATTACK MODES

1) USING ELECTRONIC MAIL

Nowadays, almost everyone has an email address to exchange data and correspondence. According to an article published in 2024, about 3.4 billion emails are sent daily, and 1.2% of them contain malicious URLs [9]. Another report states that approximately 96% of these emails contain phishing URLs [9]. Users are often attracted to these emails because the content appears desirable, leading them to visit

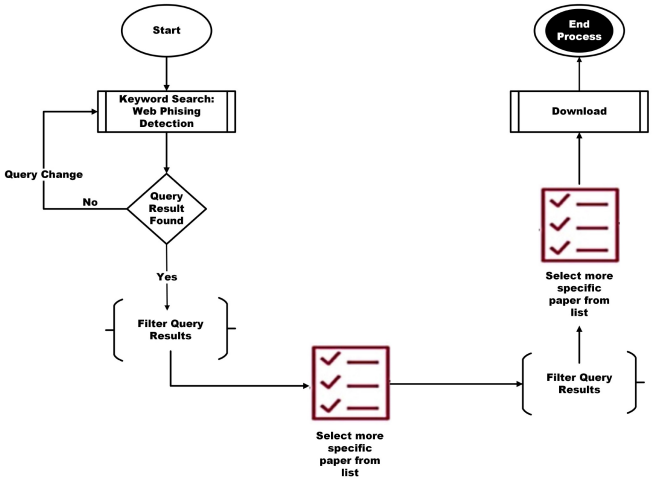

FIGURE 3. The research process flow diagram for Web phishing detection outlines the systematic steps involved in identifying, filtering, and selecting academic papers related to Web phishing detection. It begins with a keyword search, progresses through various filtering stages, including filtering by year, and concludes with the download of selected papers. This process ensures a focused and relevant literature review.

the links provided [71]. Some emails offer incentives, while others claim that "your password has been hacked; please reset your password using the following link." Upon reading these emails, users might click on the link and unknowingly provide their credentials directly to phishers [72]. Phishers then use these credentials for illegal purposes, such as cash withdrawal. According to a report published in 2022 by the Federal Bureau of Investigation (FBI), about 10.3 billion USD was lost due to email phishing [73].

2) USING SOCIAL NETWORKS

The world has become a global village due to the invention of technology and the rise of social media. People from all over the world connect on social media platforms, share images and thoughts, sell products, and launch business campaigns to promote their businesses [73]. These activities make social networks an effective tool for hackers to reach their target audiences [74]. Hackers may create fake discount sales campaigns and include phishing URLs in the descriptions, requiring users to fill in sensitive information, such as bank, credit card, or debit card details. As a result, hackers obtain the necessary information, leading to financial loss and mental distress for users. According to a CBS institution report, about $4.1 billion was lost due to social media phishing attacks [75].

3) USING SMS (SHORT MESSAGE SERVICE)

Phishing attacks have increased due to changes in the way people communicate. Since the invention of the smartphone, social communication modes have evolved [76]. People enjoy sending SMS messages to their loved ones to stay in touch because it is an inexpensive communication method. Similarly, companies send SMS messages to promote product campaigns to end users [77]. Using the same strategy,

phishers send SMS messages containing links that request the recipient to fill in personal information. Sometimes these messages claim to be surveys, while other times they announce the launch of a new product. The user may click the link, believing that they will receive a significant discount, and then fill in all the required fields. Consequently, all sensitive information is transferred to phishers, who use it for illegal activities, leading to financial loss for the user [78].

4) USING LIVE MESSENGERS

There are other ways to communicate with friends and family through live messengers such as Yahoo Messenger, Y-Mail Messenger, and Hotmail Messenger. People share their location, pictures, documents, and sensitive information via these platforms. Phishers often pretend to be company representatives, sending phishing links and requesting recipients to fill in their information to participate in a supposed lucky draw. As a consequence, users may face financial loss and account termination [79].

5) USING BLOG POSTS AND COMMUNITY FORUMS

Blog posts and community forums are websites where people share their thoughts and problems, discuss issues, and get feedback from others. They are widely used for sharing information and completing surveys, forms, and other details. Attackers may create fake surveys, pretending to be the forum owner, to obtain members' details, which are later used for illegal purposes [80].

V. HOW WEB-PHISHING WORKS

Phishing attacks differ entirely from hacking or gaining unauthorized access through various means. In phishing attacks, the phisher is typically a technical person with deep knowledge of Web development, SQL tools, machine learning, and the creation of fake Web pages. Phishing attacks involve several steps as shown in Figure 4 and outlined below.

1) *Develop Strategy:* First, phishers develop a strategy to identify and target a specific community.
2) *Development of Phishing Website:* Once the target is identified, the next step is to develop a website that closely resembles the original.
3) *Domain Registration:* The developed website is then hosted on a domain using a URL similar to the original one, with minor modifications-for example, `https://www.bankalfa.com.pk` to `https://www.bankalfaa.com.pk`.
4) *URL Shortening:* After the website is hosted, the next step is to shorten the URL to share with the target audience.
5) *Forward Phishing URL:* Once the URL is shortened, it is forwarded using various means, such as social media, email, SMS, WhatsApp, and other sources.
6) *User Click:* The phishing link contains attractive details, content, offers, and language that persuade the user to click. Users then fill in the requested information, which is transferred to the phisher's Web server.
7) *Collect Required Data:* When users click the link, they are prompted to fill in information that is directly saved on the phisher's Web server. Phishers then use this data for other purposes.
8) *Illegal Use of Data:* Phishers collect data and use it for illegal purposes, resulting in financial loss and compromise of online accounts.

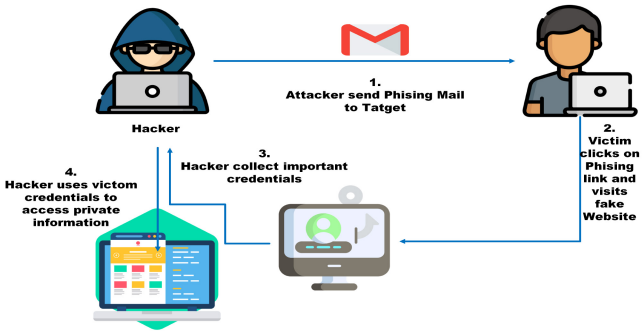

FIGURE 4. It illustrates the typical steps of a phishing attack. It begins with the attacker sending a phishing email to the target, followed by the target clicking on a phishing link. This action leads them to a fake website where their credentials are collected. Ultimately, the attacker uses these credentials to access private information on the original website.

A. PHISHING ATTACK DAMAGES

A phishing attack is a hacking activity in which the phisher obtains personal information accessed via URL, which may cause the following effects on the user:

1) By gaining unauthorized access, phishers cause financial damage to the end user without consent.
2) It may tarnish the user's online reputation, leading to reduced business opportunities.
3) Phishing attacks diminish trust in companies, resulting in decreased customer engagement and business activities.

B. POSSIBLE PHISHING TECHNIQUES

Phishing attacks employ various methods to deceive users and obtain sensitive information. The following are some possible techniques used by phishers as shown in Figure 5:

1) SOCIAL MANIPULATION

Social manipulation [81] involves tricking individuals into sharing personal information without realizing the risk of being hacked. This often involves following a URL to provide information, thinking it is necessary to secure their account. For example, receiving an email that appears to be from a bank asking for account confirmation can prompt a person to click on a link and fill out a form, believing that they are providing accurate information to their financial institution. Phishers exploit this by counting on the user to click on the URLs, leading to unauthorized access.

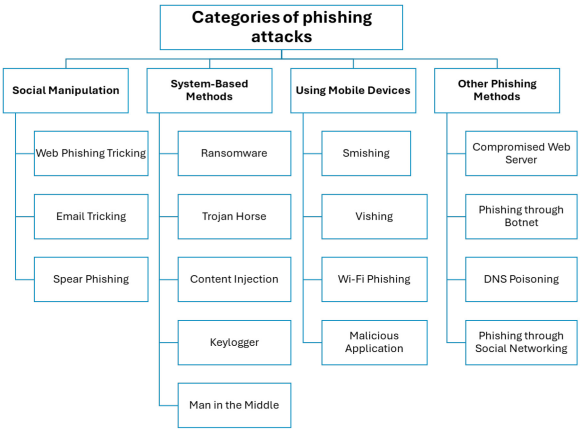

FIGURE 5. Phishing Attack Methods diagram categorizes various phishing techniques into four main groups: Social Manipulation, System-Based Methods, Using Mobile Devices, and Other Phishing Methods. Each category lists specific types of attack, highlighting the diversity and complexity of phishing tactics used to compromise security across different platforms.

1) Web Phishing Tricking: Phishers create fake Web pages that closely resemble legitimate ones to deceive users. These fake pages often mimic login screens or other interactive elements to trick users into divulging sensitive information such as usernames and passwords.

2) Web Phishing Cloning: This phishing attack involves cloning Web pages to resemble the original site. Phishers use online or offline tools to create these fake Web pages [82]. They primarily focus on replicating login pages to ensure users believe they are on the original site rather than the phishing page. The cloned site can be detected visually and using browser security features, which may alert users about the phishing page [83]. Therefore, phishers often remove identification tags, codes, divs, and frames from the phishing Web page to avoid detection [2].

3) Email Tracking: Another method of obtaining user details is by email. Phishers might send two types of email to end users:

1) The attacker designs a Web page that closely resembles the original, including visuals such as graphs and pictures. The phishing URL is hidden using JavaScript, displaying the actual URL to prevent user confusion and reinforce legitimacy.
2) Attackers send download links for registered software, which include malware. Once downloaded and installed, the malware begins sharing information with the phisher, compromising the user's system.

4) Spear Phishing: In this attack, the phisher targets an individual from a reputable organization. The attacker monitors the target's social media accounts to learn their schedule. Once enough information is gathered, the phisher contacts the target via email, pretending to be a company manager, and requests them to fill out a form for an urgent meeting. The target, believing the request is legitimate, shares sensitive information with the attacker. Using spear phishing methods, several prominent authorities have been attacked, suffering significant financial losses and data breaches [84].

2) SYSTEM-BASED METHODS

The following are the main types of system-based phishing attacks:

1) Ransomware Virus: Ransomware is a modern type of malware that deeply affects users, causing financial and sensitive data loss [85]. In this attack, phishers send harmful links containing ransomware. Once the user clicks on the link, the malware is downloaded and installed on the target computer. After installation, all files are encrypted, and a pop-up displays the attacker's account details for data recovery [86].

2) Trojan Horse Virus: The Trojan horse is another type of malware similar to ransomware, but it functions differently. The attacker sends a Trojan via email, text message, or WhatsApp link to download media or applications [87]. The Trojan then installs on the target device, running in the background without user knowledge [87]. Once installed, it sends sensitive data, including bank details, to the attacker [88].

3) Content Injection: In content injection attacks, also known as cross-site scripting (XSS), attackers exploit website vulnerabilities to add harmful code to Web pages [89]. This code, often JavaScript, can perform dangerous actions such as stealing cookies, session tokens, or other sensitive user data stored in the browser [89]. Once inserted, the code can capture what users type in forms or redirect users to fake pages that resemble legitimate websites. According to reports, about 2.0 million websites suffer from content injection attacks [47], making it a prevalent online threat. These attacks often occur due to poor input validation or lack of sanitization, allowing attackers to inject malicious scripts into trusted sites. To mitigate these risks, Web developers should implement security measures such as Content Security Policy (CSP), input validation, output encoding, and regular security audits to detect and fix vulnerabilities before exploitation [90].

4) Keylogger / Screen Logger: A keylogger is an application installed on a system that tracks user keystrokes [91]. This malicious software can be installed physically or remotely on the target system without the user's knowledge. Once installed, the software tracks keystrokes and website visits, sending the data to the phisher's email or configured location.

5) Communication Crack Phishing Attack: The communication crack phishing exploit targets vulnerabilities within wireless networks, such as open or public Wi-Fi [92]. The attacker conducts a man-in-the-middle attack using tools like Wireshark or SSLstrip to capture and decrypt traffic between users and the organization's servers [93]. Users are drawn to a rogue access point that mimics a real network, capturing sensitive information, including login credentials and financial data [94]. Alternatively, the attacker may redirect users to phishing sites for credential

harvesting or inject malware during communication [76]. The attacker exploits weaknesses in encryption, such as the KRACK vulnerability in WPA2 protocols, to decrypt and modify network traffic [93]. Mitigating such risks involves implementing robust encryption, regularly updating security protocols, and educating users about the risks of connecting to open Wi-Fi networks and recognizing phishing attempts [95].

3) USING MOBILE DEVICES

Over the decades, mobile device use has become widespread. Today, phones are primarily used for communication, information sharing, and online shopping. Phishers generate phishing links and share them via SMS, email, WhatsApp, and third-party mobile apps [96]. When users click on these links and provide information, their devices are compromised. The attacker gains control over the mobile device, leading to misuse. Several mobile phishing methods are listed below.

1) Smishing: Smishing is a blend of "SMS" and phishing, hence the name. Attackers send SMS messages with malicious links, prompting users to click and fill out forms [97]. For example, if the government announces financial assistance for needy people, phishers might generate messages with phishing links requesting sensitive personal, banking, and other information [98]. Users, thinking the message is from a legitimate agency, share sensitive information without verifying its authenticity, allowing phishers to collect the information.

2) Mali-App: Mali-App stands for malicious applications, referring to apps not verified by Google's security mechanisms [99]. These third-party apps can harm mobile devices [100]. Phishers gain unauthorized access through these apps and start collecting data and sensitive information.

3) Vishing: Similar to smishing, vishing is a type of phishing attack where attackers use voice manipulation to mimic legitimate voices [101]. The attacker sets up a VoIP platform and uses voice changer software to imitate an authentic voice. For example, an employee of a reputable company may receive a call from someone pretending to be their manager, asking for critical login details. This method is often successful due to the high level of trust involved.

4) Wi-Fi Phishing: Wi-Fi phishing is a modern phishing method involving open Wi-Fi hotspots. While users access the Internet through these hotspots, attackers monitor traffic for valuable information. Alternatively, users may be asked to fill out a registration form, which includes personal information stored in the phishing system [47].

4) OTHER PHISHING METHODS

1) Compromised Server: In this phishing attack, the phisher hacks a target website's server and uploads a malicious toolkit. The phisher silently controls the server and hosts a similar Web page to divert users, making them believe it is legitimate [102]. By compromising the server, hackers save on hosting costs. A study [103] found that about 76.5% of websites are hosted on compromised servers.

2) Phishing Using Botnets: Botnets [104] are networks of computers connected to perform specific tasks. These computers ensure the smooth operation of applications like VoIP and chat systems. However, phishers may insert malicious applications into one system, sending numerous emails from the organization's systems. This type of attack is particularly dangerous and difficult to detect.

3) DNS Injection Attack: Phishing attacks that rely on DNS manipulation have become more sophisticated and dangerous. Many fake websites are hosted, and traffic is diverted using DNS injection methods [103]. Once the DNS cache becomes infected, it starts transmitting data to malicious URLs.

5) SOCIAL NETWORKING PHISHING METHODS

Social networking sites are significant targets for phishing attacks. The following methods are used to exploit these platforms.

1) Malicious URL Sharing: Social media has become integral to daily life, especially for selling products and launching campaigns. Attackers design malicious URLs and share them with members, friends, or company contacts, asking for information related to meetings [48]. This information may include system usernames and passwords. The relevant group interacts with the URL, providing the requested information, which is then transferred to the attacker's account.

2) Masked URLs: In this attack, the phisher shares a URL while pretending to be the admin of a social media group [48]. The URL links to a dummy form that requests sensitive information. Once group members fill out the form, the attacker uses the information for unauthorized access [22]. Attackers may send messages from hacked accounts or even request loans, pretending to be legitimate contacts [105].

3) Forged Profile: Another deceptive method is using a fake profile. The attacker targets a prominent social media figure, monitoring their profile, posts, and comments daily. After gathering enough information, the attacker creates a similar profile, mirroring the content, and adds new friends to deceive others.

VI. PHISHING DETECTION BASED ON WEBPAGE SCREENSHOT

Screenshot-based phishing detection is a novel approach that utilizes visual analysis techniques to identify fraudulent Web pages [106]. In this technique, a screenshot of a webpage is analyzed for visual elements such as logos, text layout, color schemes, and design patterns to determine whether the website is genuine [107]. Machine learning algorithms are employed to compare the screenshot against legitimate and phishing sites within a database, acting as image processors for anomaly detection or identifying potential threats [108]. This technique is particularly useful for detecting phishing

attempts that visually replicate a real website, which might otherwise go unnoticed by traditional text-based detection methods [109]. Additionally, the approach can be integrated with other detection techniques, such as URL analysis and text feature extraction, to enhance the overall accuracy and reliability of phishing detection systems [110]. Phishing detection through screenshots offers powerful protection against growing phishing attacks by focusing on the visual content of Web pages [110].

VII. WEB PHISHING DETECTION DATASETS

In Web phishing detection, the latest datasets play a vital role in model training and detection. Several datasets are available online for phishing detection, varying in size from small to large. As phishing attacks continue to increase, there is a growing need for advanced models for detection. A well-trained model relies on the latest datasets with a comprehensive set of features. Table 2 provides details of some existing phishing datasets. In this section, we will review existing datasets, their features, and the models used for feature extraction using these datasets, as well as their advantages and drawbacks.

- Phisher-Tank Dataset: The Phisher-Tank dataset contains almost 2 million entries from phishing websites that are blocklisted on the Internet. Approximately 90% of these websites are offline due to removal from Internet sources, while 11,000 remain active as phishing sites. This dataset is compiled and maintained by the Talos group of companies. The Phisher-Tank also offers an online service where users can paste a Web URL to check its legitimacy. If a URL is identified as phishing, it is automatically added to the Phisher-Tank database.
- Dataset: Alexa provides an online system that analyzes website performance, checking how efficiently a website operates and whether it contains hidden phishing links. This service, launched and controlled by Amazon, helps identify and report phishing sites.
- Wein Phishing Dataset: The Jet-Wein phishing dataset is an open-source system that uses an API to blocklist phishing websites. This dataset contains approximately 15,000 phishing URLs.
- Crawl Dataset: The Common Crawl dataset is generated using a Web crawler. Once URLs are crawled, they are tested with a phishing detector to identify phishing URLs. This dataset contains over 10,000 entries.
- OpenPhish Dataset: OpenPhish is a free open-source platform for Web analysis and phishing detection. This system includes almost 19 million phishing links.

TABLE 2. Table provides a comprehensive list of datasets used in phishing detection research, categorized by their nature (Legit or Phisher), update year, and accessible URLs for further exploration.

Dataset Name	Update Year	Type	URL
Phishing-Tank [111], [112]	2024	Phisher	https://phishtank.org/developer_info.php
Alexa-dataset [112]	2022	Legit	https://www.similarweb.com/website/alexa.com/
Wein-dataset [113]	2021	Phisher	https://jeowein.net/
Crawl-dataset [114]	2021	Legit	https://commoncrawl.org/
Open-Phish-dataset [114], [115]	2021	Phisher	https://openphish.com/
Phishing-army dataset [114], [115]	2021	Phisher	https://www.phishing.army/
Kaggle-phishing dataset [116]	2021	Legit	https://www.kaggle.com/datasets/shashwatwork/phishing-/dataset-for-machine-learning
UCI-Dataset [49]	2022	Phisher	https://data.world/uci/phishing-websites
Parsed-dataset [117]	2022	Legit	https://doi.org/10.7910/dvn/omv
Yahoo-Phishing [118]	2022	Legit	https://webscope.sandbox.yahoo.com/
Yandex-phishing [118]	2022	Legit	https://Yandex.com/dev/xml/
Phished-dataset [116]	2022	Phisher	https://www.medien.ifi.lmu.de/team

A. FEATURE EXTRACTION METHODS

1) DYNAMIC FEATURE EXTRACTION METHOD

In phishing detection, accurate model performance relies on extracting relevant features from the data, enabling the model to train effectively and differentiate between phishing and legitimate websites. Researchers [66], [82], [112] have proposed a dynamic feature extraction method based on feature weights. They extracted 17 features from the dataset, categorized into three groups: address-based features, script-based features, and tag-based features. This paper [82] discusses automatic feature extraction from the URL and address bar without using third-party tools. However, the WHOIS database is used for domain name and registration data. Additionally, page scores are extracted from the Google rank database. After extracting the features, they are weighted based on their scores and averages, which are then used to train and test the model.

2) MACHINE LEARNING FEATURE EXTRACTION

In 2015, researchers proposed a new dataset containing over 11,000 instances with 30 features. These features were

extracted using machine learning models to improve phishing detection accuracy.

3) FEATURE EXTRACTION USING NLP (NATURAL LANGUAGE PROCESSING)

The integration of NLP in machine learning has enabled researchers to extract features from phishing URLs more effectively. Character-level characteristics are extracted using machine learning models, classified, and used for model training [119]. NLP facilitates feature extraction from datasets at the character level, allowing the model to use the TF-IDF method for feature scoring. Once extracted, these features are used for model training.

4) RECURSIVE FEATURE ELIMINATION METHOD

Known as REF, this method was introduced in [45]. It involves extracting all features and removing weak ones based on a threshold value.

5) USING PRINCIPAL COMPONENT ANALYSIS METHOD

Proposed in [45], this method begins with preprocessing, followed by the selection of features after removing redundant and unwanted ones. Techniques like median filtering or adaptive thresholding are commonly used. The wavelet packet transform (WPT) is another method that can be applied in this technique [120].

6) USING INFORMATION GAIN METHOD

Information Gain is a popular method for feature extraction in phishing datasets. As discussed in [45], this method uses probability functions to identify vital features based on probability scores, selecting features that meet algorithmic criteria and discarding others.

7) RELIEF RANKING FILTER METHOD

Used in [121], this algorithm extracts features based on a near-neighbor score algorithm. First applied to the UCI dataset, features are scored, compared with near nodes, and selected using the NNS algorithm. This method identified 22 features from the UCI-Phishing dataset.

8) FRS (FUZZY ROUGH SET) FEATURE EXTRACTION METHOD

This algorithm, related to rough set theory, identifies related data points and compares nodes and classes to discern between them. For example, it compares every feature of a phishing website with another to check legitimacy. Features are extracted based on class matches, represented as 0 or 1 in the original UCI phishing dataset [121], [122].

9) EL-RASHIDY FEATURE EXTRACTION METHOD

First introduced by El-Rashidy in [123], this algorithm operates in two steps. Initially, it extracts features from the dataset and trains the model using the RF ML model. During training, it assesses accuracy and removes features with lower accuracy. After training, it refines features and selects high-accuracy features for further testing. While effective for small datasets, this method is less suitable for large datasets due to computational costs.

B. URL-BASED FEATURE EXTRACTION METHOD

This method involves extracting URL properties using various techniques. These properties include URL syntax, domain name, registration and expiration dates, website age, hosting server location, IP address, and DNS details. Extracted features help identify whether a URL is legitimate or phishing. Four main categories of URL feature extraction exist:

1) Structure-Based Properties: This category examines URL structure, syntax, communication protocol, domain name, hosting server, and discriminating tokens like ',' and '.'. Features are extracted through tokenization, removing URL elements. Another method [124] involves bag-of-words comparison to identify matching URLs, extracting relevant features while discarding others. Binary notation determines domain or property length, returning 1 for matches and 0 otherwise.

2) URL Pattern Properties: This method analyzes URL statistics, such as length, domain name, hosting validity, and expiry dates. Extracted statistics are compared with legitimate URL statistics to identify relevant characteristics [125]. Term and inverse URL frequencies are calculated for feature matching. Frequency-level word counts are compared to determine matching features, using the Jaccard Index Pairwise (JIP) method to establish feature associations.

3) Domain-Based Properties: This method extracts features related to the domain name, registration and expiry dates, and other URLs using third-party tools like WHOIS, WHOAMI, and DNS-LOOKUP. Information obtained from external sources is used as model training features.

4) Ranking-Based Properties: Phishers often target well-known websites due to their high traffic volumes, increasing the likelihood of encountering unsuspecting victims. Consequently, website ranking is a critical feature during feature extraction in phishing detection systems.

1) HYPERTEXT-BASED FEATURES

These features relate to the website's source code and include different HTML tags and forms. They are significant in training data for identifying parsed HTML or phishing HTML pages. Several studies focus on these features, including [12], [43], [122], [123], [124], [125], [126], [127], [128], [129]. There are three main types of these features:

1) Text-Based Properties: This property involves analyzing the complete Web page code, including the HTML tags. First, tags are extracted, and a frequency table is used to determine how often each tag appears in the code. The analysis checks for any tampering or unusual information outside the normal structure [130], [131], [132]. These features are used for model training. Additionally, form-related tags are identified

and used as feature parameters in the training process. If a website contains graphical visuals, properties related to those graphics are also extracted to identify manipulated pages. These features are crucial for phishing detection [124].

2) Visual-Based Properties: Visual properties also play a role in phishing detection. Features such as size, width, height, brightness, and darkness [133] are extracted for model training. This helps identify malicious images on websites [126].

2) FILTER METHOD FOR FEATURE EXTRACTION

The filter method for feature extraction is based on statistical calculations and classifications using techniques like Chi-Square [132], Information Gain [134], Gain Scores, Correlation Method [135], [136], [137], and the Fisher method [136], [137].

3) WRAPPER METHOD FOR FEATURE EXTRACTION

This method selects features using machine learning models such as genetic algorithms, greedy forward selection, and classifiers [138]. It is used to determine 177 optimal features from the Web page source and URL [139].

4) TOKENIZATION AND VECTORIZATION

Tokenization is one of the techniques for feature extraction. It is considered as translating a single string into a sequence of one or more non-empty sub-strings. This method has been implemented to identify malicious URLs in previous work [136], [137]. The selected 10000 benign URLs and 10000 malicious URLs make up data1. Words such as "com" in URLs do not add value to them, so they are removed before tokenization. Tokenization is done using special characters, slash, dash, and dot. After tokenization of selected URLs, the data is then converted into the sparse matrix vector for machine learning [29].

5) CHARACTER N-GRAMS

The character N-grams extracted from URLs are overlapping sequences of N-consecutive characters, extracted from the URLs where the value of N varies between 1 and 10. For example, the first three bigrams of the URL "example.com" are "ex", "xa", "am". This is much richer than the bag-of-words approach used by researchers in [11] as it captures punctuation, misspellings, etc. in the URLs. Here N represents the length of the character substring. Table 3 provides comparative overview of different feature extraction methods.

VIII. ANTI-PHISHING METHODS

In this section, we will discuss different approaches to anti-phishing. These approaches include classical and state-of-the-art methods, including machine learning (ML) and deep learning (DL). The basic structure of anti-phishing methods is shown in Figure 6.

A. USER-BASED ANTI-PHISHING APPROACHES

1) USER-BASED ANTI-PHISHING TECHNIQUE

1) Educating Anti-Phishing Technique: Humans are often unaware of new threats, which makes education essential to learn about new methods and techniques. Education is crucial to teach people about phishing attacks, how phishing works, and how to identify phishing emails. Organizations worldwide must train their staff about phishing threats. Similarly, all government agencies should educate the public about phishing [109]. By educating employees and the general public, phishing attacks can be controlled.

2) Awareness About Security Warning Anti-Phishing Technique: Most phishing detection methods use browser plugins, which quickly identify suspicious websites and alert users when visiting a potentially dangerous site. Understanding security warning signs is crucial when human intervention is required. If users ignore these warnings, it could lead to negative consequences. Proper training on the recognition of security indicators is essential. Studies have shown that 60% of the users ignore warnings and proceed to phishing URLs without training, while the click-through rate for trained users was 0.

There are two types of warning: active warnings that prevent users from accessing phishing URLs and passive warnings that display a message while allowing access. Most contemporary Web browsers, such as Mozilla Firefox and Google Chrome, use passive warnings. Active warnings are more effective, since many users tend to ignore passive warnings. A study with 60 participants found passive warnings inadequate; 79% noticed active warnings, while only 13% noticed passive warnings [140], [141], [142].

3) Training Using Games Anti-Phishing Technique: Training methods that incorporate games are advantageous because they are convenient and easy to learn, providing a natural setting for teaching. Various developers have created interactive teaching tools to educate users on recognizing phishing attempts.

Before and after studies demonstrated the effectiveness of training games. Participants who played these games showed an increased awareness of phishing emails and websites. The training was integrated into users' daily routines, making it user-friendly. Periodically, instructive notes were sent to users after the program began. Research showed that only 30% of trained users clicked on fake links in emails they learned to recognize. Moreover, findings indicated that interactive training sessions are more successful than warning notifications [140], [141], [142].

A serious game was created to enhance users' ability to recognize phishing URLs. The game, "Phisher," is a solo, lightweight, intuitive, and narrative-driven game. The scenario begins with the player receiving several messages claiming they won a substantial sum of $600,000 and would be sent to an island if they input bank details. The player must capture the scammer using a boat, a hungry tiger, and no money, returning to the beach to survive. The game play

TABLE 3. The table categorizes extracted features from various studies into different properties such as lexical, statistical, network, reputation, textual/visual, and traffic. It lists the feature types, the number of features extracted, and the third-party services used for each category, providing a comprehensive overview of methods and tools applied in URL-based phishing detection research.

Feature	Paper	URL-based	HTML-based	No. of features	Third-party services
Lexical properties	Yassine et al.	✓	✓	112	WHOIS, Common Crawl
	Liu et al.	✓	✓	25	WordNet, Google Search
	Al-Rfou et al.	✓	✓	120	WordNet, Google Search
Statistical properties	Abualigah et al.	✓	✓	10	None
	Li et al.	✓	✓	18	None
	Wang et al.	✓	✓	12	SimilarWeb, Alexa ranking
Network properties	Alsuwaihli et al.	✓	✓	14	Social network analysis
	Wu et al.	✓	✓	22	Social network analysis, Majestic
	Zhang et al.	✓	✓	16	Social network analysis, Alexa ranking
Reputation properties	Bhatia et al.	✓	✓	22	Online reviews, social media sentiment
	Cao et al.	✓	✓	28	Online reviews, social media sentiment, Trustpilot
	Goyal et al.	✓	✓	15	Online reviews
Textual/visual properties	Feng et al.	✓	✓	18	Image analysis, sentiment analysis
	Xu et al.	✓	✓	20	Image analysis, named entity recognition
	Jiang et al.	✓	✓	24	Sentiment analysis, named entity recognition
Traffic properties	Ali et al.	✓	✓	15	Web traffic analysis
	Hassan et al.	✓	✓	12	SimilarWeb, Similarweb Pro
	Tripathi et al.	✓	✓	18	SimilarWeb, Alexa ranking

progresses as the player answers questions about phishing. The surveys of the participants in the pre and post-game events showed that the number of correct answers increased from 4-7 to 5-8 after playing the game [140], [141].

The confidence level of accuracy rose from 4.09 to 4.47 ($p < 0.05$), and the accuracy improved from 0.70 to 0.795 ($p = 4.12 \times 10^{-142}$). The false negative rate decreased from 0.22 to 0.14 ($p = 5.03 \times 10^{-091}$), while the false positive rate decreased from 0.34 to 0.25 ($p = 7.71 \times 10^{-076}$). 25% of the participants played the game more than once, indicating the appeal and participation of the game [140], [141].

4) User Response Anti-Phishing Technique: Research conducted by various academics has investigated why individuals become susceptible to phishing schemes. These studies also evaluate whether users examine the URL, browser toolbar, or other security indicators. Many computer users are targeted due to their ignorance of warning signs and indicators when using toolbars. The survey found that most of the participants had no bank account and were unfamiliar with financial jargon, making them unable to recognize 90% fraudulent sites, even when they visually resembled legitimate ones. The studies discovered that 23% of the

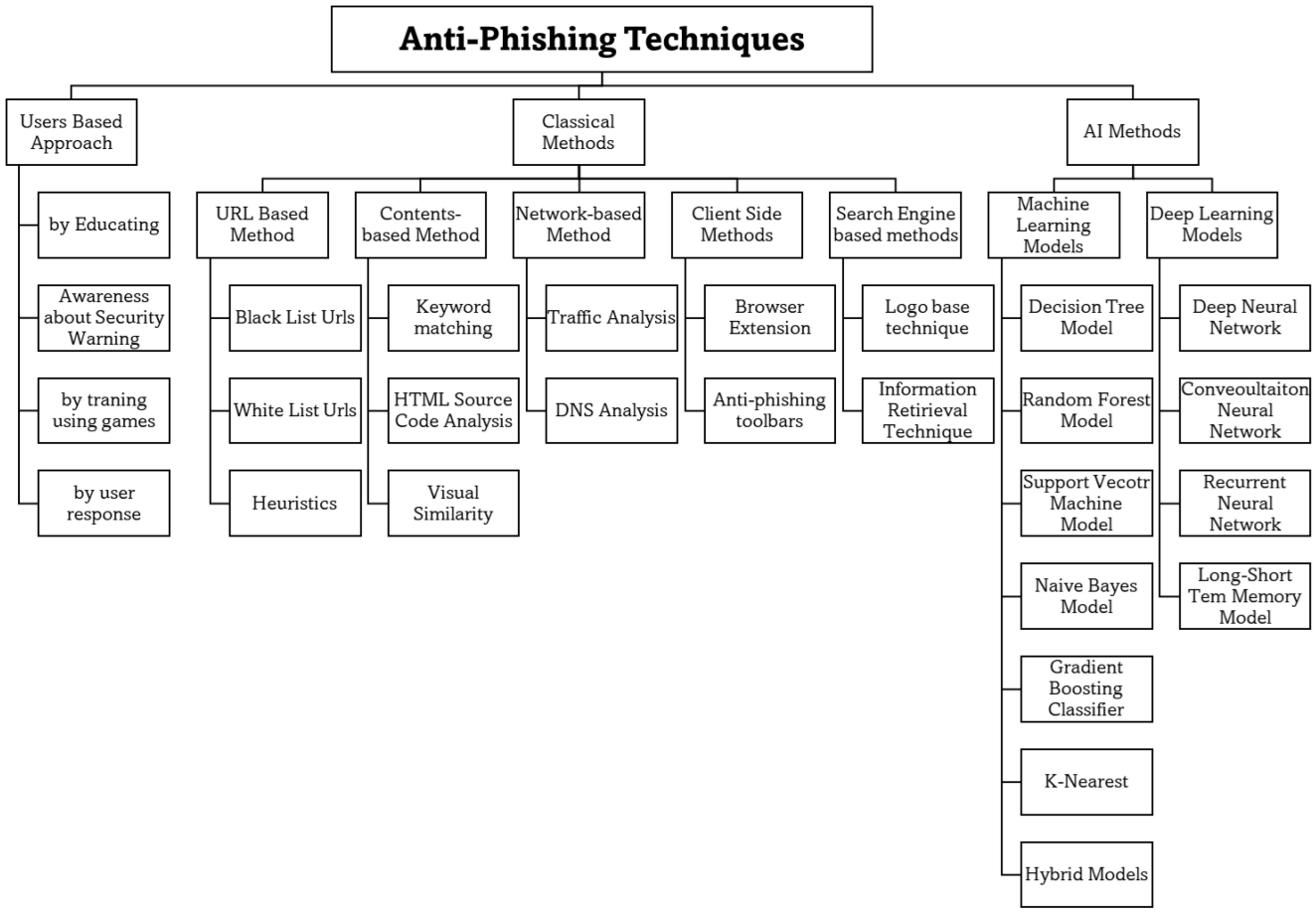

FIGURE 6. The diagram of classical and AI-based anti-phishing models categorizes various anti-phishing techniques into three main groups: user-based approaches, classical methods, and AI methods. It outlines specific strategies within each category, ranging from educating users and employing URL-based methods to advanced machine learning and deep learning models. This comprehensive framework highlights the multi-faceted approach necessary to effectively combat phishing threats.

users prefer to avoid verifying URLs. Researchers studied phishing attacks, specifically spear phishing, among 158 volunteers of various age groups. The study found that older women were more susceptible to phishing attacks compared to other demographics. Scarcity was more prevalent among young people, while reciprocation was more prevalent among elderly adults [85].

B. CLASSICAL METHODS

1) URL-BASED METHOD

1) Blocklist URLs: This method uses anti-phishing tools like Phish-Net, Google Safe Browsing, and PhishTank to generate a list of URLs. It works by matching suspected phishing URLs against the blocklist. If a match is found, the URL is identified as phishing; otherwise, it is considered legitimate. This method is beneficial for quickly identifying known phishing URLs but is not effective for new URLs. Therefore, the blocklist must be updated daily [143].

2) Whitelist URLs: This list contains only legitimate URLs with no associated phishing sources or code. All URLs that are not identified as phishing are added and managed in this list. When a URL is visited, it is compared with the whitelist to verify its legitimacy. If a URL is identified as phishing, it is marked for the blocklist. However, if it matches the whitelist, it is considered safe. The verification mechanism is often integrated within search engines or browser extensions like the Google toolbar [24].

3) Heuristics: Heuristics involve analyzing URL details such as domain name, domain path, website rank, Alexa ranking, and reputation score. A suspected phishing URL is tested against heuristic criteria. If the URL satisfies criteria such as domain name, path, address, and rank, it is considered legitimate. Otherwise, it is deemed phishing and added to the blocklist. Due to the short lifespan of phishing URLs, they rarely achieve high rankings [144].

Table 4 shows that the Blacklist method performs well, but there are some limitations associated with this approach. This method does not identify new phishing URLs, requiring the URL list to be updated daily. Similarly, whitelist and heuristic lists also need to be updated regularly. However, if all these methods are combined and implemented with automated list updates, they would provide the most effective and reliable solution for anti-phishing.

TABLE 4. The table provides a comparison between URL-based anti-phishing methods, detailing their limitations, proposed enhancements, and accuracy levels. It offers insights into how integrating these methods with advanced technologies like machine learning and AI can improve efficacy and reduce manual effort in identifying phishing threats.

Author	Method	Gaps	Solution	Accuracy
[145]	Blocklist URLs	Needs daily updates	Combine with other methods like whitelists and heuristics	High for known phishing sites
[24]	Whitelist URLs	Requires manual effort to maintain	Combine with automated tools and regular updates	High for whitelisted sites
[146]	Heuristics	Requires advanced analysis and expertise	Use machine learning and AI to improve accuracy and reduce false positives/negatives	Varies depending on the implementation

2) CONTENT-BASED METHODS

1) Keyword Matching: Keyword matching is another classical anti-phishing method used to detect potential phishing URLs. In this method, URLs are scanned for specific keywords commonly associated with phishing attacks, such as "username," "password," "login," "register," "CNIC," "credit card," "PIN code," "passcode," "email," "password," and "one-time password" [49]. If a URL string matches any of these keywords, it is flagged as phishing, and a pop-up alert warns the user to avoid visiting that URL [44].

2) HTML Source Code Analysis: HTML code is vital in anti-phishing efforts because it helps identify hidden patterns and modifications in the code. Several methods analyze and verify the contents of the source code. First, check if the form code, including input boxes or password fields, contains any hidden content or script that could redirect user input to a phisher's database.

Secondly, verify whether any SQL queries related to these fields are correctly executed or redirected. Next, examine JavaScript code to detect any malicious elements by comparing it with the original code and syntax. Additionally, verify internal and external links to ensure they refer to the correct domain, and validate security certificates with the SSL server. If all these parameters match the original page, it is declared legitimate; otherwise, it is identified as phishing [12].

3) Visual Similarity: Visual similarity matching involves analyzing CSS, text formatting, image formatting, image dimensions, and Web page content. These features are compared with authenticated and verified Web pages to determine if a page is phishing [82]. The visual similarity method is divided into four subtypes: document object comparison, CSS similarity, image similarity, and visual feature matching. The first model is trained using these features and tested in real-world scenarios. If a testing webpage matches all the features, it is marked as legitimate; otherwise, it is flagged as phishing [82].

TABLE 5. The table provides a comparison of different methods for analyzing HTML source code in anti-phishing solutions, highlighting their gaps, proposed solutions, accuracy, and overall suitability for various security needs. It underscores the trade-offs between ease of implementation and the level of security provided, suggesting combined approaches for enhanced protection.

Author	Method	Gaps	Solution	Accuracy	Decision
[44]	Keyword Matching	Limited detection capabilities, cannot identify complex attacks	Combine with other methods, use updated keyword lists	Moderate	Good for essential protection, but not suitable for high-security needs
[12]	HTML Source Code Analysis	May miss well-crafted visual phishing attempts	Combine with visual similarity and user education	High	Good for advanced protection, suitable for security-conscious users
[82]	Visual Similarity	It may be fooled by minor visual changes, which are ineffective against non-visual attacks.	Use advanced machine learning techniques combine with other methods	Very High	Best for high-security needs requires significant resources and expertise

Table 5 presents a comparative study of HTML code-based anti-phishing methods, showing that visual similarity is more efficient than HTML source code analysis and keyword matching. These detection methods can be enhanced by combining different approaches. If keyword matching, HTML source code analysis, and visual similarity methods are combined, the resulting output will be superior to other classical methods discussed in Table 4.

C. NETWORK-BASED METHOD

1) TRAFFIC ANALYSIS

The traffic analysis mechanism provides a comprehensive overview of network traffic to identify and control any suspicious activity immediately [147]. During phishing

detection using traffic analysis, various parameters such as IP addresses, domain names, packet sizes, encryption techniques, communication protocols, and phishing keywords are examined. Machine learning and deep learning models are trained using labeled data to classify phishing and legitimate traffic. However, this method may not provide 100% accuracy due to a high false positive rate. While it can help secure the system, more advanced methods are needed for precise traffic analysis [145].

2) DNS ANALYSIS

DNS anti-phishing involves verifying the domain name associated with an IP address to determine whether it is phishing or legitimate. If the results and records match, the URL or website is considered legitimate; otherwise, it is flagged as phishing [146].

D. CLIENT-SIDE METHODS

1) BROWSER EXTENSION

A browser-based solution proposed by [146] elaborates on how to be safe from phishing URLs. The author presented a new browser extension that operates in real-time, detecting phishing and legitimate URLs. The extension can block JavaScript and provide alerts for any phishing URLs.

2) ANTI-PHISHING TOOLBARS

Toolbar-based solutions are presented as browser extensions that must be installed to prevent phishing attacks. When users visit a phishing website, the toolbar assesses the website's credibility and warns users to avoid fraud.

E. SEARCH ENGINE-BASED METHODS

In this method, when a site is visited through a search engine, its page ranking in the search results is considered. The search engine indexes the website based on its lifespan and visit statistics. New websites typically do not rank at the top of search results. Search engine detection is classified into two methods:

1) LOGO-BASED TECHNIQUE

This is an older method to detect original URLs using a search engine. It involves extracting the logo of the original website and searching for it to find legitimate URLs.

2) INFORMATION RETRIEVAL TECHNIQUE

In this method, the search engine extracts features of a website, including Web page tags. The extracted tags are used in search queries to discover phishing websites. The Google Chrome browser uses a phishing detection extension to analyze website content, including domain names and Web page titles.

F. AI METHODS

AI plays a vital role in developing anti-phishing models, which perform well using different feature extraction methods to detect phishing websites. Several models have been developed and proposed as standards for phishing detection within the industry. However, due to the high rate of phishing attacks and new phishing techniques, these methods need continuous improvement. AI models are divided into two categories.

1) MACHINE LEARNING MODELS

1) Decision Tree Model: The Decision Tree is a supervised machine learning model known for its ease of use, parametric distribution, scoring distribution, and efficiency [148]. It can quickly learn and handle different types of data sets simultaneously [149]. This model operates iteratively to predict whether websites are phishing or legitimate [150].

$$E(s) = \sum_{i=1}^{C} -P_i \log_2 P_i \tag{1}$$

$$E(T, X) = \sum_{c \in X}^{C} p(c) \cdot E(c) \tag{2}$$

$$IG(T, X) = E(T) - E(T, X) \tag{3}$$

Equations (1), (2), and (3) are central to optimizing Decision Tree models for phishing detection. Entropy (E(s)) measures the impurity of a data set, conditional entropy (E(T, X)) evaluates the entropy after splitting by attribute X, and Information Gain (IG(T, X)) calculates the reduction in entropy due to the split. These metrics help to determine the most informative attributes for splitting nodes in the model.

2) Random Forest Model: The random forest (RF) is a classifier that is used to categorize data into different classes. It is a highly effective model that is often used to solve classification problems [151]. Like a decision tree, this model organizes data into various categories as given in Equation (4). It aggregates results from different nodes to predict classes [150], [152].

$$E(s) = \frac{1}{B} \sum_{i=0}^{B} f_i(x_t) \tag{4}$$

3) Support Vector Machine Model: Support Vector Machine (SVM) is a supervised machine learning model used for binary classification problems as given in equation (5), (6), (7) and (8). SVM solves linear problems using the kernel function, which eliminates the need to transform the data manually, as the SVM kernel handles this task. Additionally, there is no need to make assumptions about feature extraction or selection, as the SVM kernel manages these aspects. SVM provides a robust solution for classification problems by employing a nature-inspired optimization algorithm for phishing detection and spam identification. It classifies data into two classes using a

hyperplane, predicting the class based on the new value's position relative to the hyperplane [29].

$$w^T X = b = 0 \tag{5}$$

Equation (5): Linear hyperplane model

$$d_i = \frac{w^T x_i + b}{\|w\|} \tag{6}$$

Equation (6): The distance between a data point and the decision boundary

$$\hat{y} = \begin{cases} 1 \text{ if } w^T x + b \geq 0 \\ 0 \text{ if } w^T x + b < 0 \end{cases} \tag{7}$$

Equation (7): SVM with Linear Classifier Model

$$\begin{aligned} \text{maximize}_{(W,b)} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^{m} c_i \\ \text{subject to} \quad & y_i \left(w^T x + b \right) \geq 1 - c_i \\ & c_i \geq 0 \quad \text{for } i = 1, 2, 3, \ldots, m \end{aligned} \tag{8}$$

Equation (8): Linear SVM Classifier with Soft Margin

4) Naive Bayes Model: Naïve Bayes is a probabilistic model used as a classifier. It is based on the Bayesian model as given in equation (9) and is used to find relationships between different features. This model calculates the probability of feature occurrence using the corpus consideration method. As a supervised machine learning model, Naïve Bayes works with class label attributes to classify data. The correlation of all attributes is calculated independently, assuming that each feature contributes equally and independently to the outcome [45].

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \tag{9}$$

Equation (9): Naive Bayes Theorem

5) AdaBoost: AdaBoost is a classifier that functions similarly to a Random Forest classifier. It uses multiple weak learners and combines them into strong learners to classify data using which is based on equation (10). This model utilizes the weights of the nodes during training, transferring these weights to subsequent nodes to enhance the model's accuracy. As a supervised machine learning model, AdaBoost uses labeled data for classification. During training, the model first generates a weak tree structure, assigns scores to weak learners, and then transfers weights to the next learner, making it more robust than the previous one. In essence, the model learns from previous errors to improve its accuracy over time. The number of iterations contributes to improved accuracy [30], [31].

$$H(x) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m h_m(x) \right) \tag{10}$$

Equation. (10): AdaBoost Mathematical Model

6) Gradient Boosting Classifier: Gradient Boosting is a supervised machine learning classifier given in equation (11) that works similarly to AdaBoost. It assigns weights to the training data and learns from errors. During training, the model iteratively learns by minimizing errors and improving accuracy in subsequent iterations [28], [30], [44].

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x) \right) \tag{11}$$

Equation. (11): Gradient Boosting Mathematical Model

7) XGBoost: XGBoost is an advanced form of the Gradient Boosting algorithm that outperforms other boosting classifiers. It is highly scalable and effective for various classification problems using equations (12) and (13). XGBoost provides faster processing due to its efficient memory management and distributed processing capabilities, enabling data scientists to run large datasets on desktop processors [23].

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x) \right) \tag{12}$$

$$\text{Obj} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_t\right) + \sum_{k=1}^{K} \Omega(f_k) \tag{13}$$

8) K-Nearest Neighbor: The k-Nearest Neighbors (k-NN) algorithm is a supervised machine learning model used to solve classification and regression problems using equations (14) and (15). Classifies data points according to their similarity, measured by the Euclidean distance, where the value of k is a positive integer representing the number of nearest neighbors considered. The model calculates the nearest-neighbor score to find the closest data points and groups them into one class. Although it is simple to apply, it can be computationally expensive due to assumptions such as equally divided classes [23].

$$\hat{y}_q = \text{argmax}_y \sum_{i=1}^{N} 1(y_i = y) \cdot 1\left(x_i \in \text{KNN}(x_q)\right) \tag{14}$$

$$\hat{y}_q = \frac{1}{k} \sum_{i=1}^{N} (y_i) \cdot 1\left(x_i \in \text{KNN}(x_q)\right) \tag{15}$$

9) Logistic Regression: The logistic regression algorithm is a supervised machine learning algorithm used for classification and regression modeling as given in equations (16) and (17). It utilizes the sigmoid function to calculate the probability score of two classes and map them accordingly. Logistic regression performs well when the relationship between the classes in the dataset is linear, but its performance declines significantly in non-linear cases.

$$P(y_i = 1 \mid x_i) = \frac{1}{\left(1 + e^{-z}\right)} \tag{16}$$

$$\sigma(z) = \frac{1}{(1 + e^{z})} \tag{17}$$

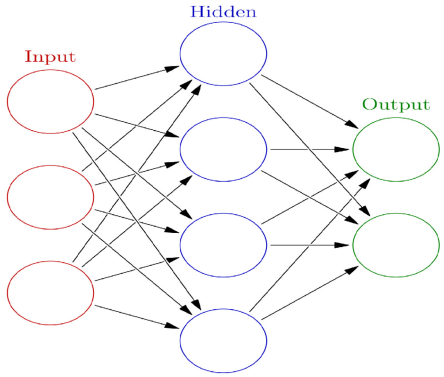

FIGURE 7. The Artificial Neural Network Layers Model diagram illustrates a basic neural network structure, comprising input, hidden, and output layers. Nodes in the input layer receive data, which then flows through interconnected nodes in the hidden layer before reaching the output layer for final processing. Arrows represent the connections through which data is forwarded, with weights that are adjusted during training to optimize the network's predictive accuracy.

2) HYBRID MODELS

1) Artificial Neural Network: An Artificial Neural Network (ANN) is a deep learning model inspired by the biological neural network of the human brain as shown in Figure 7. It consists of multiple layers, each containing numerous neurons that process data. The neurons in each layer receive input from the previous layer, process it using assigned weights and activation functions, and then pass the output to the next layer. There are two types of processing within a neural network: feed-forward and feedback. In the feed-forward process, data flow from left to right through the layers, while in feedback (or back-propagation), data flow from right to left to adjust weights based on the error between predicted and actual outcomes. The number of layers in an ANN can be customized according to the dataset and model requirements, and each layer independently transforms its data for the subsequent layer. During the initial stage of model training, nodes receive random weights, which are then adjusted using a gradient descent algorithm to achieve optimal solutions. Due to their ability to handle non-linear data, neural networks are effective for addressing complex problems.

3) DEEP LEARNING MODELS

1) Deep Neural Network (DNN): A Deep Neural Network (DNN) is an advanced form of an Artificial Neural Network (ANN). The primary distinction of DNN is the increase in the number of hidden layers. DNNs contain more hidden layers than typical ANNs, allowing for more complex data processing. The layers are arranged in sequence, starting with the input layer that receives data. The hidden layers process this data using activation functions such as sigmoid functions, employing feed-forward and feedback methods. The final layer, the output layer, provides the results of classification or detection tasks. DNN is a supervised machine learning model used for classification and detection problems [153].

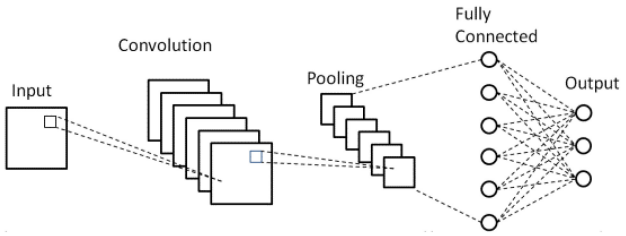

FIGURE 8. The Convolutional Neural Network Layers Structure diagram represents the architecture of a typical Convolutional Neural Network (CNN). It begins with an input layer and progresses through multiple convolutional layers where filters are applied to extract features. These are followed by pooling layers that reduce dimensionality. The architecture culminates in a fully connected layer that leads to the output.

2) Convolutional Neural Network (CNN): A Convolutional Neural Network (CNN) is a deep learning algorithm that operates using layered architecture listed in equation (18), (19), (20) and (21). The first layer is the convolutional layer, which applies convolution operations to extract features from the input data. The second layer is the fully connected layer, which is typically used for classification tasks. CNNs are primarily used in computer vision for image processing and work with two-dimensional data. Additional layers in CNNs include pooling layers for down-sampling, dropout layers to prevent overfitting, batch normalization layers to stabilize learning, and output layers for final predictions [154]. The basic structure of CNN model is given in Figure 8.

$$(f * g)(t) = \sum_{a=-\infty}^{\infty} f(a)g(t-a) \tag{18}$$

$$\text{ReLU}(x) = \max(0, x) \tag{19}$$

$$\text{max-pooling}(x) = \max(\text{neighborhood}(x)) = Wx + b \tag{20}$$

$$L(y, \hat{y}) = -\sum_{i} y_i \log \sum \left(\hat{y}^i\right) \tag{21}$$

Equation (15): Convolution Neural Network Model

3) Recurrent Neural Network: Recurrent Neural Network (RNN) is a deep learning model for language processing and text mining. In this model given in equation (22), (23), and (24), each layer is connected with the inner layer, generating a bi-directional connection. This model helps process sequential data and is helpful for URL phishing URL detection and feature extraction [155]. Figure 9 is showing basic structure of RNN model.

$$h_t = \text{activation}(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \tag{22}$$

$$y_t = \text{softmax}\left(W_{yh}h_t + b_y\right) \tag{23}$$

$$L(y, \hat{y}) = -\sum_{i} y_i \log \sum \left(\hat{y}^i\right) \tag{24}$$

Equation (16): RNN Mathematical Model with Loss Function

4) Long-Short-Term Memory Model: Long-Short-Term Memory (LSTM) is an advanced form of RNN. The basic

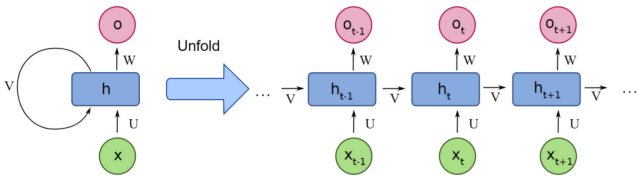

FIGURE 9. Basic Architecture of RNN Model diagram illustrates the unfolding of a Recurrent Neural Network (RNN) over time. It shows how input xxx at different time steps t is processed through the same hidden layer h with recurrent connections, and how it influences the output o at each step. The unfolding visualizes the sequence processing nature of RNNs, where weights U, V, and W are shared across all time steps, highlighting the network's ability to maintain temporal dependencies in data.

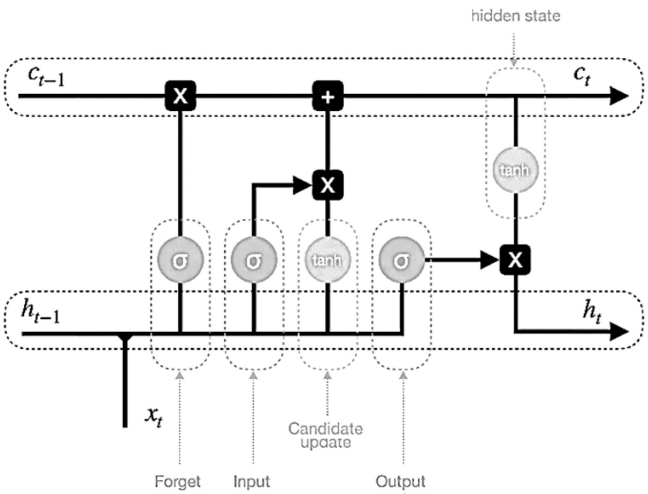

FIGURE 10. LSTM Model Architecture diagram illustrates the architecture of an LSTM cell, highlighting the processing of inputs through forget, input, and output gates to update the cell state and hidden state, thereby managing long-term dependencies in data sequences.

RNN did not support more than ten inner connections of layers. However, with the new LSTM method, it is possible to implement more than ten inner connected layers in the model as shown in figure 10. LSTM enables the application of more than 1000 processing steps by establishing the relationship of data points with each other. During processing, LSTM finds the importance scores of each step, if necessary, and keeps it within memory until the model is executed [153], [154].

IX. MODEL RECOMMENDATION FOR PHISHING DETECTION

Based on the literature study and visualization of different models using accuracy metrics as given in Figure 11, it is concluded that the **Random Forest** model performs best in classifying legitimate and manipulated websites. This model excels both in individual implementation and when used in ensemble and hybrid modeling techniques.

X. OPEN CHALLENGES & DISCUSSION

Several models have been developed for detecting Web phishing, but due to the evolving nature of phishing attacks, many challenges persist.

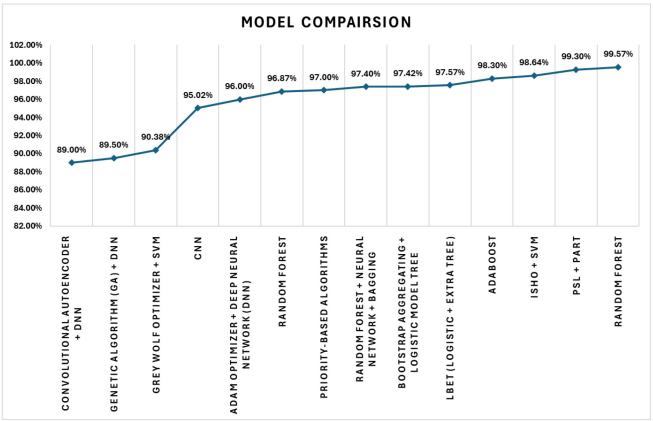

FIGURE 11. Performance Comparison of Various Phishing Detection Models: This graph illustrates the accuracy rates of different models used in phishing detection, ranging from Convolutional Autoencoder + DNN to Random Forest, highlighting the progressive improvement in detection capabilities across models.

1) Adapting to Emerging Phishing Techniques: Phishing websites are launched daily with new phishing material, making it essential to regularly update phishing dataset databases and train models with the latest data [1], [2]. This challenge is significant as existing models may become outdated quickly without frequent updates [26].
2) Dataset-Specific Model Limitations: Many models, including hybrid ones, are trained on specific datasets with particular features, which can limit their effectiveness against emerging attacks [3], [45]. The specificity of datasets poses a challenge in developing models that are versatile across different types of phishing attacks [36].
3) Feature Extraction Gaps: Current models often extract features based on fixed thresholds or statistical models, lacking a dynamic approach that considers comprehensive information such as Web page content, URL properties, visualization properties, and domain properties [17], [36], [41]. More extensive work is needed in feature extraction to enhance phishing detection research [108].
4) Autonomous System Development: Developing an autonomous system that continuously updates itself with the latest phishing attacks and maintains an up-to-date database of phishing URLs is a significant challenge [18], [24]. The need for real-time environments that automatically learn and train themselves to ensure user safety is critical [36].
5) User Awareness and Education: Despite technical advancements, many users remain unaware of phishing attacks, including educated individuals who might inadvertently click on malicious URLs [71]. Conducting training sessions and awareness campaigns is crucial to educate people about phishing threats and prevention strategies [109].

TABLE 6. Table provides a comparative analysis of various phishing detection models, detailing their method types, datasets, main challenges, limitations, and model accuracy. It highlights the performance metrics and constraints of each model, showcasing how they fare against different types of phishing datasets and under varying conditions.

Model/Ref	Method-Type	Dataset-Ref	Main Challenges	Model Limitations	Model Accuracy
Random Forest [43]	Single Model	ISCX-2016	Enhance accuracy using limited features without any third-party tool	This model was not tested with multiple datasets. The model did not support robustness.	0.9957
Random Forest [34]	Single Model	Phishing Tank	Manually extracted features rely on third-party services	Small experimental dataset	0.9950
PSL + PART [42]	Hybrid Model	Phishing Tank Bank Dataset	Trained different ML models and compared results	The dataset was based only on bank data.	0.9930
ISHO + SVM [40]	Hybrid Model	UCI Dataset	Dataset lacks original URLs, no feature extraction	Limited to specific datasets without URL information	0.9864
Adaboost [41] [69]	Single Model	UCI Dataset	Not evaluated on diverse datasets	Not enough information on dataset size	0.9830
LBET (Logistic + Extra Tree) [69]	Hybrid Model	UCI Dataset	Insufficient data sources, lack of feature extraction	Limited to specific datasets without URL information	0.9757
Bootstrap Aggregating + Logistic Model Tree [156]	Hybrid Model	UCI Dataset	Insufficient data sources, lack of feature extraction	Limited to specific datasets without URL information	0.9742
Random Forest + Neural Network + Bagging [45]	Hybrid Model	UCI Dataset	No prior research on this specific combination	Insufficient data sources, lack of feature extraction	0.9740
Priority-Based Algorithms [59]	Hybrid Model	UCI Dataset	Insufficient data sources, lack of feature extraction	Limited to specific datasets without URL information	0.9700
Random Forest [36]	Single Model	Phishing Tank	Insufficient data sources, lack of feature extraction	Limited to specific datasets without URL information	0.9687
Adam optimizer + Deep Neural Network (DNN) [60]	Hybrid Model	Phishing Tank	Insufficient data sources, lack of feature extraction	Limited to specific datasets without URL information	0.9600
CNN [29], [124]	Single Model	Phishing Tank	Large dataset size, long training time	Sensitive to URL length ignores website status	0.9502
Grey Wolf Optimizer + SVM [29]	Hybrid Model	Phishing Tank	Small dataset, limited evaluation on diverse datasets	Rule-based features only, potentially lacking complexity	0.9038
Genetic Algorithm (GA) + DNN [157]	Hybrid Model	Phishing Tank	Requires longer training due to GA feature selection	Insufficient data sources, lack of feature extraction	0.8950
Convolutional Autoencoder + DNN [158]	Hybrid Model	Phishing Tank	Lower accuracy than some methods, small dataset	Rule-based features, potentially limited information	0.8900

6) Dependency on Third-Party Services for Feature Extraction: Most feature extraction methods rely on third-party services to extract specific information such as domain names, DNS registration, and host names. These services are often paid and may not always provide up-to-date information, leading to higher error rates in models [24].
7) Handling Tiny URLs: The literature lacks specific mechanisms for handling tiny URLs, which are difficult to track and verify for phishing content. Educating

users about the credibility of URLs, whether from known or unknown sources, is crucial [12], [35]. Designing a system to identify whether tiny URLs are phished or legitimate remains a significant research challenge [125].

8) Limitations of Rule-Based and List-Based Models: Rule-based and list-based models, while effective in identifying phished or legitimate URLs, require frequent updates and can have slow detection speeds, resulting in high response times [146]. Designing systems capable of handling phishing links spread through various devices presents a substantial challenge [127].

XI. CONCLUSION

In conclusion, datasets with few phishing URLs can adversely affect model performance when tested on larger datasets. Although blacklisting, whitelisting, and rule-based detection methods are effective, they are constrained by lists or rules. Machine learning models have been introduced in some research, but these are limited to specific features. Even when using NLP-based machine learning models for feature extraction, third-party services are still necessary. A comprehensive and dynamic system capable of handling all types of attack, implementing across all devices, and updating dynamically with new phishing techniques is still needed. Continued research is necessary to develop benchmark datasets and systems for both offline and real-time detection.

ACKNOWLEDGMENT

The open access funding is provided by Qatar National Library. Besides, the authors also extend their appreciation for the necessary support of Al-Ahliyya Amman University, Jordan.

REFERENCES

[1] D. Kalla, F. Samaah, S. Kuraku, and N. Smith, "Phishing detection implementation using databricks and artificial intelligence," *Int. J. Comput. Appl.*, vol. 185, no. 11, pp. 1–11, 2023.

[2] A. Safi and S. Singh, "A systematic literature review on phishing Website detection techniques," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 35, no. 2, pp. 590–611, 2023.

[3] B. Naqvi, K. Perova, A. Farooq, I. Makhdoom, S. Oyedeji, and J. Porras, "Mitigation strategies against the phishing attacks: A systematic literature review," *Comput. Security*, vol. 132, Sep. 2023, Art. no. 103387.

[4] M. K. Pandey, M. K. Singh, S. Pal, and B. Tiwari, "Prediction of phishing websites using machine learning," *Spatial Inf. Res.*, vol. 31, no. 2, pp. 157–166, 2023.

[5] C. Cross, "'I knew it was a scam': Understanding the triggers for recognizing romance fraud," *Criminol. Public Policy*, vol. 22, no. 4, pp. 613–637, 2023.

[6] L. Brotherston, A. Berlin, and W. F. Reyor, III, *Defensive Security Handbook*. Beijing, China: O'Reilly Media, 2024.

[7] T. Xu, K. Singh, and P. Rajivan, "Personalized persuasion: Quantifying susceptibility to information exploitation in spear-phishing attacks," *Appl. Ergonom.*, vol. 108, Apr. 2023, Art. no. 103908.

[8] M. Nadeem, S. W. Zahra, M. N. Abbasi, A. Arshad, S. Riaz, and W. Ahmed, "Phishing attack, its detections and prevention techniques," *Int. J. Wireless Security Netw.*, vol. 1, no. 2, pp. 13–25, 2023.

[9] R. Goenka, M. Chawla, and N. Tiwari, "A comprehensive survey of phishing: Mediums, intended targets, attack and defence techniques and a novel taxonomy," *Int. J. Inf. Security*, vol. 23, no. 2, pp. 819–848, 2024.

[10] I. Ahmad, S. Khan, and S. Iqbal, "Guardians of the vault: Unmasking online threats and fortifying e-banking security, a systematic review," *J. Financ. Crime*, to be published.

[11] F. S. Alsubaei, A. A. Almazroi, and N. Ayub, "Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics," *IEEE Access*, vol. 12, pp. 8373–8389, 2024.

[12] S. Asiri, Y. Xiao, S. Alzahrani, S. Li, and T. Li, "A survey of intelligent detection designs of HTML URL phishing attacks," *IEEE Access*, vol. 11, pp. 6421–6443, 2023.

[13] Y. Guo, "A review of machine learning-based zero-day attack detection: Challenges and future directions," *Comput. Commun.*, vol. 198, pp. 175–185, Jan. 2023.

[14] A. S. Albahri et al., "A systematic review of trustworthy and explainable artificial intelligence in healthcare: Assessment of quality, bias risk, and data fusion," *Inf. Fusion*, vol. 96, pp. 156–191, Aug. 2023.

[15] S. Kemp, "Global overview report," 2022. [Online]. Available: https://datareportal.com/Reports/Digit.-2022-Global-Overview-Rep

[16] "APWG phishing trends report," APWG. Sep. 2022. [Online]. Available: https://apwg.org/trendsreports/

[17] B. Gontla, P. Gundu, P. Uppalapati, K. Rao, and S. Hussain, "A machine learning approach to identify phishing websites: A comparative study of classification models and ensemble learning techniques," *EAI Endorsed Trans. Scalable Inf. Syst.*, vol. 10, no. 5, p. 9, 2023.

[18] S. Santos, P. Costa, and A. Rocha, "IT/OT convergence in industry 4.0: Risks and analysis of the problems," in *Proc. Iberian Conf. Inf. Syst. Technol.*, 2023, pp. 1–8.

[19] N. Do, A. Selamat, O. Krejcar, E. Herrera-Viedma, and H. Fujita, "Deep learning for phishing detection: Taxonomy, current challenges, and future directions," *IEEE Access*, vol. 10, pp. 36429–36463, 2022.

[20] U. Agarwal et al., "Blockchain technology for secure supply chain management: A comprehensive review," *IEEE Access*, vol. 10, pp. 85493–85517, 2022.

[21] S. Zahra, M. Chishti, A. Baba, and F. Wu, "Detecting Covid-19 chaos driven phishing/malicious URL attacks by a fuzzy logic and data mining based intelligence system," *Egypt. Inform. J.*, vol. 23, no. 2, pp. 197–214, 2022.

[22] A. Jain, S. Sahoo, and J. Kaubiyal, "Online social networks security and privacy: Comprehensive review and analysis," *Complex Intell. Syst.*, vol. 7, no. 5, pp. 2157–2177, 2021.

[23] S. Abad, H. Gholamy, and M. Aslani, "Classification of malicious URLs using machine learning," *Sensors*, vol. 23, no. 18, p. 7760, 2023.

[24] I. Kotenko, "Detection of anomalies and attacks in container systems: An integrated approach based on black and white lists," in *Intelligent Information Technologies for Industry* (Lecture Notes in Networks and Systems). Cham, Switzerland: Springer, 2023.

[25] T. Pattewar, C. Mali, S. Kshire, M. Sadarao, J. Salunkhe, and A. Shah, "Malicious short URLs detection: A survey," *Int. Res. J. Eng. Technol.*, vol. 6, no. 11, pp. 1–7, 2019.

[26] O. Abiodun, A. S. Sodiya, and S. O. Kareem, "Linkcalculator—An efficient link-based phishing detection tool," *Acta Informatica Malaysia*, vol. 4, no. 2, pp. 37–44, 2020.

[27] P. Yang, G. Zhao, and P. Zeng, "Phishing Website detection based on multidimensional features driven by deep learning," *IEEE Access*, vol. 7, pp. 15196–15209, 2019.

[28] L. Tang and Q. Mahmoud, "A survey of machine learning-based solutions for phishing Website detection," *Mach. Learn. Knowl. Extraction*, vol. 3, no. 3, pp. 672–694, 2021.

[29] S. Anupam and A. Kar, "Phishing Website detection using support vector machines and nature-inspired optimization algorithms," *Telecommun. Syst.*, vol. 76, no. 1, pp. 17–32, 2021.

[30] V. Shahrivari, M. Darabi, and M. Izadi, "Phishing detection using machine learning techniques," 2020, *arXiv:2009.11116*.

[31] J. Rashid, T. Mahmood, M. Nisar, and T. Nazir, "Phishing detection using machine learning technique," in *Proc. 1st Int. Conf. Smart Syst. Emerg. Technol.*, 2020, pp. 43–46.

[32] S. Alnemari and M. Alshammari, "Detecting phishing domains using machine learning," *Appl. Sci.*, vol. 13, no. 8, p. 4649, 2023.

[33] A. Dutta, "Detecting phishing websites using machine learning technique," *PLoS ONE*, vol. 16, no. 10, 2021, Art. no. e0258361.

[34] E. Gandotra and D. Gupta, "Improving spoofed Website detection using machine learning," *Cybern. Syst.*, vol. 52, no. 2, pp. 169–190, 2021.

[35] B. Waseso and N. Setiyanto, "Web phishing classification using combined machine learning methods," *J. Comput. Theor. Appl.*, vol. 1, no. 1, pp. 11–18, 2023.

[36] G. Lokesh and G. BoreGowda, "Phishing Website detection based on effective machine learning approach," *J. Cyber Security Technol.*, vol. 5, no. 1, pp. 1–14, 2021.

[37] M. Lei, Y. Xiao, S. Vrbsky, and C. Li, "Virtual password using random linear functions for on-line services, ATM machines, and pervasive computing," *Comput. Commun.*, vol. 31, no. 18, pp. 4367–4375, 2008.

[38] A. Basit, M. Zafar, X. Liu, A. Javed, Z. Jalil, and K. Kifayat, "A comprehensive survey of AI-enabled phishing attacks detection techniques," *Telecommun. Syst.*, vol. 76, no. 1, pp. 139–154, 2021.

[39] U. Butt, R. Amin, H. Aldabbas, S. Mohan, B. Alouffi, and A. Ahmadian, "Cloud-based email phishing attack using machine and deep learning algorithm," *Complex Intell. Syst.*, vol. 9, no. 3, pp. 3043–3070, 2023.

[40] M. Sabahno and F. Safara, "ISHO: Improved spotted hyena optimization algorithm for phishing website detection," *Multimed. Tools Appl.*, vol. 81, no. 24, pp. 34677–34696, 2022-10.

[41] A. Odeh, I. Keshta, and E. Abdelfattah, "PhiBoost—A novel phishing detection model using adaptive boosting approach," *Jordanian J. Comput. Inf. Technol.*, vol. 7, no. 1, pp. 64–73, Mar. 2021.

[42] P. Barraclough, G. Fehringer, and J. Woodward, "Intelligent cyber-phishing detection for online," *Comput. Secur.*, vol. 104, May 2021, Art. no. 102123.

[43] B. Gupta, K. Yadav, I. Razzak, K. Psannis, A. Castiglione, and X. Chang, "A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment," *Comput Commun*, vol. 175, pp. 47–57, Jul. 2021.

[44] C. Singh and Meenu, "Phishing Website detection based on machine learning: A survey," in *Proc. 6th Int. Conf. Adv. Comput. Commun. Syst.*, 2020, pp. 398–404.

[45] A. Zamir, "Phishing Web site detection using diverse machine learning algorithms," *Electron. Library*, vol. 38, no. 1, pp. 65–80, Mar. 2020.

[46] M. Vijayalakshmi, S. Shalinie, M. Yang, and U. Meenakshi, "Web phishing detection techniques: A survey on the state-of-the-art, taxonomy and future directions," *IET Netw.*, vol. 9, no. 5, pp. 235–246, 2020.

[47] A. Jain and B. Gupta, "A survey of phishing attack techniques, defence mechanisms and open research challenges," *Enterprise Inf. Syst.*, vol. 16, no. 4, pp. 527–565, 2022.

[48] M. Bhattacharya, S. Roy, S. Chattopadhyay, A. Das, and S. Shetty, "A comprehensive survey on online social networks security and privacy issues: Threats, machine learning-based solutions, and open challenges," *Security Privacy*, vol. 6, no. 1, p. e275, 2023.

[49] S. Samad, "Analysis of the performance impact of fine-tuned machine learning model for phishing URL detection," *Electronics*, vol. 12, no. 7, p. 1642, 2023.

[50] M. Almousa, T. Zhang, A. Sarrafzadeh, and M. Anwar, "Phishing Website detection: How effective are deep learning-based models and hyperparameter optimization?" *Security Privacy*, vol. 5, no. 6, p. e256, 2022.

[51] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing email detection using natural language processing techniques: A literature survey," *Procedia CIRP*, vol. 189, pp. 19–28, Jul. 2021.

[52] M. Korkmaz, O. Sahingoz, and B. Diri, "Feature selections for the classification of Webpages to detect phishing attacks: A survey," in *Proc. 2nd Int. Congr. Human–Comput. Interact., Optim. Robot. Appl.*, 2020.

[53] H. R. M. Gowda, M. V. Adithya, S. G. Prasad, and S. Vinay, "Development of anti-phishing browser based on random forest and rule of extraction framework," *Cybersecurity*, vol. 3, no. 1, p. 20, 2020.

[54] P. Kumar, T. Jaya, and V. Rajendran, "SI-BBA—A novel phishing Website detection based on swarm intelligence with deep learning," *Mater. Today Proc.*, vol. 80, pp. 3129–3139, Apr. 2023.

[55] L. Abdulrahman, S. Ahmed, Z. Rashid, Y. Jghef, T. Ghazi, and U. Jader, "Web phishing detection using Web crawling, cloud infrastructure and deep learning framework," *J. Appl. Sci. Technol. Trends*, vol. 4, no. 1, pp. 54–71, 2023.

[56] P. Kalaharsha and B. Mehtre, "Detecting phishing sites—An overview," Mar. 2021, *arXiv:2103.12739*.

[57] R. Pravali, S. Raha, Y. Rachana, and D. Kamesh, "Ensemble machine learning model for phishing intrusion detection and classification from URLs," 2023.

[58] L. Jovanovic, "Improving phishing Website detection using a hybrid two-level framework for feature selection and XGBoost tuning," *J. Web Eng.*, vol. 22, no. 3, pp. 543–574, 2023-07.

[59] A. Lakshmanarao, P. P. Rao, and M. Krishna, "Phishing website detection using novel machine learning fusion approach," in *Proc. Int. Conf. Artif. Intell. Smart Syst.*, 2021, pp. 1164–1169.

[60] L. Lakshmi, M. Reddy, C. Santhaiah, and U. Reddy, "Smart phishing detection in Web pages using supervised deep learning classification and optimization technique ADAM," *Wireless Pers. Commun.*, vol. 118, no. 4, pp. 3549–3564, 2021-06.

[61] A. Karim, M. Shahroz, K. Mustofa, S. Belhaouari, and S. Joga, "Phishing detection system through hybrid machine learning based on URL," *IEEE Access*, vol. 11, pp. 36805–36822, 2023.

[62] R. Zieni, L. Massari, and M. Calzarossa, "Phishing or not phishing? A survey on the detection of phishing websites," *IEEE Access*, vol. 11, pp. 18499–18519, 2023.

[63] M. Shaukat, R. Amin, M. Muslam, A. Alshehri, and J. Xie, "A hybrid approach for alluring ads phishing attack detection using machine learning," *Sensors*, vol. 23, no. 19, p. 8070, 2023.

[64] L. Yang, J. Zhang, X. Wang, Z. Li, Z. Li, and Y. He, "An improved ELM-based and data preprocessing integrated approach for phishing detection considering comprehensive features," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113863.

[65] V. Praveena et al., "Optimal deep reinforcement learning for intrusion detection in UAVs," *Comput., Mater. Continua*, vol. 70, no. 2, pp. 2639–2653, 2022.

[66] P. Chinnasamy, K. S. Sathya, B. J. A. Jebamani, A. Nithyasri, and S. Fowjiya, "Deep learning: Algorithms, techniques, and applications—A systematic survey," in *Deep Learning Research Applications for Natural Language Processing*. Hershey, PA, USA: IGI Global, 2023, pp. 1–17.

[67] M. Korkmaz, O. Sahingoz, and B. Diri, "Detection of phishing websites by using machine learning-based URL analysis," in *Proc. 11th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Kharagpur, India, 2020, pp. 1–7.

[68] M. Adebowale, K. Lwin, and M. Hossain, "Intelligent phishing detection scheme using deep learning algorithms," *J. Enterprise Inf. Manage.*, vol. 36, no. 3, pp. 747–766, 2023.

[69] Y. Alsariera, V. Adeyemo, A. Balogun, and A. Alazzawi, "AI meta-learners and extra-trees algorithm for the detection of phishing websites," *IEEE Access*, vol. 8, pp. 142532–142542, 2020.

[70] A. Maci, A. Santorsola, A. Coscia, and A. Iannacone, "Unbalanced Web phishing classification through deep reinforcement learning," *Computers*, vol. 12, no. 6, p. 118, 2023.

[71] L. Gallo, D. Gentile, S. Ruggiero, A. Botta, and G. Ventre, "The human factor in phishing: Collecting and analyzing user behavior when reading emails," *Comput. Security*, vol. 139, Apr. 2024, Art. no. 103671.

[72] R. Paudel and M. N. Al-Ameen, "Priming through persuasion: Towards secure password behavior," *Proc. ACM Human-Comput. Interact.*, vol. 8, no. CSCW1, pp. 1–27, 2024.

[73] H. Gururaj, V. Janhavi, and V. Ambika, *Social Engineering in Cybersecurity: Threats and Defenses*. Boca Raton, FL, USA: CRC Press, 2024.

[74] A. Juanna, M. A. S. Monoarfa, R. Podungge, and R. Tantawi, "Identification of trends in business promotion and marketing using video-based content on social media," *Jambura Sci. of Manage.*, vol. 6, no. 2, pp. 88–103, 2024.

[75] N. Akyeşilmen and A. Alhosban, "Non-technical cyber-attacks and international cybersecurity: The case of social engineering," *Gaziantep Univ. J. Social Sciences*, vol. 23, no. 1, pp. 342–360, 2024.

[76] D. Senecal, *The Reign of Botnets: Defending Against Abuses, Bots and Fraud on the Internet*. Hoboken, NJ, USA: Wiley, 2024.

[77] K. Church and R. De Oliveira, "What's up with WhatsApp? Comparing mobile instant messaging behaviors with traditional SMS," in *Proc. 15th Int. Conf. Human–Comput. Interact. Mobile Devices Services*, 2013, pp. 352–361.

[78] Z. Alkhalil, C. Hewage, L. Nawaf, and I. Khan, "Phishing attacks: A recent comprehensive study and a new anatomy," *Front. Comput. Sci.*, vol. 3, Mar. 2021, Art. no. 563060.

[79] P. Syiemlieh, G. M. Khongsit, U. M. Sharma, and B. Sharma, "Phishing-an analysis on the types, causes, preventive measuresand case studies in the current situation," *IOSR J. Comput. Eng*, vol. 9, pp. 2278–8727, Jan. 2015.

[80] W. Kim, O.-R. Jeong, C. Kim, and J. So, "The dark side of the Internet: Attacks, costs and responses," *Inf. Syst.*, vol. 36, no. 3, pp. 675–705, 2011.

[81] N. Knopf, "Social engineering: How crowdmasters, phreaks, hackers, and trolls created a new form of manipulative communication—Robert W. Gehl and Sean T. Lawson (Cambridge, MA, USA: MIT Press, 2022, 344 p.)," *IEEE Technol. Soc. Mag.*, vol. 42, no. 1, pp. 13–15, Mar. 2023.

[82] B. Dooremaal, P. Burda, L. Allodi, and N. Zannone, "Combining text and visual features to improve the identification of cloned Webpages for early phishing detection," in *Proc. ACM Int. Conf. Availability, Reliability and Security*, 2021, p. 60.

[83] I. Tomicic, "Social engineering aspects of email phishing: An overview and taxonomy," in *Proc. 46th MIPRO ICT Electron. Conv.*, 2023, pp. 1201–1207.

[84] S. Dadvandipour and A. Ganie, "Analyzing and predicting spear-phishing using machine learning methods," *Multidiszciplináris Tudományok*, vol. 10, no. 4, pp. 262–273, 2020.

[85] H. Oz, A. Aris, A. Levi, and A. Uluagac, "A survey on ransomware: Evolution, taxonomy, and defense solutions," *ACM Comput. Surveys*, vol. 54, no. 11S, p. 238, 2022.

[86] Ekta and U. Bansal, "A review on ransomware attack," in *Proc. Int. Conf. Secure Cyber Comput. Commun.*, 2021, pp. 221–226.

[87] S. Ullah, T. Ahmad, A. Buriro, N. Zara, and S. Saha, "TrojanDetector: A multi-layer hybrid approach for trojan detection in android applications," *Appl. Sci.*, vol. 12, no. 21, 2022, Art. no. 10755.

[88] I. Riadi, Sunardi, and D. Aprilliansyah, "Analysis of Anubis trojan attack on android banking application using mobile security Labware," *Int. J. Saf. Security Eng.*, vol. 13, no. 1, pp. 31–38, 2023.

[89] A. Hannousse, S. Yahiouche, and M. C. Nait-Hamoud, "Twenty-two years since revealing cross-site scripting attacks: A systematic mapping and a comprehensive survey," *Comput. Sci. Rev.*, vol. 52, May 2024, Art. no. 100634.

[90] F. Kalantari, M. Zaeifi, T. Bao, R. Wang, Y. Shoshitaishvili, and A. Doupé, "Context-auditor: Context-sensitive content injection mitigation," in *Proc. 25th Int. Symp. Res. Attacks, Intrusions Defenses*, 2022, pp. 431–445.

[91] S. Yadav, A. Mahajan, M. Prasad, and A. Kumar, "Advanced keylogger for ethical hacking," *Int. J. Eng. Appl. Sci. Technol.*, vol. 5, no. 1, pp. 634–38, 2020.

[92] K. Hussain, A. R. Rahmatyar, B. Riskhan, M. A. U. Sheikh, and S. R. Sindiramutty, "Threats and vulnerabilities of wireless networks in the Internet of Things (IoT)," in *Proc. IEEE 1st Karachi Sect. Humanitarian Technol. Conf. (KHI-HTC)*, 2024, pp. 1–8.

[93] I. Despotopoulos, "Wireless local area network security and modern cryptographic protocols: WEP & WPA1/2/3," M.S. thesis, Dept. Inform. Comput. Eng., Univ. West Attica, Aigaleo, Greece, 2024.

[94] B. Tsouvalas and N. Nikiforakis, "Knocking on admin's door: Protecting critical Web applications with deception," in *Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment*, 2024, pp. 283–306.

[95] M. A. I. Mallick and R. Nath, "Navigating the cyber security landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments," *World Sci. News*, vol. 190, no. 1, pp. 1–69, 2024.

[96] U. Joseph and M. Jacob, "Real time detection of phishing attacks in edge devices using LSTM networks," in *Proc. AIP Conf.*, 2022, Art. no. 20001.

[97] R. Ulfath, I. Sarker, M. Chowdhury, and M. Hammoudeh, "Detecting smishing attacks using feature extraction and classification techniques," in *Big Data, IoT, and Machine Learning* (Lecture Notes on Data Engineering and Communications Technologies), vol. 95. Singapore: Springer, 2022.

[98] S. Tang, X. Mi, Y. Li, X. Wang, and K. Chen, "Clues in Tweets: Twitter-guided discovery and analysis of SMS spam," in *Proc. ACM Conf. Comput. Commun. Security*, 2022, pp. 2751–2764.

[99] R. Mayrhofer, J. Stoep, C. Brubaker, and N. Kralevich, "The android platform security model," *ACM Trans. Privacy Security*, vol. 24, no. 3, p. 19, 2021.

[100] M. Suleman, T. Soomro, T. Ghazal, and M. Alshurideh, "Combating against potentially harmful mobile Apps," in *Proc. AICV*, 2021, pp. 154–173.

[101] M. E. Armstrong, K. S. Jones, and A. S. Namin, "How perceptions of caller honesty vary during vishing attacks that include highly sensitive or seemingly innocuous requests," *Human Factors*, vol. 65, no. 2, pp. 275–287, 2023.

[102] "scholar (11)".

[103] S. Mahdavifar, N. Maleki, A. Lashkari, M. Broda, and A. Razavi, "Classifying malicious domains using DNS traffic analysis," in *Proc. IEEE Int. Conf. Dependable, Auton. Secure Comput. Int. Conf. Pervasive Intell. Comput. Int. Conf. Cloud Big Data Comput. Int. Conf. Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech*, 2021, pp. 60–67.

[104] H. Owen, J. Zarrin, and S. Pour, "A survey on botnets, issues, threats, methods, detection and prevention," *J. Cybersecurity Privacy*, vol. 2, no. 1, pp. 74–88, 2022.

[105] H. Kilavo, L. Mselle, R. Rais, and S. Mrutu, "Reverse social engineering to counter social engineering in mobile money theft: A tanzanian context," *J. Appl. Security Res.*, vol. 18, no. 3, pp. 546–558, 2023.

[106] M. Wang, L. Song, L. Li, Y. Zhu, and J. Li, "Phishing Webpage detection based on global and local visual similarity," *Expert Syst. Appl.*, vol. 252, Oct. 2024, Art. no. 124120.

[107] A. Brunstein, "Automatic Web crawler for malicious websites classification," M.S. thesis, Corso di laurea magistrale Data Science Eng., Politecnico di Torino, Turin, Italy, 2024.

[108] D.-J. Liu and J.-H. Lee, "A CNN-based SIA screenshot method to visually identify phishing websites," *J. Netw. Syst. Manage.*, vol. 32, no. 1, p. 8, 2024.

[109] O. Sarker, A. Jayatilaka, S. Haggag, C. Liu, and M. A. Babar, "A multi-vocal literature review on challenges and critical success factors of phishing education, training and awareness," *J. Syst. Softw.*, vol. 208, Feb. 2024, Art. no. 111899.

[110] S. Das Guptta, K. T. Shahriar, H. Alqahtani, D. Alsalman, and I. H. Sarker, "Modeling hybrid feature-based phishing websites detection using machine learning techniques," *Ann. Data Sci.*, vol. 11, no. 1, pp. 217–242, 2024.

[111] A. Aljammal, S. taamneh, A. Qawasmeh, and H. Salameh, "Machine learning based phishing attacks detection using multiple datasets," *Int. J. Interact. Mobile Technol.*, vol. 17, no. 5, pp. 71–83, 2023.

[112] M. Sánchez-Paniagua, E. Fidalgo, E. Alegre, and R. Alaiz-Rodríguez, "Phishing websites detection using a novel multipurpose dataset and Web technologies features," *Expert Syst. Appl.*, vol. 207, Nov. 2022, Art. no. 118010.

[113] Q. Li, G. Zhong, C. Xie, and R. Hedjam, "Weak edge identification network for ocean front detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[114] L. Xue, "mT5: A massively multilingual pre-trained text-to-text transformer," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguist. Human Lang. Technol.*, 2021, pp. 1–17.

[115] S. Gopal and C. Poongodi, "Mitigation of phishing URL attack in IoT using H-ANN with H-FFGWO algorithm," *KSII Trans. Internet Inf. Syst.*, vol. 17, no. 7, pp. 1916–1934, 2023.

[116] H. Alqahtani, "Evolutionary algorithm with deep auto encoder network based Website phishing detection and classification," *Appl. Sci.*, vol. 12, no. 15, p. 7441, 2022.

[117] K. Apoorva and S. Sangeetha, "Analysis of uniform resource locator using boosting algorithms for forensic purpose," *Comput. Commun.*, vol. 190, pp. 69–77, Jun. 2022.

[118] V. Mazzeo, A. Rapisarda, and G. Giuffrida, "Detection of fake news on COVID-19 on Web search engines," *Front. Phys.*, vol. 9, Jun. 2021, Art. no. 685730.

[119] A. Aljofey, Q. Jiang, Q. Qu, M. Huang, and J. Niyigena, "An effective phishing detection model based on character level convolutional neural network from URL," *Electronics*, vol. 9, no. 9, pp. 1–24, 2020.

[120] E. S. Gualberto, R. T. De Sousa, T. P. De Brito Vieira, J. P. C. L. Da Costa, and C. G. Duque,, "The answer is in the text: Multi-stage methods for phishing detection based on feature engineering," *IEEE Access*, vol. 8, pp. 223529–223547, 2020.

[121] S. Shabudin, N. Sani, K. Ariffin, and M. Aliff, "Feature selection for phishing Website classification," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 4, pp. 587–595, 2020.

[122] A. Singh and S. Misra, "A comparison of performance of rough set theory with machine learning techniques in detecting phishing attack," in *Advances in Computing, Informatics, Networking and Cybersecurity* (Lecture Notes in Networks and Systems), vol. 289. Cham, Switzerland: Springer, 2022.

[123] M. El-Rashidy, "A smart model for Web phishing detection based on new proposed feature selection technique," *Menoufia J. Electron. Eng. Res.*, vol. 30, no. 1, pp. 97–104, 2021.

[124] A. Thahira and A. John, "Phishing Website detection using LGBM classifier with URL-based lexical features," in *Proc. IEEE Silchar Subsection Conf.*, 2022, pp. 1–7.

[125] H. Zhao, Z. Chen, and R. Yan, "Malicious domain names detection algorithm based on statistical features of URLs," in *Proc. IEEE 25th Int. Conf. Comput. Supported Cooperative Work Design*, 2022, pp. 11–16.

[126] R. Rao, T. Vaishnavi, and A. Pais, "CatchPhish: Detection of phishing websites by inspecting URLs," *J. Ambient Intell. Humaniz. Comput.*, vol. 11, no. 2, pp. 813–825, 2020.

[127] F. Kausar, B. Al-Otaibi, A. Al-Qadi, and N. Al-Dossari, "Hybrid client side phishing websites detection approach," *Int. J. Adv. Comput. Sci. Appl.*, vol. 5, no. 7, p. 9, 2014.

[128] C. Tan, K. Chiew, K. Yong, S. Sze, J. Abdullah, and Y. Sebastian, "A graph-theoretic approach for the detection of phishing Webpages," *Comput. Secur.*, vol. 95, Aug. 2020, Art. no. 101793.

[129] I. Kara, M. Ok, and A. Ozaday, "Characteristics of understanding URLs and domain names features: The detection of phishing websites with machine learning methods," *IEEE Access*, vol. 10, pp. 124420–124428, 2022.

[130] F. Sadique, R. Kaul, S. Badsha, and S. Sengupta, "An automated framework for real-time phishing URL detection," in *Proc. 10th Annu. Comput. Commun. Workshop Conf.*, 2020, pp. 335–341.

[131] A. Bozkir and M. Aydos, "LogoSENSE: A companion HOG based logo detection scheme for phishing Web page and E-mail brand recognition," *Comput. Secur.*, vol. 95, Aug. 2020, Art. no. 101855.

[132] M. Pandey, M. Singh, S. Pal, and B. Tiwari, "Prediction of phishing websites using stacked ensemble method and hybrid features selection method," *SN Comput. Sci.*, vol. 3, no. 6, p. 488, 2022.

[133] P. Indrasiri, M. Halgamuge, and A. Mohammad, "Robust ensemble machine learning model for filtering phishing URLs: Expandable random gradient stacked voting classifier (ERG-SVC)," *IEEE Access*, vol. 9, pp. 150142–150161, 2021.

[134] M. S. Munir Prince, A. Hasan, and F. M. Shah, "A new ensemble model for phishing detection based on hybrid cumulative feature selection," in *Proc. IEEE 11th Symp. Comput. Appl. Ind. Electron.*, 2021, pp. 7–12.

[135] L. Rani, C. Foozy, and S. Mustafa, "Feature selection to enhance phishing Website detection based on URL using machine learning techniques," *J. Soft Comput. Data Min.*, vol. 4, no. 1, pp. 30–41, 2023.

[136] J. Moedjahedy, A. Setyanto, F. Alarfaj, and M. Alreshoodi, "CCrFS: Combine correlation features selection for detecting phishing websites using machine learning," *Future Internet*, vol. 14, no. 8, p. 229, 2022.

[137] Y. Mansour and M. Alenizi, "Enhanced classification method for phishing emails detection," *J. Inf. Security Cybercrimes Res.*, vol. 3, no. 1, pp. 58–63, 2020.

[138] A. Alhussan, H. Al-Mahdawi, and A. Kadi, "Spam detection in connected networks using particle swarm and genetic algorithm optimization: Youtube as a case study," *Int. J. Wireless Ad Hoc Commun.*, vol. 6, no. 1, pp. 8–18, 2023.

[139] A. Ramana, K. Rao, and R. Rao, "Stop-Phish: An intelligent phishing detection method using feature selection ensemble," *Soc. Netw. Anal. Min.*, vol. 11, no. 1, p. 110, 2021.

[140] M. Grubbs, "Anti-phishing game-based training: An experimental analysis of demographic factors," *SSRN Electron. J.*, 2022, Preprint. [Online]. Available: http://dx.doi.org/10.2139/ssrn.4011558

[141] J. Brickley, K. Thakur, and A. Kamruzzaman, "A comparative analysis between technical and non-technical phishing defences," *Int. J. Cyber-Security Digit. Forensics*, vol. 10, no. 1, pp. 28–41, 2021.

[142] A. Chattopadhyay, C. Maschinot, and L. Nestor, "Mirror on the wall—What are cybersecurity educational games offering overall: A research study and gap analysis," in *Proc. Front. Educ. Conf.*, 2021, pp. 1–8.

[143] S. Bell and P. Komisarczuk, "An analysis of phishing blacklists: Google safe browsing, OpenPhish, and PhishTank," in *Proc. ACM Int. Conf. ACSW*, 2020, pp. 1–11.

[144] S. Chanti and T. Chithralekha, "Classification of anti-phishing solutions," *SN Comput. Sci.*, vol. 1, no. 1, p. 11, 2020.

[145] M. Korkmaz, E. Kocyigit, O. Sahingoz, and B. Diri, "A hybrid phishing detection system using deep learning-based URL and content analysis," *Elektronika ir Elektrotechnika*, vol. 28, no. 5, pp. 80–89, 2022.

[146] R. Zaimi, M. Hafidi, and M. Lamia, "Survey paper: Taxonomy of Website anti-phishing solutions," in *Proc. 7th Int. Conf. Social Netw. Anal., Manage. Security*, 2020, pp. 1–8.

[147] T. Suleman, "A survey on Web phishing detection techniques: A taxonomy-based approach," *LGU Int. J. Electron. Crime Investigation*, vol. 5, no. 2, pp. 1–12, 2021.

[148] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative analysis of intrusion detection systems and machine learning-based model analysis through decision tree," *IEEE Access*, vol. 11, pp. 80348–80391, 2023.

[149] S. Mishra, P. Mallick, H. Tripathy, A. Bhoi, and A. González-Briones, "Performance evaluation of a proposed machine learning model for chronic disease datasets using an integrated attribute evaluator and an improved decision tree classifier," *Appl. Sci.*, vol. 10, no. 22, p. 8137, 2020.

[150] A. Alsufyani and S. Alzahrani, "Social engineering attack detection using machine learning: Text phishing attack," *Indian J. Comput. Sci. Eng.*, vol. 12, no. 3, pp. 743–751, 2021.

[151] S. Wang, S. Khan, C. Xu, S. Nazir, and A. Hafeez, "Deep learning-based efficient model development for phishing detection using random forest and BLSTM classifiers," *Complexity*, vol. 2020, pp. 1–7, Sep. 2020.

[152] S. Sindhu, S. Patil, A. Sreevalsan, F. Rahman, and A. Saritha, "Phishing detection using random forest, SVM and neural network with backpropagation," in *Proc. Int. Conf. Smart Technol. Comput., Elect. Electron.*, 2020.

[153] A. Ozcan, C. Catal, E. Donmez, and B. Senturk, "A hybrid DNN–LSTM model for detecting phishing URLs," *Neural Comput. Appl.*, vol. 35, no. 7, pp. 4957–4973, 2023.

[154] G. Xu, T. Ren, Y. Chen, and W. Che, "A one-dimensional CNN-LSTM model for epileptic seizure recognition using EEG signal analysis," *Front. Neurosci.*, vol. 14, Dec. 2020, Art. no. 578126.

[155] Z. Alshingiti, R. Alaqel, J. Al-Muhtadi, Q. Haq, K. Saleem, and M. Faheem, "A deep learning-based phishing detection system using CNN, LSTM, and LSTM-CNN," *Electronics*, vol. 12, no. 1, p. 232, 2023.

[156] V. Adeyemo, A. Balogun, H. Mojeed, N. Akande, and K. Adewole, "Ensemble-based logistic model trees for Website phishing detection," in *Communications in Computer and Information Science*. Singapore: Springer, 2021.

[157] W. Ali and A. Ahmed, "Hybrid intelligent phishing Website prediction using deep neural networks with genetic algorithm-based feature selection and weighting," *IET Inf. Secur.*, vol. 13, no. 6, p. 11, 2019.

[158] D. Thanammal and D. Sujatha, "Phishing Website detection using novel features and machine learning approach," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 7, pp. 2648–2653, 2021.

SHAKEEL AHMAD was born in Hafizabad, Pakistan, in 6 September 1986. He received the graduation degree in business and finance from the University of Punjab, Lahore, Pakistan, in 2008, and the master's degree in computer science and information technology from the Division of Science and Technology, University of Education, Township Campus Lahore, Pakistan, in 2012. He is an Eminent Professional Educationist working as a Subject Specialist (Computer Science) with the School Education Department, Pakistan, from last ten years. He has started his professional career as a Subject Specialist (Computer Science) in 2014. He also started his career as Research Assistant in machine learning and deep learning in 2015. He also done, certification of Computer Application offered by the Government of Punjab, Pakistan, in 2008. His expertise spans research and development in machine learning and object detection algorithms, technical office management and administration, technical report writing, SOP writing, training management, and training policy administration across primary to advanced technical levels, showcasing his exceptional versatility and proficiency.

MUHAMMAD ZAMAN (Memeber, IEEE) received the Master of Science degree in computer science from the COMSATS University of Islamabad. He is a Committed Lecturer with the University of Lahore. He, driven by an unwavering dedication to the progression of knowledge, possesses extensive research acumen in numerous fields, including but not limited to medical image processing, remote sensing, and natural language processing. The individual's substantial research pursuits and contributions demonstrate a profound enthusiasm for artificial intelligence, machine learning, deep learning, computer vision, and reinforcement learning. His numerous publications in computer vision have substantiated his contributions to the field. Furthermore, he has served as a mentor and guide to a considerable number of M.S. students as they have completed their research theses and papers. This demonstrates his dedication to cultivating the subsequent generation of scholars and innovators.

AHMAD SAMI AL-SHAMAYLEH received the master's degree in information systems from The University of Jordan, Jordan, in 2014, and the Ph.D. degree in artificial intelligence from the University of Malaya, Malaysia, in 2020. He is currently an Assistant Professor with the Faculty of Information Technology, Al-Ahliyya Amman University, Jordan. His research interests include: Artificial intelligence, human computer interaction, IoT, arabic NLP, arabic sign language recognition, language resources production, the design and evaluation of interactive applications for handicapped people, multimodality, and software engineering.

RAHIEL AHMAD was born in Hafizabad, Pakistan, in 4 February 1987. He received the bachelor's degree in software engineering from COMSATS University Islamabad and the master's degree in computer science (specialized in AI & ML) from the University of Lahore, Pakistan, in 2021. With a rich and diverse professional background, his expertise spans research and development in machine learning and object detection algorithms showcasing his exceptional versatility and proficiency.

SHAFI'I MUHAMMAD ABDULHAMID is currently an Assistant Professor with the Department of Information Technology, Community College of Qatar. His research interests include soft computing, machine learning, fog computing, and cloud computing security.

ISMAIL ERGEN received the Doctoral degree in artificial intelligence and graphic design from Arel University. He is a distinguished professional with a rich background in art, design, and emerging technologies. After completing his university education in Turkey, he relocated to the United States, where he spent 15 years honing his skills and expanding his knowledge base. During this period, he completed his Proficiency in Art education with the Academy of Art University, San Francisco, one of the world's foremost art institutions. His career in America spanned over a decade, during which he worked in various roles related to advertising, art, and business. He is also an innovative entrepreneur, having established a toy company in the United States that specializes in designing cutting-edge 3-D toys. Currently, he holds faculty position with Istinye University, Istanbul, where he delves into the impact of emerging technologies on game design. His work encompasses game design, artificial intelligence, experience design, as well as UX and UI, reflecting his multifaceted expertise and ongoing commitment to advancing the field of design through technology.

ADNAN AKHUNZADA (Senior Member, IEEE) received the M.S. degree in information security and the Ph.D. degree in network security. Seamlessly navigating the intersection of ICT industry and academia, he stands as a testament to excellence and innovation. Renowned for high-impact publications, U.S. Patents, and commercial products. His patented cybersecurity and AI innovations have secured multimillion-dollar projects for global corporations, such as Vinnova and EU Horizon. In recognition of his outstanding scholarly contributions, Stanford University acknowledged him as one of the top 2% scientists globally in 2023. He leverages his strong cybersecurity skills and cutting-edge technological knowledge to solve industrial problems and develop state-of-the-art security tools, techniques, and frameworks. He is a Postdoctoral Fellow of Cybersecurity. His expertise in cybersecurity and AI, secure future Internet, modeling and designing secure and dependable software defined networks, and large-scale distributed systems (Cloud, Fog, Edge, IoT, IoE, IIoT, and CPS); lightweight cryptographic next generation communication protocols, QoS/QoE, and adversarial machine learning is helping shape the future of secure and dependable systems. He is a Professional Member of ACM, possesses a rich and accomplished tenure of 15 years in research and development.