IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 8, 2021, accepted January 21, 2022, date of publication February 17, 2022, date of current version April 8, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3151903

Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions

NGUYET QUANG DO[1], ALI SELAMAT[1,2,3,4], (Member, IEEE), ONDREJ KREJCAR[1,4], ENRIQUE HERRERA-VIEDMA[5,6], (Fellow, IEEE), AND HAMIDO FUJITA[1,5,7,8], (Life Senior Member, IEEE)

[1]Malaysia–Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur, Wilayah Persekutuan Kuala Lumpur 50088, Malaysia
[2]School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Johor Baru, Johor 80000, Malaysia
[3]Media and Games Center of Excellence (MagicX), Universiti Teknologi Malaysia, Johor Baru, Johor 80000, Malaysia
[4]Center for Basic and Applied Research, Faculty of Informatics and Management, University of Hradec Kralove, 050003 Hradec Kralove, Czech Republic
[5]Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI), University of Granada, 18011 Granada, Spain
[6]Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[7]i-SOMET incorporated Association, Morioka 020-0104, Japan
[8]Regional Research Center, Iwate Prefectural University, Iwate 020-0693, Japan

Corresponding authors: Nguet Quang Do (milkydove83@gmail.com); Ali Selamat (aselamat@utm.my); and Hamido Fujita (hfujita@i-somet.org)

This work was supported in part by the Ministry of Higher Education under the Fundamental Research Grant Scheme under Grant FRGS/1/2018/ICT04/UTM/01/1; and in part by the Faculty of Informatics and Management, University of Hradec Kralove, through SPEV project under Grant 2102/2022.

ABSTRACT Phishing has become an increasing concern and captured the attention of end-users as well as security experts. Existing phishing detection techniques still suffer from the deficiency in performance accuracy and inability to detect unknown attacks despite decades of development and improvement. Motivated to solve these problems, many researchers in the cybersecurity domain have shifted their attention to phishing detection that capitalizes on machine learning techniques. Deep learning has emerged as a branch of machine learning that becomes a promising solution for phishing detection in recent years. As a result, this study proposes a taxonomy of deep learning algorithm for phishing detection by examining 81 selected papers using a systematic literature review approach. The paper first introduces the concept of phishing and deep learning in the context of cybersecurity. Then, taxonomies of phishing detection and deep learning algorithm are provided to classify the existing literature into various categories. Next, taking the proposed taxonomy as a baseline, this study comprehensively reviews the state-of-the-art deep learning techniques and analyzes their advantages as well as disadvantages. Subsequently, the paper discusses various issues that deep learning faces in phishing detection and proposes future research directions to overcome these challenges. Finally, an empirical analysis is conducted to evaluate the performance of various deep learning techniques in a practical context, and to highlight the related issues that motivate researchers in their future works. The results obtained from the empirical experiment showed that the common issues among most of the state-of-the-art deep learning algorithms are manual parameter-tuning, long training time, and deficient detection accuracy.

INDEX TERMS Cybersecurity, deep learning, machine learning, phishing detection.

I. INTRODUCTION

Phishing detection based on machine learning (ML) have received tremendous attention and interest from researchers in the cybersecurity community over the past decade. Extensive researches have been conducted to review the application of ML in various solutions to detect evolving phishing attacks [1]–[3]. Deep learning (DL), a subset of ML, has recently emerged as a potential alternative to traditional ML approaches. However, there are limited studies that discuss in depth the application of DL in phishing detection, their advantages and disadvantages, the current issues, and future research directions to address these challenges [4]–[6]. Notably, there is no study that provides a comprehensive review of the current challenges and future directions for DL algorithms with regards to phishing detection using a systematic literature review (SLR) approach. To the best of our knowledge, this is the first study that discussed phishing detection and DL in a single SLR paper.

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed Farouk.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

TABLE 1. Limitations of Existing Studies and Novelty of This Research Work.

Reference	Taxonomy	Current challenges	Future directions	Remark
[1]	✓			Reviewed only conventional ML techniques Did not include DL approaches in the literature Did not discuss the existing issues or suggest the future research directions
[4]	✓			Did not examine the most recent DL techniques for phishing detection Limited discussion on open challenges and future directions
[5]	✓			Lacked an exhaustive analysis on DL-based phishing detection approach Did not discuss the current issues or future research directions
[6]	✓			Did not investigate the most recent DL algorithms Did not discuss the open challenges or recommend future research directions
[2]		✓	✓	Lacked an extensive review on different types of phishing attacks and DL algorithms
[3]		✓	✓	Lacked an in-depth classification of phishing detection methods Emphasized more on traditional ML techniques
[7]	✓	✓		Focused only on the role and influences of features used for learning Did not analyze DL for phishing detection in detail Contained limited discussion on future research directions
[8]	✓	✓	✓	Concentrated more on conventional ML approaches
[9]	✓	✓	✓	Did not provide an in-depth analysis of DL for phishing detection
[10]	✓	✓	✓	Lacked a discussion on DL classifier to detect phishing attacks
Our research work	✓	✓	✓	Provide an in-depth analysis of DL algorithm for phishing detection using SLR approach Include the state-of-the-art DL techniques Discuss the current challenges and future research areas

TABLE 1 provides a comparison between our research and the related surveys on the topic of interest. The related studies were reviewed and compared from the perspectives of: (i) proposing a taxonomy of phishing detection, ML, or DL, (ii) providing a detailed discussion on the current challenges facing DL in phishing detection, and (iii) offering recommendations for future research. It is observed that among these studies, some authors provided taxonomies of the related topics, but did not discuss the open issues and future research areas [1], [4]–[6]. In contrast, other authors lacked an exhaustive review and classification of phishing detection; yet, they included current challenges and future directions in their studies [2], [3]. The authors in [7] conducted an in-depth benchmarking and evaluation on phishing detection, but primarily focused on the importance of features used for learning. Even though all three viewpoints above were considered in [8]–[10], the authors emphasized more on conventional ML techniques and did not provide a detailed analysis of DL for phishing detection.

Whereas, our research is different from the existing studies in which it provides an in-depth analysis of the DL algorithm for phishing detection through an SLR approach. Moreover, our study also includes the state-of-the-art DL techniques, and most importantly, discusses the current challenges and future research direction for DL in the phishing detection domain. This study is intended to guide researchers and developers, to whom DL and phishing detection would be of primary concerns. The in-depth analysis in this research has led to several key contributions.

- We adopted a SLR approach to analyze the relevant studies and selected a total of 81 articles based on several criteria to support this research.
- We proposed a taxonomy of phishing detection and DL by dividing them into several categories. In addition, we also surveyed numerous DL algorithms and discussed their strengths and weaknesses.
- We identified the current challenges and key issues related to DL in the field of phishing detection, and provided recommendations for future research areas.
- We conducted an empirical analysis of various DL architectures for phishing detection, and highlighted several issues previously discussed in the literature to identify possible gaps for future research directions.

The rest of this paper is organized as follows. Section II provides background knowledge of phishing attacks, DL, and the adopted SLR approach that leads to the selection of 81 reviewed papers. Section III presents a taxonomy of phishing detection and DL to classify them according to several categories. Section IV discusses current issues and challenges facing DL in an attempt to fight against phishing attacks. Section V identifies potential research gaps and recommends future research directions. An empirical analysis is included in Section VI to map current issues with existing research gaps. Finally, Section VII concludes the paper and proposes future works.

II. BACKGROUND

This section consists of two main sub-sections to provide a comprehensive understanding of the research topic. The first section provides the definition of phishing and DL, while the second section describes the SLR approach used in this paper.

A. DEFINITION

This sub-section provides a brief introduction of phishing attacks and DL algorithms. A basic knowledge about phishing and its operation will assist in the understanding of why

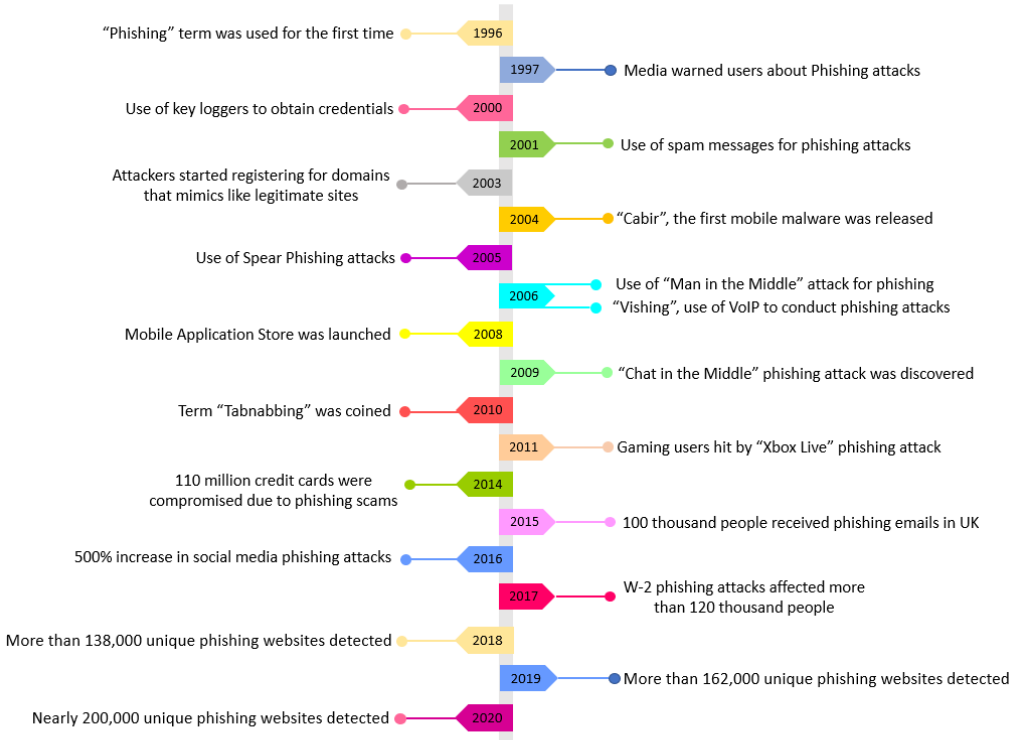

FIGURE 1. Evolution of phishing attacks from 1996 to 2020.

DL has emerged as a promising solution to detect phishing activities.

1) PHISHING

Phishing is a type of digital theft that disguises itself as legitimate or genuine sources to steal uses' private and confidential information. It has become a popular attacking approach in cyberspace by utilizing web applications' vulnerabilities and end users' ignorance, which is a security issue that needs to be addressed [11].

The evolution of phishing attacks is illustrated in FIGURE 1 [12]. Back in 1996, the term ''phishing'' was first introduced, and phishing attacks were slowly spread through various communication media over the years. It started with spam messages, mobile malware, spear-phishing to ''Man in the Middle'', Vishing, ''Chat in the Middle'', ''Tabnabbing'', ''Xbox Live'', etc. Phishing attacks started becoming a serious issue and caught more attention among researchers when a major incident happened in 2014, causing a huge financial loss. With the advent of the Internet and the popularity of social media, the number of phishing attacks has increased rapidly since 2016 and continued to grow in an upward trend. According to the latest statistics from APWG (Anti Phishing Working Group), the number of phishing attacks has grown tremendously since March 2020 and doubled over the course of the year [13].

Since phishing has become a serious security issue, understanding how it operates is an utmost important task in the detection and prevention of such cybersecurity threat. The life cycle of a typical phishing attack is shown in FIGURE 2, consisting of five phases [14]. The first phase is called reconnaissance or planning phase, in which the phishers choose the communication media, select the phishing vector, and identify potential victims [12], [15]. The second phase is weaponization or preparation phase, whereby phishers prepare phishing materials to be propagated to their targeted victims [14]. The next stage is distribution or phishing phase, as phishers start to deploy the baits by delivering the phishing materials to victims [16]. The following stage is called exploitation or penetration phase, where phishers exploit victims' weaknesses by luring them into giving up their private

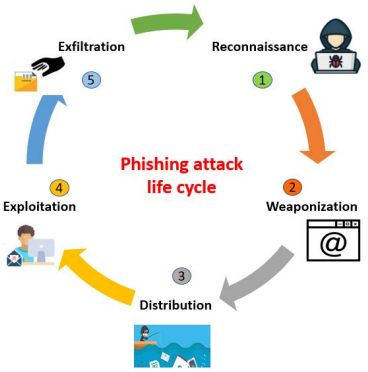

FIGURE 2. Phishing attack life cycle.

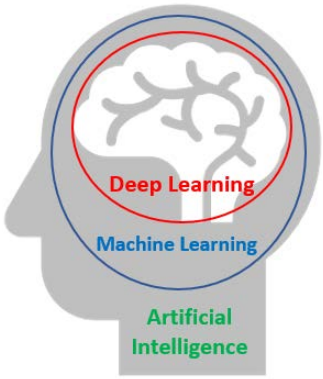

FIGURE 3. Venn diagram of AI, ML, and DL.

and confidential information. [17]. The final stage is known as exfiltration or data acquisition phase. The phishing operation has succeeded at this point, and phishers had successfully obtained the information they intended to take when planning the phishing attack initially. Phishers can decide to take further actions to gain financial benefits, or use the collected information for other purposes [12].

2) DEEP LEARNING (DL)

Phishing appears to be an effective way for cybercrime to occur because most users are unable to identify phishing websites or emails [18]. One of the current challenges in dealing with cyberthreats, especially phishing attacks, is lacking of cyber security solution, and Artificial Intelligence (AI) is believed to be the next frontier in cyber security defense [19].

ML is a part of AI that teaches machine the ability to learn like human beings. DL is a subset of ML derived from a neural network model (FIGURE 3). Traditional ML techniques refer to the learning methods that require human expertise to perform feature extraction and selection [20]. Feature selection is separated from classification task in a classical ML model, and these two processes cannot be combined together to optimize the model's performance. However, DL fills this gap by integrating these two processes in a single phase to detect and classify phishing attacks effectively and efficiently [21]. Although traditional ML approaches provide high accuracy and low false-positive rate, they still require manual feature engineering and depend on third-party services [22]. In contrast, DL models can learn and extract features automatically without human intervention. This eliminates the need for manual feature engineering and third-party services dependency. Moreover, traditional ML with manual feature engineering fails to deal with multi-dimensional and large-scale datasets in the big data era [23]. DL, however, can to handle a significant amount of data and becomes a powerful tool for phishing detection that requires more attention in the cybersecurity community. There was no study that combined DL and phishing detection in a SLR approach despite the increasing attention given to these two domains. Therefore, a detailed process of selecting relevant studies was described

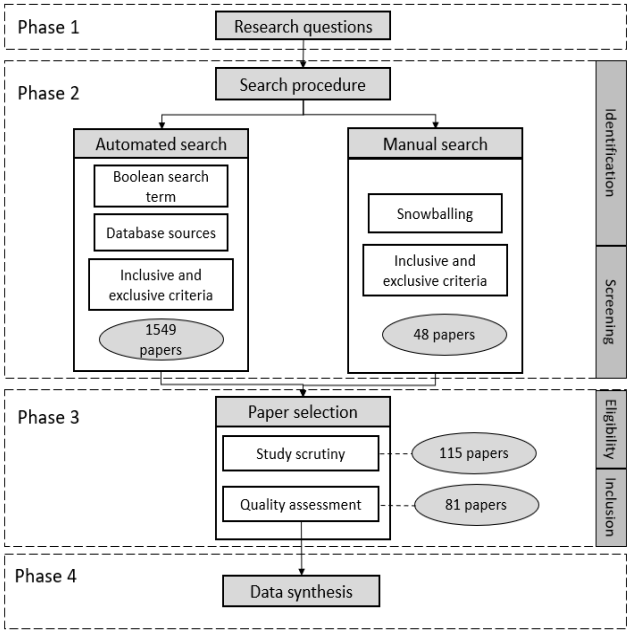

FIGURE 4. SLR research method.

in this paper, to examine the current trends and patterns in the existing research on DL for phishing detection. The primary purpose of conducting this SLR is to analyze the pros and cons of the state-of-the-art DL techniques, identify the current issues, highlight the research gaps, and recommend future research directions.

B. SYSTEMATIC LITERATURE REVIEW

This study adopted an approach suggested by Kitchenham [24] to conduct a SLR on the research topic. FIGURE 4 illustrates the process of selecting the relevant studies, consisting of four phases: research questions, search procedure, paper selection and data synthesis.

1) PHASE 1: RESEARCH QUESTIONS

This SLR aims to examine the application of DL techniques in the phishing detection domain, which raises the following research questions (RQs):

RQ1: What are the existing DL techniques used to detect phishing attacks in cyberspace?

RQ2: What are the advantages and disadvantages of the existing DL techniques?

RQ3: What are the major challenges facing DL and the future research directions in phishing detection?

2) PHASE 2: SEARCH PROCEDURE

An automatic search method was used by running a Boolean search string on several database resources to find the answers for the RQs above. The term was described as follows: (deep learning OR "DL") AND (phishing detection OR phish detection). Five different online databases were used in this study to search for the most relevant papers published

TABLE 2. Quality Assessment Questions.

No	QA questions
1	Is the DL technique described?
2	Is the experiment reported and explained in the study?
3	Are the strengths and weaknesses of DL techniques mentioned?
4	Are the issues and challenges for DL or phishing detection identified?
5	Are the future directions stated in the research study?

between 2018 and 2021. These include: Web of Science (WoS), IEEEXplore, Springer Link, Science Direct, and Google Scholar.

3) PHASE 3: PAPER SELECTION

This SLR applied a paper selection process based on PRISMA guidelines [25] which consists of several stages, such as automatic search, duplicity removal, title and abstract screening, full-text selection, and snowballing [26]. Quality assessment (QA) is the next step after the paper selection process that aims to evaluate the selected papers' quality.

TABLE 2 shows a list of five QA questions used in this SLR to obtain the most relevant studies capable of answering the RQs. A weighting or scoring technique [24] was adopted, where three possible scores can be given to an answer of each QA question: ''1'' for ''Yes'', ''0.5'' for ''Partly'', and ''0'' for ''No''. Eighty one (81) papers were selected for this study based on the sum of the total score to all five QA questions.

Appendix B shows the detailed scores of QA questions to ensure that the selected papers are the most relevant to the RQs and this SLR study.

4) PHASE 4: DATA EXTRACTION AND SYNTHESIS

A qualitative analysis software (Nvivo) was used in this study to extract data from 81 selected papers. The extracted data comprised of authors' names, published year, paper's title, objective, methodology, findings, and future works. Other related fields, such as publisher's name, quartile, impact factor, and citation count, were also included as the selected papers' quality indicators. The extracted data went through a process called data synthesis to answer the RQs, and was illustrated using visualization techniques such as tables, figures, and charts to present the findings.

5) THREATS TO VALIDITY (TTV)

Four common threats to validity were taken into consideration while carrying out this research, including constructing validity, internal validity, external validity, and conclusion validity [27]. Minimizing the risks of these TTVs helped to reduce the probability of missing relevant studies as much as possible and to make sure that the paper selection process was unbiased.

To sum up, 81 papers were selected for this research study based on three perspectives mentioned in Section I, and according to several selection criteria from a systematic literature review. By adopting an approach proposed by Kitchenham [24], following a selection process from PRISMA guidelines [25], applying the scoring technique adopted by previous authors [21], [24], and considering several threats to validity [27], we hold the belief that the reviewed articles are among the most relevant studies related to the research area, and more importantly, are selected based on objective criteria, and without biases.

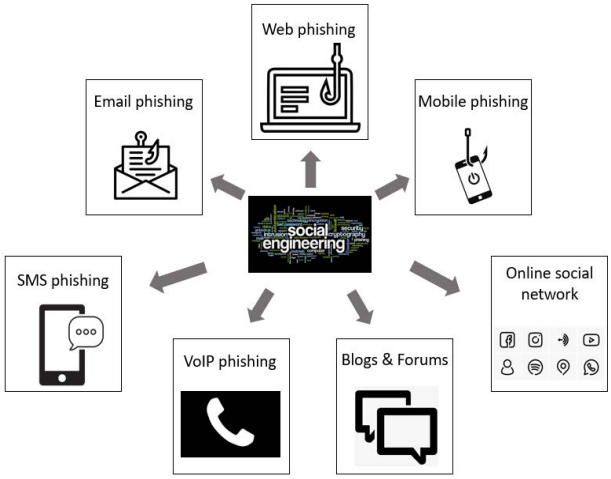

FIGURE 5. Phishing through social engineering.

III. TAXONOMY

The selected studies were analyzed and classified into different categories to answer RQ1 and RQ2. Phishing detection was classified according to various media and methods. Whereas, DL was divided into several categories based on the application areas, techniques and datasets.

A. PHISHING DETECTION

1) CLASSIFICATION BY MEDIA

Cyber criminals carry out phishing attacks through various media, and social engineering is one of them [28]. Social engineering is a technique of deceiving users into giving up their valuable and sensitive information such as username, password or credit card number [17]. Instead of targeting the systems, social engineering attacks aimed at the users who are the weakest link in the security chain [10]. Common social engineering methods for phishing attacks include Website, Email, Short Message Service (SMS), Voice over Internet Protocol (VoIP), Mobile Devices, Blogs and Forums, and Online Social Network (OSN) [8] as shown in FIGURE 5.

a: PHISHING THROUGH WEBSITE

Website phishing is the most common phishing attacks in cyberspace where attackers build the websites to make them look identical to the genuine ones [29]. The attackers' primary goal is to trick users into believing that these websites are trustworthy since they are the replica of well-known sources such as Google, eBay, Amazon, Paypal, etc. Thereby, attackers can gain personal and financial details from the users by taking advantage of their ignorance and

carelessness [12]. Since the phishers' target is the users and not their devices, website phishing is challenging regardless of how robust a phishing detection system is. Both technical and psychological solutions are required in the prevention and mitigation of such phishing attacks [17].

b: PHISHING THROUGH EMAIL

Cyber criminals usually send emails to online users claiming that they are from trusted companies to perform email phishing. They design the phishing emails to disguise themselves as legitimate organizations and urge the end-users to visit a fake website through a hyperlink included in it [28]. Users are often asked to update their information through this link and when they do so, phishers steal their confidential information for financial gain or other illegal purposes. Email phishing can be further divided into two groups: spear phishing and whaling [17].

Spear phishing targets at specific individuals, groups or organizations rather than random users with the final intention of obtaining confidential and sensitive information [16]. It is a well-planned attack where phishers initially collect information and details of their targeted victims, and then send emails pretending they are sent from a colleague, supervisor or manager in the same organization [30]. Spear phishing has a higher success rate as compared to other conventional methods because attackers disguise themselves as someone whom the victim knows and include content that is relevant to the victim in the email to avoid any suspicion [15].

Whaling is similar to spear phishing except that its targets are high-profile executives such as corporate CEOs, government officials or political leaders [16]. Phishers choose their victims based on their privileged access to the information or the authority they hold within the organization [15]. Phishers invest relatively more time and effort in this type of attack to enhance the success rate since the profit that is potentially earned from it is significant.

c: PHISHING THROUGH SMS

SMS phishing, also known as Smishing, is one of the popular attacks carried out on mobile phones. Smishing attackers usually send text messages to mobile phone users together with a link embedded in it [12]. When users click this link, they will be either redirected to a fake website or end up downloading and installing malicious software (malware) on their phones. Individuals can exchange short text messages at their fingertips nowadays with the advancement in mobile technology [15]. Such convenience allows attackers to approach their victims easily in an attempt to steal their private information. Even though SMS has become less popular due to the emergence of the Internet and other applications, Smishing still imposes a major threat in cyber security since text messages have been used as one of the common methods for online account verification [3].

d: PHISHING THROUGH VoIP

Besides SMS, voice is another medium for phishing attacks to take place in the cyber environment. VoIP phishing, or Vishing, is a type of phishing attack conducted over telephone systems or VoIP systems using voice technology [28]. Phishers often collect details about the victims prior to their conversation, such as name, address, phone number and other personal information, to gain more trust from the victims and make the attacks less suspicious. Vishing also has a high rate of success because some people believe that communicating with another human is more reliable than with a machine [15]. In addition, phone call receivers tend to make more mistakes during a phone call since they do not have enough time to think before responding or answer without proper consideration, and accidentally reveal their private and sensitive information to the phishers.

e: PHISHING THROUGH MOBILE DEVICES

Phishing through mobile phones has become more common recently as more and more people are relying on their phones to carry out their daily activities, from checking emails to paying bills, from browsing the Internet to online shopping, etc. [3]. This makes mobile phone users become potentially easy targets to phishers who plan to perform phishing attacks. Users may fall victim to such attacks while browsing or downloading an application from untrusted websites [12]. Once the malicious software is installed, it will collect the user's credentials and send them to the phishers for financial gain. Users usually find it difficult to distinguish between phishing and legitimate websites due to the small screen of mobile phones, limiting the amount of information to be displayed on the user interface, and the lack of security indicators of an application [15].

f: PHISHING THROUGH OSN

Social networking has become an indispensable part of the Internet, and millions of people's lives around the world. Online social network (OSN) such as Facebook, Twitter, Instagram, etc., become a new ground of attacks for phishers to perform their phishing activities [28]. Social network sites allow online users to interact, exchange and share information with each other, making it easier for phishers to conduct their illegal acts. Phishers mimic themselves as someone whom the users know of on these online social platforms and exploit their trust to gain financial benefits by taking advantage of these sites' popularity [12].

2) CLASSIFICATION BY METHODS

Phishing detection can be classified according to different methods, such as list-based, heuristic-based, visual similarity, machine learning, deep learning, and hybrid. Examples of each method are displayed in FIGURE 6, and their abbreviations are explained in TABLE 3.

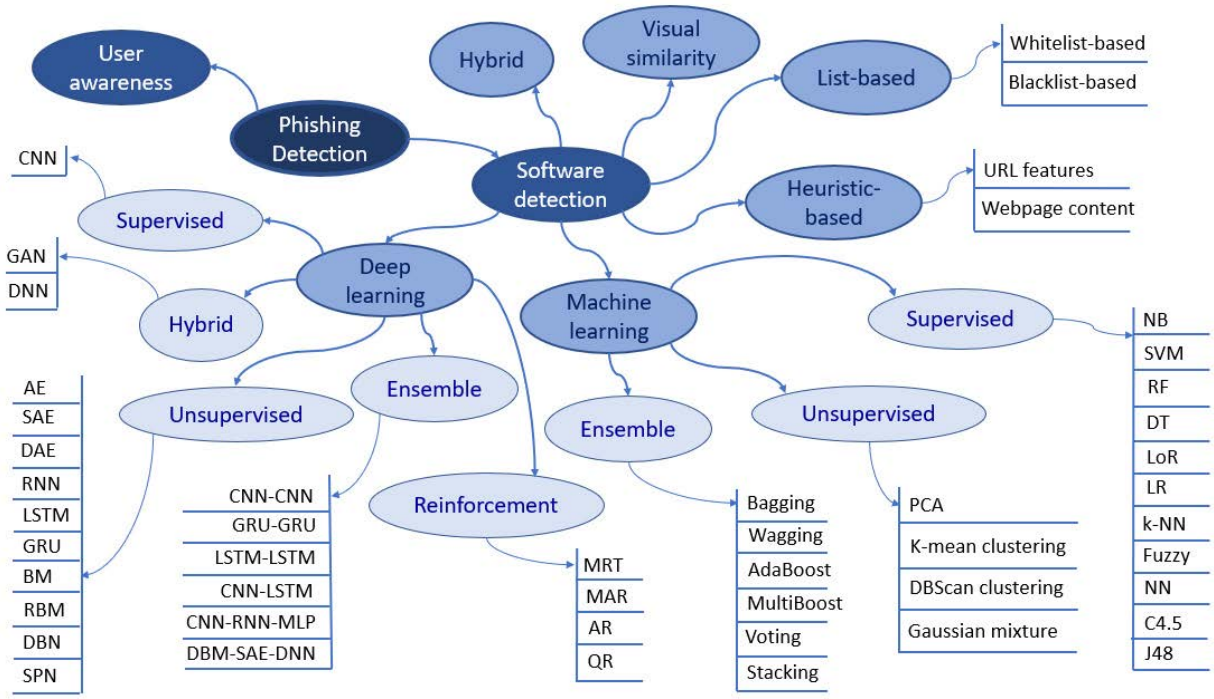

FIGURE 6. Taxonomy of phishing detection method.

a: LIST-BASED METHOD

List-based is a phishing detection approach used to differentiate between phishing and legitimate webpages based on a collected list of trusted and suspicious websites. The list-based approach can be divided into two groups: blacklist and whitelist [10]. Blacklist is a list of malicious or suspicious websites in which users should not access. When users try to access any URL in the blacklist, they will be warned of potential phishing attacks and prevented from accessing the website [31]. On a contrary, a whitelist is a collection of all legitimate and trusted websites. Any webpages that are not included in the whitelist will be considered suspicious. Once users attempt to access webpages that are not listed as secure sites, they will be alert of the possible risk [12]. The blacklist-based approach is comparatively effective in phishing detection because it offers a low false-positive rate and provides simplicity in design and ease of implementation [32]. However, the main drawback of this approach is an inability to classify new malicious websites and to recognize non-blacklisted or temporary phishing pages [31]. As a result, it is unable to detect unknown or zero-day attacks. In addition, blacklists need to be updated frequently and require human intervention and verification. Hence, they consume a great amount of resources and are prone to human error [33]. Due to these limitations, it is advisable to combine list-based method with other approaches which can handle zero-day attacks, at the same time keeping the low false-positive rate.

b: HEURISTIC-BASED METHOD

Developed from list-based, heuristic-based phishing detection approach depends on numerous features extracted from the webpages' structure to identify fake and untrusted sites. These features will be fed into a classifier to build an effective phishing detection model [31]. Phishing site characteristics in a heuristic-based approach are created based on several hand-crafted features, such as URL-based features, webpage contents, etc. Phishing webpages are detected by evaluating, examining, and analyzing these manually selected components [22]. Unlike blacklist, the heuristic-based approach can detect potential phishing attacks once the webpages are loaded, even before their URLs are updated in the blacklist. Since heuristic method has better generalization capability, it can be used to detect new phishing attacks. Yet, such method is only limited to a number of common threats, and is unable recognize newly evolving attacks [9]. Besides, heuristic-based method tends to have a higher false-positive rate as compared to blacklist [8]. Consequently, it can be combined with other approaches to solve the high false-positive rate problem.

c: VISUAL SIMILARITY

Phishing webpages are detected by checking and comparing the visual representation of the websites in visual similarity approach, rather than analyzing the source code behind it [17]. Identification of malicious webpages can be done by finding the resemblance with legitimate sites in page layout, page style, etc. Another method is to take the snapshot of the targeted websites and compare with the ones in the database using image processing technologies [34]. Phishing detection based on visual features of webpages' appearance relies on the assumption that phishing sites are similar to the legitimate ones [5], which might not always be the case. Plus, it requires

TABLE 3. List of Acronyms for ML and DL Techniques.

Acronym	Explanation
NB	Naïve Bayes
SVM	Support Vector Machine
RF	Random Forest
DT	Decision Tree
LoR	Logistic Regression
LR	Linear Regression
k-NN	K Nearest Neighbor
PCA	Principle Component Analysis
DBScan	Density-Based Spatial Clustering of Applications Noise
NN	Neural Network
CNN	Convolutional Neural Network
DNN	Deep Neural Network
MLP	Multilayer Perceptron
RNN	Recurrent Neural Network
GRU	Gated Recurrent Unit
LSTM	Long-Short Term Memory
AE	Autoencoder
SAE	Stacked Autoencoder
DAE	Denoising Autoencoder
BM	Boltzmann Machine
RBM	Restricted Boltzmann Machine
DBM	Deep Boltzmann Machine
DBN	Deep Belief Network
SPN	Sum Product Network
GAN	Generative Adversarial Network
MRT	Multi-task Reinforcement
MAR	Multi-agent Reinforcement
AR	Asynchronous Reinforcement
QR	Q-learning Reinforcement

higher computational cost since storing snapshots of websites need more space than storing their URL. Similar to the heuristic-based method, phishing detection based on visual similarity has higher false-positive rates than list-based [35].

d: MACHINE LEARNING (ML)

Features are extracted and classified using ML techniques in ML-based approach. The accuracy of the classification technique depends on the selected algorithm [36]. This algorithm will be used to produce an accurate classifier model to differentiate between phishing and legitimate websites [31]. Examples of frequently-used ML techniques include Naïve Bayes (NB), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), k-Nearest Neighbor (kNN), J48, C4.5, etc [3], [7], [10], [29]. Similar to heuristic, ML approach can detect zero-hour phishing attacks, which is an advantage over the blacklist method [1]. Moreover, it also has additional advantages as compared to the heuristic approach. For instance, ML techniques can construct their own classification models when a significant set of data is available, without the need to manually analyze data to understand the complicated relationship among them. Unlike heuristic, ML can achieve low false positive rate [8]. ML classifiers can also evolve to adapt to the changes in phishing trends as the phishing tactics evolve.

e: DEEP LEARNING (DL)

DL architecture is built based on neural networks with the ability to discover hidden information in the complex data through level-by-level learning [37]. DL approach has become more and more popular in the phishing detection domain with the recent development of DL technologies [2]. Although DL requires a more significant dataset and longer training time than the traditional ML method, it can extract the features automatically from raw data without any prior knowledge [23]. Various DL-based techniques have been employed recently to enhance the performance of classification for phishing detection [22]. Popular algorithms based on DL architecture include Convolutional Neural Network (CNN) [22], [38]–[41], Deep Neural Network (DNN) [42]–[45], Recurrent Neural Network (RNN) [46], [47], Long Short-Term Memory (LSTM) [44], [48]–[50], Gated Recurrent Unit (GRU) [48], [51], [52], and Multi-Layer Perceptron (MLP) [53]–[55], etc. It is believed that DL algorithms will become a promising solution for phishing detection in the near future due to a wide range of benefits that they offer [3].

f: HYBRID METHOD

The hybrid approach combines different classification techniques to achieve better performance in detecting malicious websites [22]. For instance, in a hybrid model where two different algorithms are combined, the dataset is trained using the first algorithm and then the result is passed to the second algorithm for training [36]. The overall accuracy of the hybrid model is believed to be higher than those from each individual algorithm. When new solutions are proposed to encounter various phishing attacks, cyber criminals will always take advantage of the vulnerabilities of the solutions and come up with new methods and produce new attacks [56]. Therefore, it is recommended to use hybrid models since a single approach has its own drawbacks that need to be addressed. Hybrid models combine different classification techniques to merge their advantages and resolve their individual disadvantages. As a result, phishing detection using a hybrid algorithm offers higher accuracy and provides a more decisive classification of phishing [3].

B. DEEP LEARNING

Since DL is getting more and more popular as one of the effective phishing detection methods, it has become a topic of interest in this study. The following section classifies DL into several classes, including application areas, techniques and datasets.

1) CLASSIFICATION BY APPLICATION AREAS

Intrusion detection, malware detection, spam detection, and phishing detection are common areas that applied DL algorithms (FIGURE 7) [57]–[61].

Intrusion detection is a technique to discover network security violations from both outsiders and insiders by monitoring and analyzing the traffic generated from various components in the network [62]. The primary purpose of an intrusion detection system (IDS) is to manage hosts and networks, monitor the behaviors of computer systems, give

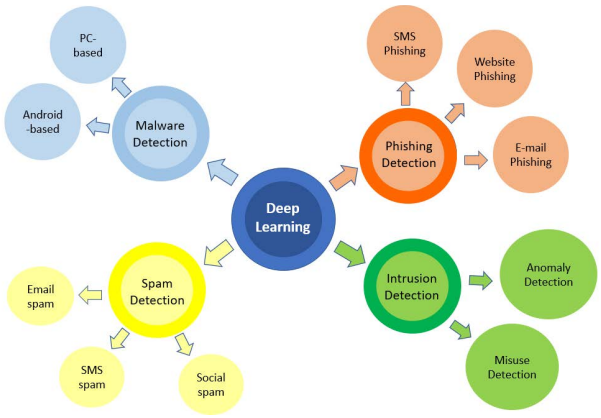

FIGURE 7. Main branches of applying DL in cybersecurity.

warnings if suspicious behaviors are found and take specific actions to respond to these illegal and unauthorized activities [63]. IDS can be divided into three types: anomaly detection, misuse detection, and hybrid [59]. Normal behavior in anomaly detection is defined and used as a baseline. Then, abnormal behaviors are identified by comparing them to the normal ones. Whereas, suspicious behaviors are represented as signatures in misuse detection, also known as signature-based detection. A signature database is established, and network attacks are identified if they match these signatures. Hybrid is a combination technique that leverages the advantages of both anomaly and misuse detection methods. There have been many research conducted to develop DL-based models for intrusion detection systems [23], [64], [65], since DL-based methods can detect unknown malicious attacks, reduce false alarm rates and enhance the detection accuracy.

Malware detection is a method to detect malicious software that aims to interrupt a system's normal operation, bypass authentication, collect personal information, and take control of the device without users' realization. Examples of common malware include worms, viruses, Trojan, botnet, rootkits, adware, spyware, ransomware, etc. [66]. Malware has become a major concern among cybersecurity experts in recent years; thus, having an effective and robust detection approach is crucial to handle rapidly evolved malware threats [61]. Malware detection methods can be categorized into two groups: PC-based and Android-based. Android malware detection appears to be more popular due to an increase in the adoption of mobile devices using the Android operating system nowadays [59]. Since DL approaches have achieved successful results in different fields, they can also be applied to malware identification and classification. The utilization of DL for malware detection offers an effective solution to distinguish various malware and their variants. In addition, DL improves model accuracy and reduces the complexity in dimension, time, and computational resources [67].

Spam detection is an approach to identify unsolicited and unwanted messages sent electronically to a large number of recipients by someone they do not know of [68]. Spam can be classified according to multiple communication media, namely email spam, SMS spam and social spam. Email spam fills up the user's mailbox with undesired messages and unimportant emails. Meanwhile, SMS spam is usually distributed among mobile devices. Social spam has become more and more popular with the advent of the Internet and online social network, impacting social media users [69]. However, problems caused by spam messages can be prevented by spam classification and filtering. DL techniques can improve the effectiveness of spam filtering methods by developing and implementing spam detection systems [59], [70].

Phishing detection is another domain in cybersecurity that DL proved to be an effective solution [59], [61], [70], [71]. Similar to spam, phishing can also be spread through several communication channels, such as email, SMS, website, online social network, etc [8]. However, phishing has malicious intentions and is typically more dangerous as compared to spam. Spam emails, for instance, are delivered to users regardless of their consent and are often used for advertising purposes. Spam emails consume users' time, devices' memory and network bandwidth. On the other hand, phishing emails impose higher risk since they involve stealing sensitive information which can lead to huge financial loss [72]. DL efforts toward phishing detection have become a primary focus of this study due to the severe damages that phishing can potentially cause and the benefits that DL offers to mitigate these damages.

2) CLASSIFICATION BY TECHNIQUES

DL techniques can be classified into five categories: discriminative (supervised), generative (unsupervised), hybrid, ensemble, and reinforcement as illustrated in FIGURE 8 [23], [59], [61], [73], [74]. A list of abbreviations for various DL techniques is provided in TABLE 4.

Discriminative DL models are used for supervised learning to distinguish patterns for classification, prediction or recognition tasks [23]. They work with labeled data to predict output by observing the inputs [75]. Popular discriminative DL models are Convolutional Neural Network (CNN), Multilayer Perceptron (MLP), etc. [74]

Generative DL models are used for unsupervised learning to learn automatically from an unlabeled dataset [23]. Generative architectures leverage the advantages of data synthesis and pattern analysis to model the input data and generate random samples similar to the existing ones. They can describe the correlation among the input data's properties to achieve better feature representation [59]. Examples of generative DL models include Autoencoder (AE), Restricted Boltzmann Machine (RBM), Deep Belief Network (DBN), etc. [73], [74].

Hybrid approach combines both discriminative and generative modes in a single architecture and therefore, benefits from both models [76]. Generative models are used as subcomponents for two purposes in a hybrid DL architecture, either parameter learning through feature representations or improved optimization to generate better discriminative

TABLE 4. List of Acronyms for DL Techniques.

Acronym	Explanation
SCNN	Singular Convolutional Neural Network
MCNN	Multi-convolutional Neural Network
VCNN	Variants of Convolutional Neural Network
ACNN	Acoustic model of Convolutional Neural Network
LWCNN	Light Weight Convolutional Neural Network
FFNN	Feedforward Neural Network
VAE	Variational Autoencoder
DAE	Denoising Autoencoder
SDAE	Stacked Denoising Autoencoder
DRBM	Deep Restricted Boltzmann Machine
ARNN	Acoustic model of Recurrent Neural Network
BRNN	Bidirectional Recurrent Neural Network
BiLSTM	Bidirectional Long-Short Term Memory
BiGRU	Bidirectional Gated Recurrent Unit
DRL	Deep Reinforcement Learning

models [75]. DNN and GAN are examples of DL techniques belong to this category.

Ensemble deep learning (EDL) models can be constructed by organizing multiple individual DL algorithms in parallel or sequential. There are two types of EDL architectures, namely homogeneous and heterogeneous [74]. A homogeneous EDL model combines DL techniques of the same genre (CNN-CNN, LSTM-LSTM, GRU-GRU, etc.). Meanwhile, a heterogeneous EDL model integrates DL techniques from different genres (CNN-LSTM, CNN-RNN-MLP, etc.). The theory behind EDL is that each individual DL algorithm has its pros and cons. EDL architectures join their advantages and resolve their disadvantages, provide better results, and prove to be more effective in phishing detection [70].

Reinforcement learning is an adaptive learning approach used to obtain proficiency for optimal behavior. The basic concept of reinforcement learning involves an agent who performs an action based on trial and error, and interacts with an unknown environment that returns feedbacks through numerical rewards [77]. Current research has shown a growing interest in deep reinforcement learning (DRL) [77], and it is anticipated that DRL will become one of the promising directions in the near future, as it has not been fully explored and experimented for designing a phishing detection model [59]. Examples of DRL are Multi-task Reinforcement (MTR), Multi-agent Reinforcement (MAR), Asynchronous Reinforcement (AR), Q-learning Reinforcement (QR), etc. [71].

Most of the existing literatures classified DL techniques into three main classes: discriminative, generative and hybrid [23], [59], [75], [76]. However, they did not include the ensemble DL and deep reinforcement learning approaches. The taxonomy proposed in this study introduces these two additional categories into the classification of DL techniques since they play essential roles in solving various security issues, including phishing attacks detection [70], yet their potential have not been fully exploited and need to be further examined [23]. On the one hand, ensemble DL methods merge the advantages of individual DL algorithms, cure their disadvantages, and improve the overall performance of the phishing detection model. Ensemble DL is different from the hybrid approach because hybrid methods combine supervised and unsupervised learning, while ensemble models are formed by stacking different DL algorithms. For instance, DNN is a hybrid DL technique, but DNN-SAE is an ensemble DL model. On the other hand, deep reinforcement learning has been implemented in a wide range of applications, such as pattern recognition, autonomous navigation, air traffic control, defense technologies, etc. [59]. As a result, it has opened a promising direction for research in the cybersecurity domain [70], including the detection of phishing attacks in the cyber environment.

Moreover, various frequently-used DL techniques for phishing detection were identified based on the analysis of 81 selected articles using SLR approach, as shown in FIGURE 9. LSTM and BiLSTM are the most popular DL techniques with a percentage of 34%, followed by CNN with almost equivalent distribution (30%). DNN and MLP contributed the same portion of 8%, while only 1 out of 10 articles implemented GAN or DRL in their studies. LSTM and CNN have been widely used in previous research partly because of their numerous benefits. LSTM models solve the vanishing or exploding gradient issues exist in the traditional recurrent neural network are suitable for handling time-series sequence data [21]. Meanwhile, CNN models are best suited for highly efficient and fast feature extraction from raw and complex data. CNN architectures provide more promising and robust results because they reduce the network complexity and speed up the learning process [61]. LSTM and CNN are well fitted for phishing webpage detection due to these benefits, as phishing websites contain multi-dimensional data such as text, images or both. In general, each DL algorithm has its strengths that can be leveraged, and weaknesses that need be improved. Therefore, it is essential to analyze the pros and cons of individual DL mechanism to build an effective model to detect phishing. Appendix C listed the advantages and disadvantages of several DL algorithms used in the previous studies.

Appendix D to Appendix Q provide details of DL techniques used in the literature. These DL algorithms are classified according to their application, platform, and dataset. It is observed that DL has been used to detect website phishing or email phishing. In addition, DL was also utilized for either feature extraction or classification purpose. Platforms that were used for the design of these DL models include Matlab, JavaScript, C++, Weka, Python, and RStudio. Last but not least, the datasets used for the implementation of these DL algorithms were also analyzed to examine their performance in detecting phishing websites and emails, which will be discussed in the next section.

3) CLASSIFICATION BY DATASETS

An in-depth examination of 81 reviewed papers also indicated that although phishing attacks can be conducted through different types of media (voice, SMS, online social network,

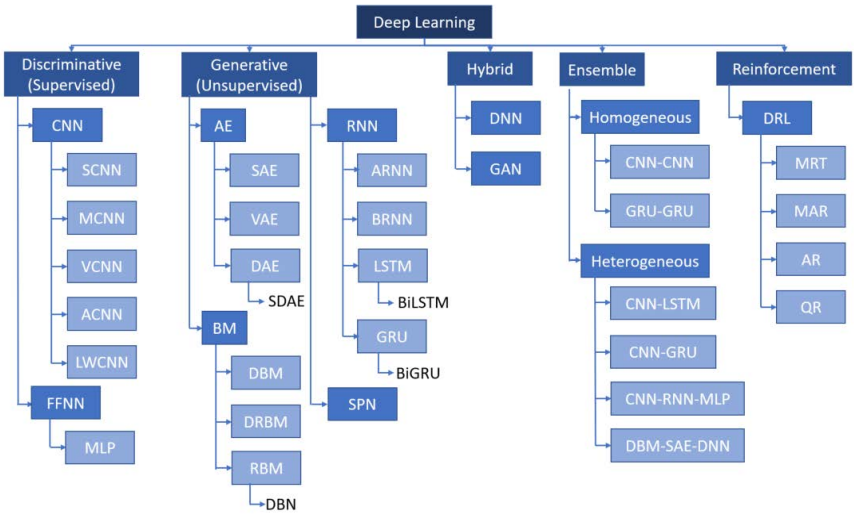

FIGURE 8. Taxonomy of DL techniques.

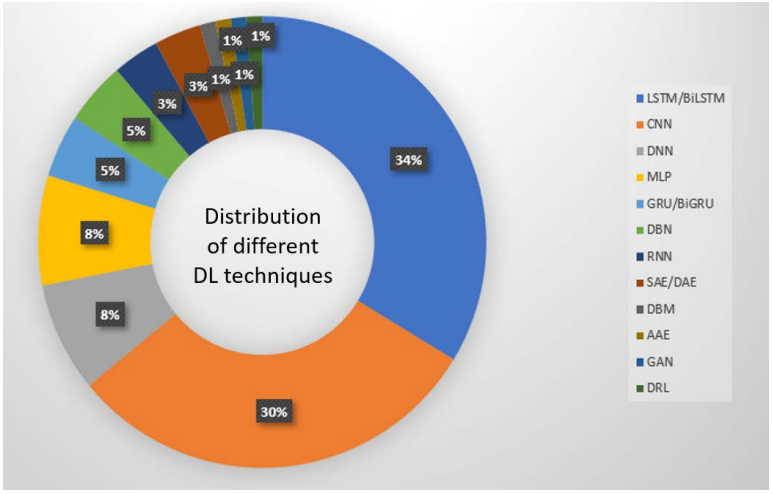

FIGURE 9. Distribution of different DL techniques.

etc. [8]), website and email are the most common phishing attacks in cyberspace. Among the reviewed articles selected for this research study, most of them belongs to the former group (47 articles), while a minority of them fit into the latter category (12 articles). In addition, different datasets are used for website and email phishing.

a: EMAIL PHISHING DATASET

Since emails typically hold private and confidential information, datasets for email phishing are limited. This restriction also applies to the publicly available ones [70]. Email phishing datasets contain two types of email, namely ham and spam (or phishing) [78]–[80]. FIGURE 10 displays the distribution of datasets for email phishing among 81 selected papers for this study. Spam Assassin and Enron are the most widely-used datasets for email phishing, with an equivalent distribution of 19%. Spam Assassin contains both ham and spam emails obtained from the SpamAssasin project [81], while Enron consists of more than 500 thousand emails generated by 158 employees from the Enron Corporation [80].

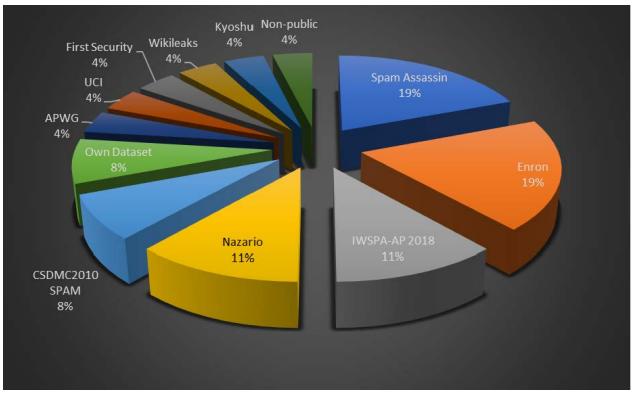

FIGURE 10. Distribution of datasets for phishing email.

Other popular datasets are from the First Security and Privacy Analytics Anti-Phishing Shared Task (IWSPA-AP-2018) and Nazario phishing corpus, with both occupies 11% of the total email phishing datasets. Email corpus provided by the organizer of IWSPA-AP-2018 competition consists of two sub-tasks to build and train a classifier to distinguish

TABLE 5. List of publicly available datasets for email phishing detection.

No	Dataset	Type		Website	Created by	Email part	Dataset size	Related work
		Ham	Spam					
1	Spam Assassin	✓	✓	https://spamassassin.apache.org/	Apache Spam Assassin Project	Body + Header	>9,000	[80], [81], [83]–[85]
2	Enron	✓		https://www.cs.cmu.edu/~enron/	CALO Project	Body + Header	>500 thousand	[78]–[80], [83], [84]
3	IWSPA-AP-2018	✓	✓	https://dasavisha.github.io/IWSPA-sharedtask/	Organizers of IWSPA 2018 competition	Body + Header	Ham: 9,174 Spam: 1,132 Total: 10,306	[83], [86], [87]
4	Nazario		✓	https://monkey.org/~jose/phishing/	Jose Nazario	-	-	[80], [81], [85]
5	CSDMC 2010 SPAM	✓	✓	https://github.com/jdwilson4/Intro-to-Machine-Learning/tree/master/Data/SPAMData	Organizers of data mining competition	Body + Header	Ham: 1,378 Spam: 2,949 Total: 4,327	[88], [89]
6	APWG		✓	https://github.com/APWG/ecx	Anti-Phishing Working Group	Body	-	[79]
7	UCI		✓	https://archive.ics.uci.edu/ml/datasets.php	University California Irvine	-	-	[78]

ham or phishing emails from spam and legitimate ones. The first sub-task contains emails with only the body part, while the second sub-task comprises of emails with both body and header [56], [68].

The Nazario phishing corpus was created by Jose Nazario, and contained only phishing emails [80]. Other datasets used for email phishing detection involve CSDMC2010 SPAM, APWG, UCI, etc. A list of the most common datasets to detect phishing email is provided in TABLE 5.

b: WEBSITE PHISHING DATASET

Based on the analysis of 81 selected papers, the most frequently-used datasets for website phishing detection include Phish Tank, Alexa, DMOZ, UCI, and Common Crawl. Phish Tank is the most popular depository that provides phishing URLs to train a classifier to differentiate between malicious and genuine websites (FIGURE 11). A majority (34%) of the articles used Phish Tank as their dataset to collect phishing URLs, followed by Alexa and DMOZ (9% and 8%, respectively), two databases provide legitimate URLs for training and testing purposes [75], [82]. UCI is another common repository consisting of both malicious and legitimate URLs for machine learning and phishing detection [42]. Meanwhile, Common Crawl is a corpus of web crawl data comprised of only legitimate sites [48]. A list of the most popular datasets for website phishing detection is provided in TABLE 6.

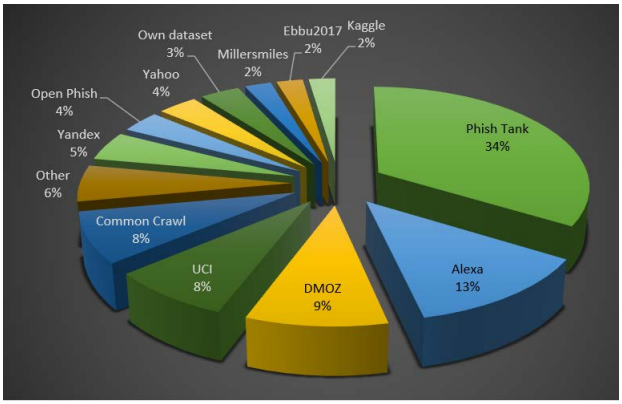

FIGURE 11. Distribution of datasets for phishing website.

IV. CURRENT CHALLENGES

This section analyzes the current issues found in the literature and proposes possible solutions to solve the challenges identified in the study, and to answer RQ3.

A. FEATURE ENGINEERING

Traditional ML algorithms, as discussed in the previous section, require manual feature engineering to extract features for phishing detection purposes [20]. The feature extraction and selection process are based on experiment and professional knowledge, which is tedious, labor-intensive, and susceptible to human errors [22]. Some researchers select features according to their own experience, while others examine different statistical techniques to determine the best-reduced set of optimal features [21]. Handcrafted feature selection is often done manually and still requires much labor and domain expert, limiting the performance of phishing detection.

B. ZERO-DAY ATTACKS

Classical ML techniques still suffer from the lack of efficiency in detecting zero-day phishing attacks [10]. The detection model must explore new behaviors and be able to dynamically adapt to reflect the changes in newly evolving phishing patterns to handle these types of attacks effectively. The majority of the existing classification techniques are unable to explore these new behaviors and incapable of adapting themselves to reflect the changes in the environment [92]. As a result, they fail to detect unknown or newly

TABLE 6. List of publicly available datasets for website phishing detection.

No	Dataset	Type		Website	Created by	Dataset size	Related work
		Legitimate	Phishing				
1	PhishTank		✓	https://www.phishtank.com/	OpenDNS	-	[22], [34], [37]–[40], [43]–[48], [50], [52], [54], [77], [82], [90]–[100]
2	Alexa	✓		https://www.alexa.com/	Amazon	-	[22], [34], [44], [46], [49], [82], [92], [93], [95], [97], [101]
3	DMOZ	✓		https://dmoz-odp.org/ https://dmoztools.net/docs/en/rdf.html	Open Directory Project	-	[43], [58], [82], [88], [91], [93], [96]
4	UCI	✓	✓	https://archive.ics.uci.edu/ml/index.php	University California Irvine	-	[43], [80], [97]–[101]
5	Common Crawl		✓	http://commoncrawl.org/	Common Crawl	-	[22], [39], [47], [48], [52], [90], [91]
6	Yandex	✓		https://yandex.com/dev/xml/	Yandex	-	[22], [43], [77], [99]
7	Open Phish		✓	https://openphish.com/	Open Phish	>13,000	[46], [93], [95]
8	Yahoo	✓		http://dir.yahoo.com/	Yahoo	-	[38], [45], [50]
9	Kaggle	✓	✓	https://www.kaggle.com/datasets	Kaggle Inc.	>50,000	[97], [102]

evolved phishing attacks. However, DL algorithms can detect zero-day attacks more efficiently [73].

C. DL ALGORITHM

There are many different DL algorithms and each of them has particular characteristics suitable for some specific applications. For example, CNN architecture provides better results when processing two-dimensional data with grid topologies, such as images and videos, due to the high correlation between pixels and neural networks [21]. Meanwhile, RNN is more suitable for sequential data, natural language and text processing [58]. In addition, more attention has been paid to supervised DL, yet the main disadvantage of discriminative learning is that it requires a massive amount of labelled data; collecting them is very costly and time-consuming [9], [59]. Therefore, it is challenging to choose the right algorithm best suited for a target application in the context of cybersecurity. Selecting an inappropriate algorithm might produce unpredictable outputs, leading to a waste of efforts and affect the model's effectiveness and accuracy [70].

D. COMPUTATIONAL CONSTRAINTS

Each stage in the phishing detection model, like data preprocessing, feature selection and classification, adds an extra level of computational complexity to the overall model. The computation complexity increases as neurons and layers in deep neural network architecture increase [58]. The use of Graphics Processing Units (GPUs) to accomplish maximum operations in minimum amount of time makes DL models more expensive to build [105]. This problem magnifies when new data arrives and model retraining is required [9]. Thus, computational complexity is one of the major issues in DL, and it is a challenging task for future researchers to build an effective phishing detection model with less computational resources [106].

Recent research by MIT suggested that the DL model's computational requirements have been growing significantly, which exceeds the ability that specialized hardware can handle. Additional enhancement will soon be needed since the development in hardware is slower than the improvement in DL computing power, which limits DL models' performance. Furthermore, complex DL models using GPUs and TPUs in their implementation have certain effects on the environment and energy consumption. The amount of carbon dioxide emitted from such models is approximately five times an average car's lifetime emission. This suggests that future researchers should start looking for alternative techniques more computationally efficient than DL [21].

E. DATASET

Dataset issues can be divided into four categories: availability, diversity, recency, and quality [10]. Firstly, there are limited resources for phishing email datasets since some organizations hesitate to share their information due to privacy issues [70]. Other publicly available phishing website datasets contain dead, duplicate, or incomplete links, which cannot be accessed by users. Furthermore, there are individual or organizations encountered or conducted research on phishing attacks, but did not submit to the crowd-sourcing sites; hence, new phishing emails or websites are not made publicly accessible [10]. As a result, researchers and developers have difficulty in finding available datasets to work with. This can become a major obstacle because DL requires a significant amount of data to train deep neural networks [21].

Secondly, the diversity of datasets is also an essential factor that can hinder the performance of DL models. If features in the datasets are not extensive and representative enough, the DL model will not have a great generalization ability [7]. DL models trained on datasets that only contain patterns of known attacks generally will not perform well when facing

new phishing patterns. DL models will not be able to detect or classify them correctly, especially when attackers contaminate the datasets with adversarial samples (adversarial attacks [9]) to deceive the model into learning phishing attacks as legitimates (active attackers [7], [10]). This will affect the robustness of the underlying DL model [60].

Thirdly, there are different datasets made publicly available to train DL models, but not all of them are up-to-date (lack of recency). Moreover, issues caused by limited resources also lead to model training and validation on old and obsolete data. DL models trained on such datasets might fail to detect modern phishing patterns and produce low detection accuracy [7].

Finally, the efficiency and effectiveness of DL models depend on the nature and characteristics of the input data (data quality). If the input data contain ambiguous, missing or meaningless values and outliers, DL models might produce incorrect output results. Non-representative, poor quality, irrelevant features, and imbalanced datasets can lead to low detection accuracy [7], [10]. Therefore, it is crucial to have relevant and high-quality input data to produce better outcomes in DL models. In other words, one potential solution is to improve the existing pre-processing techniques or propose new data preparation methods to enhance the effectiveness of DL models in the phishing detection domain [70]. Significant improvement in model performance can sometimes be achieved from higher quality data rather than more sophisticated algorithms. Even though the cybersecurity community has recognized DL as a promising algorithm for detecting phishing attacks, there is still a lack of high-quality datasets exist in this field [107].

To sum up, DL generally requires significant datasets to achieve high detection accuracy. Thus, data resources containing a small number of instances, non-diverse data, outdated or highly imbalanced samples might cause overfitting problems [59]. Similarly, datasets comprising of old phishing attacks, do not represent real attack scenarios and behaviors, or do not possess real-time properties might not provide reliable performance results [23]. Models built on such datasets will suffer from the lack of efficiency, effectiveness, and accuracy in phishing detection.

F. PARAMETER OPTIMIZATION

The parameters in DL models include, but are not limited to, the number of hidden layers in the neural networks, number of neurons in each layer, number of epochs, type of activation function, type of optimizer, learning rate, and dropout rate, etc. [59]. There is no standard guideline for an optimal set of parameters that can produce the best performance accuracy. Researchers usually need to conduct a series of experiments to fine-tune these parameters [34], [39], [41], [42], [44], [75], [97], [108]. This process is time-consuming and requires much effort.

G. EVALUATION METRICS

A set of performance metrics need to be measured after training the DL model to evaluate the effectiveness and efficiency of the underlying algorithm. The most common computational metrics are False Positive Rate (FPR), False Negative Rate (FNR), accuracy, precision, recall, etc. [1], [59], [73]. Sufficient evaluation metrics are crucial in assessing the performance of a phishing detection system. A single metric is not representative of the high performance of a DL algorithm, but computing all the performance measures are not always the case in some of the studies [59]. In addition, appropriate evaluation metrics also play a vital role that need to be considered [7]. Especially in the case of imbalanced datasets, accuracy and error rate are not entirely suitable for performance evaluation. Instead, other metrics, such as Receiver Operating Characteristic (ROC) curve, and Area Under the Curve (AUC), are more desirable [10].

H. INFERENCE JUSTIFICATION

One of the main advantages of DL over ML is its ability to explore the hidden correlation between features, learn and make intelligent decisions on its own by building complex algorithms in multi-layer neural networks [21]. However, the major drawback of DL models is its inability to justify the inference it makes [42]. Since numerical weights represent the underlying knowledge inside DL models, it is not possible to explain the logic behind the assumptions, decisions, and conclusions that a neural network makes [10]. What DL models learn from the data is not interpreted, and DL models' internal operation is almost unknown, like a black box. Consequently, it would be difficult to understand the correlation between the input features and the output results [59]. The problem caused by inference justification becomes more challenging when it comes to solving errors. When there is an error in a DL model, it is extremely hard to diagnose and identify the main cause of the underlying error, since the output results are almost uninterpretable [58], [63]. Therefore, it is suggested that the causes of the attacks should be analyzed thoroughly to design an effective DL model for cybersecurity applications [71].

Neural networks are considered as black boxes since their internal operations are unknown to humans [9]. DL algorithm consists of multiple processing layers to learn data representation through multi-level abstraction. Yet, human experts have not determined the layer of abstractions but are learned from input data through some generic learning algorithm [109]. Since it is not possible to give a reasonable justification about the relationship between inputs and outputs in neural networks, more attention should be paid to the underlying mechanism inside DL models, even though DL algorithms practically perform well and have caught much interest among recent researchers [60].

I. BATCH LEARNING

Batch learning refers to the learning algorithms in which an entire training dataset was obtained prior to model training. Batch learning is used in both traditional ML and DL techniques since it offers ease of use and implementation. Nevertheless, batch learning still has several drawbacks, such

as expensive retraining, high memory and computational constrains, inability to detect newly evolving threats, and poor adaptation to concept drift [9]. Online learning, however, can solve the problems caused by batch learning and suggests a promising direction for future research in the phishing detection domain.

J. TIME COMPLEXITY

DL requires a significant amount of data and a substantial amount of time to train the model [103]. Datasets used for training neural networks usually contain millions of samples [45], [46], [84], [94], [111]. As a result, they need longer time to train the model to obtain high-performance accuracy.

Another factor that might delay the model's training time is limited processing and storage facilities [73].

Time complexity is an issue in threats detection and detecting phishing attacks is not an exception [73]. The existing detection techniques have been mainly developed for batch processing and not for real-time detection. As a result, traditional ML approaches lack efficiency in classifying phishing attacks in real-time scenarios. DL, on the other hand, can solve the problem caused by time complexity by using GPUs in its design and implementation [112]. In addition, big data technologies, such as Apache Spark or Hadoop, can help reduce the time complexity since they offer real-time processing capabilities [113].

Phishing webpages are short-lived; thus, there exists a need for real-time detection of phishing websites [10]. Phishing attacks are normally deployed in a short duration of time, usually in a few days or weeks, making it difficult for security experts to detect. The detection mechanism needs to be fast to capture zero-day attacks because the time-scale of phishing attacks are short. Therefore, real-time detection is a crucial part of a practical phishing detection system [7], [9].

K. BIG DATA CHALLENGES

The big data era imposes new challenges for phishing detection [9], especially when classical machine leaning techniques cannot handle a significant amount of data. DL, in contrast, can overcome this issue since it can deal with big data and perform better when the dataset size is getting more significant. DL when combined with big data, has the ability to manage and analyze a large amount of information in a short amount of time. However, the training process of DL models on such a tremendous amount of data with a single processor is not an easy task. Although GPUs and TPUs have been used to improve the training speed and reduce the training time of DL algorithms, the overall process still consumes a significant amount of time and needs high data processing capabilities [109].

All of the problems mentioned above are mapped to the existing DL techniques and classified into three groups: solved, partly solved, and not yet solved as shown in TABLE 7. For example, dimensional complexity is the major limitation of CNN models. However, this issue can be partly resolved by implementing dimensionality reduction techniques, such as RBM, DBN, AE or DAE [20], [23], [67], [70], [74]. In addition, vanishing or exploding gradient is a well-known drawback of RNN algorithm, which was overcome by its variants, namely LSTM and GRU. Even though the vanishing gradient cannot be completely resolved since it still occurs in long sequences, LSTM solves the problems of long-term dependencies and performs better than traditional RNN models [70], [107], [114]. In general, the problem of manual feature engineering is eliminated in DL, since DL algorithms extract features automatically from raw data without the need of prior knowledge. Although DL proved to be a promising solution for detecting zero-day phishing attacks, this issue has not been completely resolved as phishing tactics have evolved rapidly with the recent development of technologies. Other common issues among DL algorithms are high computational cost and manual parameter optimization. The optimal set of parameters generate the highest detection accuracy is still debatable. Last but not least, all current DL architectures lack of inference justification where the internal operation of DL models is unexplainable until recently. The following section proposes possible directions for future research based on the identified research gaps to help solve some of these problems.

V. FUTURE DIRECTIONS

This section provides an answer to RQ3 in which future research directions are suggested from the perspective of DL and act as a guideline for researchers and developers to mitigate phishing attacks in cyberspace.

A. CHOOSING THE RIGHT APPROACH

Since manual feature engineering can cause biases, DL algorithms becomes an alternative that can improve the efficiency of phishing detection. It was proven that DL models without manual feature extraction could perform better than traditional ML with feature extraction [76]. Moreover, classical ML methods are unable to explore the hidden correlation between these features. Whereas DL algorithms can extract information from the tremendous amount of data, find the correlation in the extracted data and handle multiple feature selection autonomously [73]. DL algorithms appear to be a promising solution since they avoid handcrafted feature selection, third-party service dependency, overcome false positive rate and improve detection accuracy [66]. DL has not been extensively studied in the phishing detection domain despite all of these advantages. Therefore, more attention should be paid to DL as it is a potential research direction in the near future [10].

B. SELECTING AN APPROPRIATE DL MODEL

There is a variety of DL techniques used to detect phishing attacks in the cyber environment. It is extremely important to choose the right algorithm for a specific application as it will affect the final outcomes. Therefore, it is essential for researchers to understand the reasons behind selecting a certain DL architecture, as failing to do so might result in

TABLE 7. Issues of Existing DL Techniques.

No	Issue	Technique											
		DNN	CNN	RNN	MLP	LSTM	GRU	RBM	DBM	DBN	AE	GAN	DRL
1	Manual feature engineering	●	●	●	●	●	●	●	●	●	●	●	●
2	Detect zero-day attacks	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐	◐
3	Hight computational cost		○		○			○		○	○		
4	Dimensional complexity		○					◐		◐	◐		
5	Vanishing/ exploding gradient			○		◐	◐						
6	Long-term dependencies			○		●	●						
7	Ability to handle sequential data			●		●	●						
8	Ability to handle image data		●									●	
9	Long training time requirement	○	◐	○		○	○			◐			
10	Lack of feature representation							○	●	●			
11	Parameter optimization	○	○	○	○	○	○	○	○	○	○	○	○
12	Inference justification	○	○	○	○	○	○	○	○	○	○	○	○

● Solved ◐ Partly solved ○ Not yet solved

an ineffective phishing detection model. For instance, it is expected that unsupervised DL will become more and more popular in the near future [9]. Semi-supervised learning is another potential research direction besides unsupervised to handle the massive amount of unlabeled data in cyberspace. Most of the current DNN models are now using unsupervised layer-wise pre-training and supervised fine-tuning, which is computationally expensive. However, suppose supervised and unsupervised learning can be combined in a powerful semi-supervised DNN model. In that case, there will be no need to have a separate layer-wise pre-training phase. As a result, it can increase detection accuracy while minimizing the computational cost [59].

In addition, wrong choices of DL design or implementation, based on low level of maturity on applying DL techniques, would lead to biased classification results. DL approaches offer a wide range of possibilities that have not yet been fully exploited. Researchers and developers would fail to explore the full potential of DL architectures by overlooking these. For instance, DL models are capable of capitalize multimodal (heterogeneous) input data and handle multiple classification tasks in addition to single-modal and binary classification [115]. On the one hand, multimodal DL-based classifier can learn the hierarchical representation of all the available modalities in the input data automatically, instead of performing the manual feature engineering process on a specific modal. On the other hand, multitask DL approach can reduce computational overhead, thereby limiting redundancy by sharing part of the feature engineering procedure. This improves generalization and provides better classification results, which would help with solving the task of phishing detection.

C. EMPLOYING OTHER COMPUTATIONALLY EFFICIENT TECHNIQUES

The training process in DL models is performed on a significant dataset and consumes many computational resources. Transfer learning can be used to overcome this problem by detecting phishing attacks with the same patterns. This can be done because transfer learning utilizes pre-trained models to solve similar problems and then train only the fully connected layer for a new classification task without building classifiers from scratch for different types of phishing attacks. Transfer learning can be applied without biases from features; only adequate data of the new attack is sufficient for the task [7], [106].

Besides transfer learning, lifelong learning and online learning can also be applied to solve the problem of computational constrains [9]. Online learning is a scalable learning algorithm that learns from data, make updates and predictions sequentially. In online learning, data is treated as stream of instances, making it more efficient than traditional batch learning.

Additionally, computational costs can be reduced by implementing distributed computing and distributed algorithms. Different jobs are distributed among several machines in the hybrid network to speed up the process and improve performance efficiency. Big data technologies, such as Apache Spark, can be applied to handle this task by utilizing

the parallel computing capabilities to process the data with feasible computational resources [73].

One of the major limitations of phishing detection models is resource constraint, and DL algorithms are just computationally expensive. It is suggested that edge or fog nodes can be used to offload the computational constraints for effective phishing detection without increasing its computational cost [21]. Edge computing offers a more scalable platform of computational processes and power storage. Leveraging edge computing will facilitate handling this problem by allocating the computation process to several resources over the cloud [23].

Another approach to minimize computational cost is to integrate neuromorphic computing with DL. Neuromorphic computing is different from deep neural networks in both structure and principle. All neurons are activated by the activation function in the current deep neural networks, for example, Rectified Linear Unit (ReLU), Sigmoid, Tank, etc. However, unlike neural networks, all neurons are not activated every time in neuromorphic computing. This allows the model to achieve higher efficiency and lower power consumption. Neuromorphic computing help reduce the need for software and hardware development, leading to an increase in computational speed and a decrease in computational complexity [107].

D. SELECTING, LABELLING AND TRAINING DATASET

The efficiency and effectiveness of phishing detection solutions depend on the selection, labelling, and training of a dataset. First, some datasets are not available, non-diverse, out of date, or highly imbalanced. Thus, it is essential to select a recent and balanced dataset that contains various phishing patterns to detect newly evolving attacks in the live environment [21]. Second, supervised ML techniques required labelled data for training, yet the amount of labelled data is limited as compared to all the available data on the web. Therefore, researchers can apply active learning or crowdsourcing techniques, in which individuals and organizations can label and share malicious URLs, to handle the difficulty in acquiring labelled data or learning with limited amount of labelled data [9]. Third, pre-trained detection models might fail to handle new types of attacks once the phishers modify the nature of malicious websites or URLs [10]. Hence, retraining on a more recent dataset is required to fight against active attackers when testing data contains different characteristics from training data [7]. Furthermore, adversarial trainings can be used to handle adversarial attacks by minimizing the negative influence caused by monotonous samples or polluted data on DL algorithms. Combining DL with reinforcement learning is another possible solution, although it is unlikely to completely avoid adversarial attacks [60].

In addition, researchers can either increase the sample data or reduce the data dimension to solve the unbalanced dataset problem. On the one hand, small datasets can cause biases and suffer from a lack of generalization of new phishing patterns. It is possible to make the sample datasets balanced by increasing the sample data. On the other hand, training on significant datasets is also a challenging and time-consuming process. In this case, the dimensionality reduction technique can improve the performance accuracy and reduce computational complexity [117].

E. FINE-TUNING HYPER-PARAMETERS

It is essential to fine-tune several parameters in the DL architecture to build a robust and competent model for phishing detection. Fine-tuning is a process to optimize the performance of a training model by changing the number of hidden layers, neurons, epochs, learning rate, etc., in the neural network. This process aims to obtain the optimal combination of parameters that yield the best performance accuracy. Researchers can follow a set of pre-defined rules or formulas to calculate these values or narrow down the range of possibilities for these parameters. Nevertheless, there will exist some rules that are not always applicable or feasible in specific scenarios. Researchers need to examine all different combinations of parameters as much as possible in such circumstances, and choose the optimal parameter setting for a neural network with the best output results. Besides, a self-organizing neural network is another option for fine-tuning parameters in DL models. This technique allows the network to learn incrementally by adding or removing neurons according to different criteria [59].

F. PICKING THE BEST MEASURES

Another concern that needs to be considered in future research is choosing the appropriate metrics to evaluate the performance of phishing detection models. Researchers and developers must be careful in selecting performance metrics for model evaluation in highly imbalanced datasets. It might not suitable to use Accuracy, Precision, Recall, and F1-Score for class-imbalance issue to assess the effectiveness of the phishing detection systems [10]. Conventional metrics like accuracy cannot capture the true performance of a detection classifier in the case of imbalanced data. Instead, confusion matrix and Areas Under the Curve are more desirable. Other metrics made for imbalanced dataset are Geometric Mean (G-Mean), Matthew's Correlation Coefficient (MCC), or balanced detection rate, etc. [7], [10]

G. EMPLOYING EXPLAINABLE NEURAL NETWORK

It is advisable to design and implement a DL expert system to generate knowledge automatically from training data and to overcome the problem of lack of inner explanation in deep neural networks [99]. The refined rules are extracted from a trained neural network and then is replaced with the knowledge base of an expert system by combining these two methods in a hybrid model. The neural network will become more convincing and reliable as its internal operations are explainable.

Several efforts have been made to help reveal the internal interpretation of DL algorithms [118], [119]. However, these techniques were applied in different research domains

(discriminative image localization, depression recognition from facial images) and have not been employed for cybersecurity purposes. When applied, explainable neural work can potentially assist security experts in determining the input conditions under which output is produced. Especially in the cybersecurity domain, understanding the output results of a cyber threat detection model would give security experts a valuable insight into preventing and mitigating such cyber threats.

H. INTEGRATING VARIOUS TECHNIQUES IN A HYBRID MODEL

Another future direction is to combine different DL techniques in a hybrid approach to gain optimized performance accuracy in phishing detection. The theory behind this method is that each individual DL algorithm has its pros and cons. We can merge their advantages and resolve their disadvantages by integrating different DL techniques in a single approach to provide a more robust model for detecting phishing attacks [70].

I. DEVELOPING A ROBUST, SCALABLE AND FLEXIBLE PHISHING DETECTION SYSTEM

Phishing attacks are continuously evolving with the advancement in information technologies, as phishers try to come up with a countermeasure for every new solution that security experts suggest. As a result, it is essential to have a robust detection system with a set of features that go beyond the common attacks, and a diverse, recent and high-quality dataset for model training [7]. Researchers should train the DL model on one dataset and test on different data to ensure the robustness of phishing detection systems. This is also known as generalization experiment, or cross-domain system testing, to verify the performance of a phishing detection model in classifying various types of attacks [10]. Since phishers always change their attacking tactics to bypass the defense mechanism, model retraining alone might not be sufficient to cope with newly emerging attacks. Therefore, a robust phishing detection system is a system with high adaptability, which can adjust to reflect the changes in the real-world environment, given the variety of phishing attacks, the newly evolving attacks types, and the numerous scenarios in which such attacks can happen [9].

Besides adaptability, scalability is another requirement for future phishing detection models. A phishing detection system should be able to handle millions of instances in the training data in the big data era. Researchers can employ more efficient and scalable learning algorithms, such as online learning, or efficient stochastic optimization algorithm, to meet the scalability requirement [9]. Moreover, big data technologies like Apache Spark and Apache Flink can process data in-memory. In-memory processing allows data to be analyzed in real-time, and real-time processing is extremely important, especially in detecting security threats. Incorporating DL and big data technologies will help to improve the performance and efficiency of security analytics [73].

It is crucial for a phishing detection system to be flexible enough for easy design, implementation, improvement, and extension, considering the complexity of phishing webpage classification based on DL [9]. The flexibility requirements include quick model update upon the arrival of new training data, being easy to change the classification model when needed, being flexible to be extended for model training to cope with new attack types, and finally being able to interact with users when required.

An example of a robust, scalable and flexible phishing detection system is an anti-phishing framework or web browser plug-in that can perform multiple tasks, such as detecting, preventing, and reporting, once a suspicious website is found. An ability to quickly report phishing attacks to the organization from the user's end is an essential feature that can be added to the existing phishing detection solutions. The time organizations lost on remediation after being attacked by cyber criminals can negatively impacts the productivity and profitability of their businesses. Therefore, it is vital to provide a feasible model that can detect and report phishing attacks as automatically and quickly as possible so that they cannot cause any further damage to the organizations. It is expected that in the future, an all-inclusive phishing detection system can be implemented in such a way that it can detect, report, and prevent malicious websites without requiring the user's involvement. When users are asked for their credentials or personal information, the developed framework or web browser plug-in should be able to check if the website is legitimate and notify the users beforehand. Therefore, a scalable and robust phishing detection solution is needed to perform website health checking during user browsing in the near future [8].

To sum up, many solutions have been proposed to detect phishing attacks, but there is no single solution to detect all attack types in the vast space of the cyber environment. Whenever researchers develop a new solution to fight against phishing attacks, phishers will take advantage of the vulnerabilities in the current solution and come up with a new attacking strategy to deceive the users. A list of current issues and challenges, together with their recommendation and future research directions are provided in TABLE 8, with the hope that it will contribute to the mitigation of phishing attacks evolving rapidly in recent years.

VI. EMPIRICAL ANALYSIS

This section provides an empirical analysis of several DL algorithms to manifest some of the current issues discussed above. First, the dataset and a list of features used in the experiment are mentioned. Then, the experiment setup is briefly described. Finally, the existing problems that DL is facing in phishing detection is highlighted from the experiment results.

A. DATASET

The dataset used for the experiment in this study was obtained from University California Irvine Machine Learning Repository (UCI) which has been widely used by various authors

TABLE 8. Issues, Recommendation and Future Direction.

No	Issue/Challenge	Recommendation	Future research direction	Reference
1	Feature engineering	-Extract features automatically from raw data	-Employ deep learning algorithm	[10], [21], [66], [73], [76]
2	Zero-day attacks	-Explore new phishing behaviors and dynamically adapt to reflects the changes in new phishing patterns	-Apply real-time retraining and online learning technique to detect newly evolving phishing attacks	[10], [21], [73]
3	DL algorithm	-Examine the characteristics of a specific DL architecture that is best suited for a particular application	-Verify the intension behind selecting a specific DL model	[9], [21], [58], [59], [70]
4	Computational constraint	-Require significant improvement to reduce computational cost -Look for other computationally efficient techniques than DL	-Apply transfer learning, edge computing, lifelong learning, online learning -Employ distributed computing by using bid data technologies -Integrate neuromorphic computing with DL	[9], [21], [23], [66], [73], [74], [76], [107], [116]
5	Dataset	- Train the model using available, up-to-date, diverse, and high-quality datasets	-Apply unsupervised, semi-supervised ML, online active learning, and crowdsourcing	[7], [9], [10], [21], [23], [59], [60], [70], [107], [117]
6	Parameter optimization	-Follow a set of pre-defined rules or formulas	-Examine all combinations of parameters -Apply self-organizing neural network	[59]
7	Evaluation metrics	-Select appropriate performance metrics to evaluate the system performance	-Use other metrics in highly imbalanced dataset: MCC, balanced detection rate, etc.	[1], [7], [10], [18], [59], [73]
8	Inference justification	-Generate and explain the unknown knowledge inside DL model -Analyze the underlying cause of attacks in detail	-Combine neural network with expert systems in a hybrid model -Employ explainable neural network	[9], [10],[20], [41], [58], [21], [42], [59], [63], [71], [109]
9	Batch learning	-Learn from data, make updates and predictions sequentially -Treat data as streams of instances	- Apply online learning	[9]
10	Time complexity	-Minimize the time complexity	-Require real-time detection -Implement GPU component in deep learning -Employ big data technologies	[7], [9], [10], [60], [73], [112], [113]
11	Accuracy deficiency	-Combine different DL techniques to gain optimized performance accuracy	-Integrate various DL algorithms in a hybrid model to improve detection rate	[21], [70]
12	Robustness, scalability, flexibility	-Build a robust, scalable, and flexible phishing detection system	-Conduct generalization experiment, or cross-domain system testing - Apply online learning, big data technologies	[7]–[10], [73]
13	Big data	-Analyze data in real-time by using in-memory processing technology	-Incorporate deep learning algorithms with big data technologies	[9], [73], [109]

in their research [83], [97], [99], [101]. The dataset consists of 11055 URLs, in which 6157 are legitimate and 4898 are phishing. The dataset was divided into two parts, 80% for training, 20% for testing, and contained a total of 30 features.

FIGURE 12 is a heatmap displaying the correlation matrix of the features. The correlation range is from -0.6 to 1, where 1 is the highest positive correlation and -1 is the lowest negative correlation. The closer to 1 the correlation is, the more positively correlated the features are. In other words, as one increases, so does the other. Specifically in this dataset, feature Favicon and Using Popup Window are highly correlated. No other feature in the dataset has a high correlation except for Favicon and Using Popup Window. Moreover, some features have a negative correlation, and others are positively correlated. Negative correlations mean one feature marks the URL as phishing, while the other does not [97].

B. EXPERIMENT SETUP

Various DL models were built in the experiment using Python programming language with Tensorflow on Google Colaboratory. Tensorflow is an end-to-end open-source platform for machine learning. It provides tools, libraries and resources, allowing researchers and developers to build, train, and deploy machine learning models. Google Colaboratory enables users to compile and execute python in their own browser. Google Colaboratory provides an interactive environment in which executable code, text, images, HTML, etc., can be combined in a single document. Codes are executed on Google's cloud servers, allowing users to leverage the power of Google's hardware. Plus, several DL models were built in this empirical study, including DNN, MLP, CNN, RNN, LSTM, GRU, and AE. Parameter settings for these DL architectures are listed in TABLE 9.

TABLE 9. Parameter Settings for Various DL Models.

Parameter settings	No of epoch	Batch size	Optimizer	Learning rate	Activation function in the hidden layer	Activation function in the output layer
Value	50	32	Adam	0.001	ReLU	Sigmoid

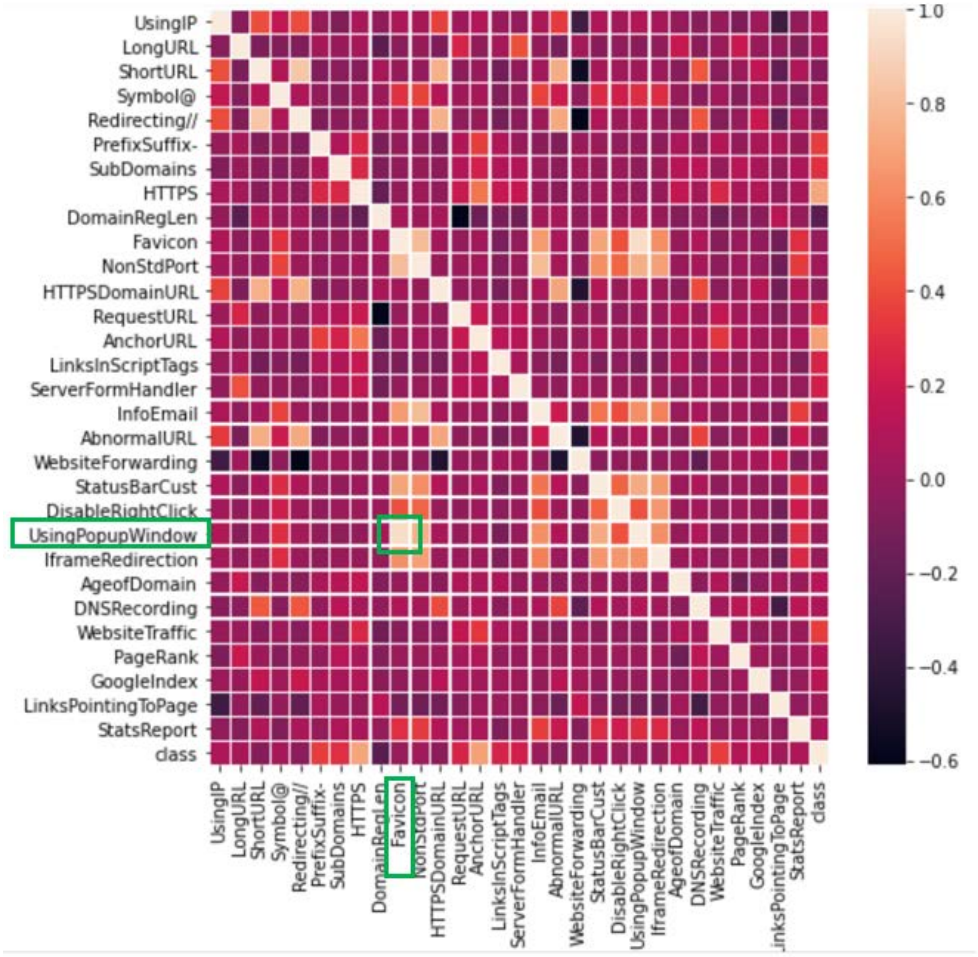

FIGURE 12. Correlation matrix of features.

TABLE 10. Performance Metrics of Various DL Models.

No	Model	FPR (%)	FNR (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC (%)	Accuracy (%)
1	DNN	3.80	3.48	96.52	97.13	96.83	99.16	**96.38**
2	MLP	5.07	3.18	95.95	96.82	96.38	99.13	**95.97**
3	CNN	3.66	5.58	94.42	97.40	95.89	99.23	**95.21**
4	RNN	3.88	4.85	95.15	97.00	96.07	99.26	**95.57**
5	RNN-RNN	9.12	2.57	97.43	92.30	94.80	98.86	**94.35**
6	LSTM	2.98	10.46	89.54	97.88	93.52	98.02	**92.49**
7	LSTM-LSTM	7.88	4.91	95.09	93.70	94.39	98.54	**93.76**
8	BiLSTM-BiLSTM	7.32	7.32	92.68	94.46	93.56	98.10	**92.67**
9	GRU	4.21	8.24	91.56	96.66	94.15	98.71	**93.49**
10	GRU-GRU	3.77	4.61	95.39	97.09	96.23	99.35	**95.75**
11	BiGRU-BiGRU	3.99	5.11	94.89	96.77	95.82	99.16	**95.39**
12	AE	5.95	5.92	90.63	94.08	92.32	90.90	**91.27**

C. RESULT AND DISCUSSION

It is essential to select a set of parameters with the best performance accuracy when building each DL model. These parameter settings can vary among different DL models, including the number of hidden layers in the neural networks, the number of neurons in each hidden layer, the

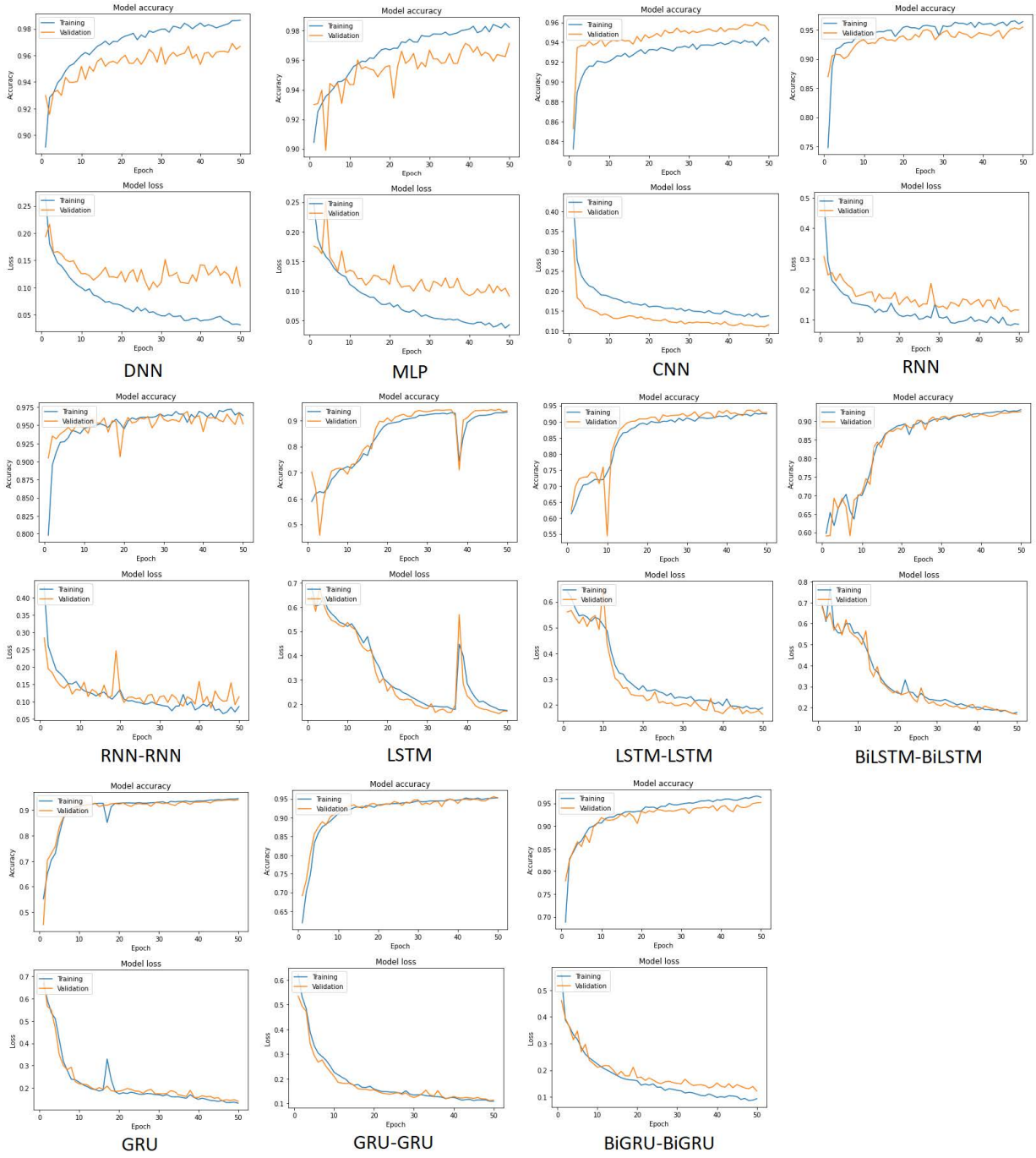

FIGURE 13. Accuracy and loss of various DL models.

number of epochs, batch size, type of optimizer, learning rate, type of activation function, etc. The same set of parameters was used in this research across all DL models just for the purpose of empirical analysis to highlight the current issues of DL in phishing detection. Fine-tuning will be added in future research to find the optimal set of parameters for each DL model that can produce the highest detection accuracy.

The loss and accuracy of various DL models during training and validation are illustrated in FIGURE 13. The accuracy for each DL model is shown in the upper graph, while the loss function is displayed in the lower plot. As the number of epochs grows, the accuracy starts to increase, while the loss function begins to decrease. The training accuracy, or training loss, is represented by a blue line, whereas the validation result is displayed in orange. A large gap between training and

TABLE 11. Selected studies for this slr.

ID	Author and year	Title	Cite	Document type	Journal Quartile	Journal Impact Factor	Ref
P1	Adebowale et al, 2019	Deep Learning with Convolutional Neural Network and Long Short-Term Memory for Phishing Detection	5	Conference paper	-	-	[90]
P2	Adebowale et al, 2020	Intelligent Phishing Detection Scheme Using Deep Learning Algorithms	4	Journal article	Q2	2.659	[91]
P3	Ahmad & Alsmadi, 2021	Machine Learning Approaches to IoT Security: A Systematic Literature Review	2	Review article	Q1	9.936	[21]
P4	Al-Ahmadi & Alharbi, 2020	A Deep Learning Technique for Web Phishing Detection Combined URL Features and Visual Similarity	-	Journal article	-	-	[120]
P5	Al-Ahmadi & Lasloum, 2020	PDMLP: Phishing Detection using Multilayer Perceptron	1	Journal article	-	-	[97]
P6	Aldweesh et al, 2020	Deep Learning Approaches for Anomaly-Based Intrusion Detection Systems: A Survey, Taxonomy, and Open Issues	57	Journal article	Q1	5.921	[23]
P7	Aljofey et al, 2020	An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network from URL	3	Journal article	Q2	2.412	[22]
P8	Al-milli & Hammo, 2020	A Convolutional Neural Network Model to Detect Illegitimate URLs	2	Conference paper	-	-	[38]
P9	Alotaibi et al, 2020	Mitigating Email Phishing Attacks using Convolutional Neural Networks	1	Conference paper	-	-	[81]
P10	Amanullah et al, 2020	Deep Learning and Big Data Technologies for IoT Security	67	Review article	Q2	2.816	[73]
P11	Arshey & Angel, 2020	An Optimization-Based Deep Belief Network for the Detection of Phishing E-mails	1	Journal article	Q4	0.704	[78]
P12	Asharf et al, 2020	A Review of Intrusion Detection Systems Using Machine and Deep Learning in Internet of Things: Challenges, Solutions and Future Directions	8	Review article	Q2	2.412	[74]
P13	Basit et al, 2020	A Comprehensive Survey of AI-enabled Phishing Attacks Detection Techniques	8	Journal article	Q3	1.734	[8]
P14	Bello et al, 2020	Detecting Ransomware Attacks Using Intelligent Algorithms: Recent Development and Next Direction from Deep Learning and Big Data Perspectives	-	Journal article	Q1	4.594	[76]
P15	Berman et al, 2019	A Survey of Deep Learning Methods for Cyber Security	144	Review article	-	-	[58]
P16	Butez & Win, 2019	Detection of Phishing Websites using Generative Adversarial Network	-	Conference paper	-	-	[92]
P17	Castillo et al, 2020	Email Threat Detection Using Distinct Neural Network Approaches	-	Workshop proceeding	-	-	[79]
P18	Chatterjee & Namin, 2019	Detecting Phishing Websites through Deep Reinforcement Learning	21	Conference paper	-	-	[77]
P19	Chen, 2020	Deep Learning for Cybersecurity: A Review	-	Conference paper	-	-	[60]
P20	Chen et al, 2021	Cyber Security in Smart Cities: A Review of Deep Learning-based Applications and Case Studies	2	Journal article	Q1	5.268	[107]
P21	Digwal & Kavya, 2020	Detection of Phishing Website Based on Deep Learning	-	Journal article	-	-	[121]
P22	Dixit & Silakari, 2021	Deep Learning Algorithms for Cybersecurity Applications: A Technological and Status Review	1	Review article	Q1	7.707	[71]
P23	Elnagar & Thomas, 2018	A Cognitive Framework for Detecting Phishing Websites	2	Conference paper	-	-	[122]
P24	Fang et al, 2019	Phishing Email Detection Using Improved RCNN Model with Multilevel Vectors and Attention Mechanism	24	Journal article	Q1	3.745	[83]
P25	Feng & Yue, 2020	Visualizing and Interpreting RNN Models in URL-based Phishing Detection	3	Conference proceeding	-	-	[48]
P26	Feng et al, 2019	A Phishing Webpage Detection Method Based on Stacked Autoencoder and Correlation Coefficients	4	Journal article	-	-	[34]
P27	Feng et al, 2020	Web2Vec: Phishing Webpage Detection Method Based on Multidimensional Features Driven by Deep Learning	1	Journal article	Q1	3.745	[82]
P28	Geetha & Thilagam, 2020	A Review on the Effectiveness of Machine Learning and Deep Learning Algorithms for Cyber Security	2	Journal article	Q1	6.730	[61]
P29	Gupta et al, 2020	Machine Learning Models for Secure Data Analytics: A Taxonomy and Threat Model	54	Review article	Q2	2.816	[117]
P30	Halgas et al, 2019	Catching the Phish: Detecting Phishing Attacks Using Recurrent Neural Network (RNNs)	6	Conference paper	-	-	[80]
P31	Hatcher & Yu, 2018	A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends	230	Journal article	Q1	3.745	[57]

TABLE 11. ***(Continued.)*** **Selected studies for this slr.**

P32	Huang et al, 2019	Phishing URL Detection via CNN and Attention-Based Hierarchical RNN	9	Conference paper	-	-	[46]
P33	Kurnaz & Gwad, 2018	Deep Auto-Encoder Neural Network for Phishing Website Classification	1	Journal article	-	-	[103]
P34	Li et al, 2020	LSTM based Phishing Detection for Big Email Data	1	Journal article	-	-	[111]
P35	Liu & Lang, 2019	Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey	74	Review article	Q2	2.474	[63]
P36	Liu et al, 2019	Malicious Websites Detection via CNN based Screenshot Recognition	3	Conference paper	-	-	[102]
P37	Mahdavifar & Ghorbani, 2019	Application of Deep Learning to Cybersecurity: A Survey	71	Journal article	Q1	4.438	[59]
P38	Mahdavifar & Ghorbani, 2020	DeNNeS: Deep Embedded Neural Network Expert System for Detecting Cyber Attacks	3	Journal article	Q1	4.774	[42]
P39	Naway & Li, 2018	A Review on the Use of Deep Learning in Android Malware Detection	21	Journal article	-	-	[67]
P40	Nguyen et al, 2018	A Deep Learning Model with Hierarchical LSTMs and Supervised Attention for Anti-Phishing	10	Workshop proceeding	-	-	[86]
P41	Odeh et al, 2020	Efficient Detection of Phishing Websites Using Multilayer Perceptron	3	Journal article	-	-	[54]
P42	Phoka & Suthaphan, 2019	Image Based Phishing Detection Using Transfer Learning	1	Conference paper	-	-	[116]
P43	Phomkeona & Okamura, 2020	Zero-day Malicious Email Investigation and Detection Using Features with Deep-learning Approach	-	Journal article	-	-	[84]
P44	Pooja & Sridhar, 2020	Analysis of Phishing Website Detection Using CNN and Bidirectional LSTM	-	Conference paper	-	-	[123]
P45	Qamar et al, 2019	Mobile Malware Attacks: Review, Taxonomy & Future Directions	46	Journal article	Q1	0.528	[66]
P46	Rao et al, 2019	PhishDump: A Multi-model Ensemble Based Technique for the Detection of Phishing Sites in Mobile Devices	4	Journal article	Q2	2.725	[93]
P47	Rasymas & Dovydaitis, 2020	Detection of Phishing URLs by Using Deep Learning Approach and Multiple Features Combinations	-	Journal article	-	-	[94]
P48	Saha et al, 2020	Phishing Attacks Detection using Deep Learning Approach	3	Conference paper	-	-	[55]
P49	Sahingoz et al, 2018	Phishing Detection from URLs by Using Neural Networks	6	Journal article	-	-	[43]
P50	Sarker, 2021	Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspectives	5	Survey article	-	-	[70]
P51	Selvaganapathy et al, 2018	Deep Belief Network Based Detection and Categorization of Malicious URLs	24	Journal article	-	-	[75]
P52	Shirazi et al, 2020	Improved Phishing Detection Algorithms using Adversarial Autoencoder Synthesized Data	1	Conference paper	-	-	[95]
P53	Singh et al, 2020	Phishing Detection from URLs Using Deep Learning Approach	-	Conference paper	-	-	[124]
P54	Sohn, 2021	Deep Belief Network Based Intrusion Detection Techniques: A Survey	1	Journal article	Q1	5.452	[62]
P55	Somesha et al, 2020	Efficient Deep Learning Techniques for the Detection of Phishing Websites	6	Journal article	-	-	[44]
P56	Soon et al, 2020a	Comparison of Simple Feedforward Neural Network, Recurrent Neural Network and Ensemble Neural Networks in Phishing Detection	-	Journal article	-	-	[88]
P57	Soon et al, 2020b	Comparison of Ensemble Simple Feedforward Neural Network and Deep Learning Neural Network on Phishing Detection	1	Journal article	-	-	[89]
P58	Sountharranjan et al, 2020	Dynamic Recognition of Phishing URLs Using Deep Learning Techniques	2	Journal article	-	-	[96]
P59	Srinivasan et al 2020	DURLD: Malicious URL Detection Using Deep Learning-Based Character level Representations	1	Journal article	-	-	[101]
P60	Torroledo et al 2018	Hunting Malicious TLS Certificates with Deep Neural Networks	19	Conference proceeding	-	-	[49]
P61	Vigneshwaran et al 2020	Multiple Features Driven Phishing Detection based on Deep Learning	-	Journal article	-	-	[125]
P62	Vinayakumar et al, 2018	DeepAnti-PhishNet: Applying Deep Neural Networks for Phishing Email Detection	4	Workshop proceeding	-	-	[87]
P63	Vrbancic et al, 2018	Parameter Setting for Deep Neural Networks Using Swarm Intelligence on Phishing Websites Classification	7	Journal article	Q4	0.689	[45]

TABLE 11. *(Continued.)* **Selected studies for this slr.**

P64	Wang et al, 2019	Bidirectional LSTM Malicious Webpages Detection Algorithm Based on Convolutional Neural Network and Independent Recurrent Neural Network	11	Journal article	Q2	3.325	[47]
P65	Wang et al, 2020	Deep Learning-Based Efficient Model Development for Phishing Detection Using Random Forest and BLSTM Classifiers	3	Research article	Q2	2.462	[104]
P66	Wason, 2018	Deep Learning: Evolution and Expansion	58	Journal article	Q3	1.902	[109]
P67	Wei et al, 2020	Accurate and Fast URL Phishing Detector: A Convolutional Neural Network Approach	12	Journal article	Q2	3.111	[39]
P68	Weiping et al, 2019	PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks	7	Journal article	Q4	1.288	[97]
P69	Wu et al, 2020	Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey	2	Journal article	Q4	1.288	[114]
P70	Xiao et al, 2020	CNN-MHSA: A Convolutional Neural Network and Multi-head Self-attention Combined Approach for Phishing Detection Websites	11	Journal article	Q1	5.535	[40]
P71	Ya et al, 2019	NeuralAS: Deep Word-Based Spoofed URLs Detection Against Strong Similar Samples	-	Conference paper	-	-	[98]
P72	Yang, 2020	Research on Website Phishing Detection Based on LSTM RNN	3	Conference paper	-	-	[50]
P73	Yang et al, 2019a	Phishing Website Detection Based on Multidimensional Features Driven by Deep Learning	54	Journal article	Q1	3.745	[37]
P74	Yang et al, 2019b	Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network	40	Journal article	Q1	3.745	[51]
P75	Yazhmozhi et al, 2020	Anti-Phishing System Using LSTM and CNN	-	Conference paper	-	-	[99]
P76	Yerima & Alzaylaee, 2020	High Accuracy Phishing Detection Based on Convolutional Neural Networks	4	Conference paper	-	-	[41]
P77	Yi et al, 2018	Web Phishing Detection Using a Deep Learning Framework	38	Journal article	Q3	1.819	[108]
P78	Yu, 2020	Phishing Websites Detection Based on Hybrid Model of Deep Belief Network and Support Vector Machine	-	Conference paper	-	-	[100]
P79	Yuan et al, 2019	A Character-Level BiGRU-Attention for Phishing Classification	-	Conference paper	-	-	[52]
P80	Zhu, 2020	Online Meta-Learning Firewall to Prevent Phishing Attacks	-	Journal article	Q1	4.774	[85]
P81	Sahoo et al, 2019	Malicious URL Detection using Machine Learning: A Survey	171	Journal article	-	-	[9]

validation results is also known as overfitting problem [96]. Overfitting usually occurs when the model performs well on the training set, but poorly on the validation set, causing the training accuracy to be much higher than the validation accuracy. As a result, the smaller the gap between the blue and orange lines, the better the phishing detection model. In other words, the faster the training and validation graphs converge, the more efficient the DL algorithm. Most of the time, issues caused by overfitting can be prevented by using regularization techniques, such as batch normalization, early stopping or dropout [22], [23], [94], [101]. As can be seen from the graphs, CNN, LSTM and GRU models are less prone to overfitting problem since they implemented dropout function. In contrast, DNN and MLP algorithm might suffer from overfitting because none of the regularization techniques were used in the implementation of these DL models.

The results obtained from the experiments are summarized in TABLE 10. A set of metrics used to evaluate the performance of DL algorithms consists of FPR, FNR, Precision, Recall, F1-Score, AUC, and Accuracy. An effective and efficient phishing detection model is expected to have high Precision, Recall, F1-score, AUC, and Accuracy, while low in FPR and FNR measures. From these figures, it is observed that the accuracy of the ensemble DL model is higher than the individual model. For instance, among the LSTM models, ensemble LSTM architectures have slightly higher accuracy than a single LSTM model. The accuracy rate of LSTM-LSTM and BiLSTM-BiLSTM models are 93.76% and 92.67%, respectively, whereas that of the single LSTM architecture is 92.49%. Similarly, GRU-GRU has the highest accuracy level (95.75%) among the three GRU models, while a single GRU has the lowest accuracy measure (93.49%).

These results are consistent with what has been discussed in the previous section, in which ensemble DL models combine the strengths and resolve the weaknesses of individual models to achieve higher performance accuracy. It is also observed from the experiment that LSTM and GRU take longer training time as compared to any other models. In addition, among the LSTM architectures, the duration to train ensemble LSTM models is longer than the training time of a single LSTM model. These results are also in accordance with the previous literature in which the more complex the DL architecture is, the longer the training time. Therefore,

TABLE 12. Quality assessment score of the selected studies.

ID	QA1	QA2	QA3	QA4	QA5	Total
P1	1	1	1	0	0	3
P2	1	1	1	0	0	3
P3	1	0	1	1	1	4
P4	1	1	1	0	0	3
P5	1	1	0.5	0	0	2.5
P6	1	0	0	1	1	3
P7	1	1	0.5	0	0	2.5
P8	1	1	0.5	0	0	2.5
P9	1	1	0.5	0	0	2.5
P10	1	0	0	1	1	3
P11	1	1	0.5	0	0	2.5
P12	1	0	1	1	1	4
P13	0.5	0	0	1	1	2.5
P14	0	0	0.5	1	1	2.5
P15	1	0	1	1	1	4
P16	1	1	0.5	0	0	2.5
P17	1	1	1	0	0	3
P18	1	1	0.5	0	0	2.5
P19	1	0	1	1	1	4
P20	1	0	1	1	1	4
P21	1	1	0.5	0	0	2.5
P22	1	0	1	1	1	4
P23	1	0.5	1	0	0	2.5
P24	1	1	1	0	0	3
P25	1	1	1	0	0	3
P26	1	1	0.5	0	0	2.5
P27	1	1	1	0	0	3
P28	1	0	1	0.5	0	2.5
P29	1	0	1	0.5	0	2.5
P30	1	1	1	0	0	3
P31	0.5	0	0	1	1	2.5
P32	1	1	0.5	0	0	2.5
P33	1	1	0.5	0	0	2.5
P34	1	1	0.5	0	0	2.5
P35	1	0	1	0.5	0.5	3
P36	1	1	1	0	0	3
P37	1	0	1	1	1	4
P38	1	1	0.5	0	0	2.5
P39	0.5	0	1	1	1	3.5
P40	1	1	0.5	0	0	2.5
P41	1	1	0.5	0	0	2.5
P42	1	1	0.5	0	0	2.5
P43	1	1	0.5	0	0	2.5
P44	1	1	0.5	0	0	2.5
P45	0.5	0	0	1	1	2.5
P46	1	1	0.5	0	0	2.5
P47	1	1	0.5	0	0	2.5
P48	1	1	0.5	0	0	2.5
P49	1	1	0.5	0	0	2.5
P50	1	0	1	1	1	4
P51	1	1	1	0	0	3
P52	1	1	1	0	0	3
P53	1	1	0.5	0	0	2.5
P54	1	0	0	1	0.5	2.5
P55	1	1	1	0	0	3
P56	1	1	1	0	0	3
P57	1	1	0.5	0	0	2.5
P58	1	1	0.5	0.5	0	3
P59	1	1	0.5	0	0	2.5
P60	1	1	0.5	0	0	2.5
P61	1	1	0.5	0	0	2.5
P62	1	1	1	0	0	3
P63	1	1	0.5	0	0	2.5
P64	1	1	1	0	0	3
P65	1	1	0.5	0	0	2.5
P66	0.5	0	0	1	1	2.5
P67	1	1	0.5	0	0	2.5
P68	1	1	1	0	0	3
P69	1	0	1	0.5	0.5	3
P70	1	1	0.5	0	0	2.5
P71	1	1	0.5	0	0	2.5
P72	1	1	0.5	0	0	2.5
P73	1	1	0.5	0	0	2.5
P74	1	1	0.5	0	0	2.5
P75	1	1	0.5	0	0	2.5
P76	1	1	0.5	0	0	2.5
P77	1	1	0.5	0	0	2.5
P78	1	1	0.5	0	0	2.5
P79	1	1	0.5	0	0	2.5
P80	1	1	0.5	0	0	2.5
P81	0.5	0.5	0	1	1	3

besides having an effective DL model that can produce high detection accuracy, it is also crucial to reduce the training duration, since longer training time requires higher computational resources.

In short, the empirical results obtained from the experiment of various DL models have manifested the following issues that need to be addressed. First, there is no specific guideline for an optimal set of parameters that yield the best performance accuracy in detecting phishing attacks. Researchers need to find-tune these parameters manually by conducting very tedious and time-consuming series of experiments. Second, individual DL models might produce lower accuracy as compared to ensemble or hybrid models. As a result, it is recommended to combine different DL algorithms in a phishing detection model to have an effective and robust solution to fight against phishing attacks. Last but not least, training duration is another factor that needs to be taken into consideration. Even though ensemble and hybrid DL models have higher accuracy, they might also take a longer time to train. This becomes a problem because a longer duration requires higher computational cost, which reduces the model's efficiency.

This section has assessed the classification performance of different DL algorithms and discussed their related limitations by analyzing several DL models in a practical context. The empirical analysis was performed with recently published, publicly available and commonly-used dataset for benchmarking and evaluation in phishing detection. In addition, the performance of various DL models was also evaluated with a set of standard metrics frequently used for validation in the phishing detection domain. Altogether, the benchmarking dataset, the evaluation metrics, and the empirical results were discussed to highlight the overlooked issues along with the perspectives that encourage researchers to explore DL and navigate the future research directions of phishing detection in this regard.

VII. CONCLUSION AND FUTURE WORK

To sum up, DL has caught much attention among researchers across numerous application domains. DL can handle complex data and extract raw features automatically without prior

TABLE 13. Strengths and weaknesses of various DL techniques

No	Technique	Strengths	Weaknesses	Reference
1	CNN	(i) Well-fitted to multi-dimensional data such as image and speech signals (ii) Best suited for highly efficient and fast feature extraction from raw data (iii) Provided promising and robust results in many other applications (iv) Produce higher accuracies in resolving complex tasks (v) More scalable and require less training time	(i) Performance decreases when applied to non-spatial data (ii) Require high computational power (iii) Lack the ability to learn contextual information (iv) Require a big dataset of targeted images	[20], [21], [23], [58], [61], [67], [70], [74], [76], [82]
2	LSTM	(i) Can solve the vanishing gradient problem (ii) Can learn long-term dependencies (iii) Appropriate for solving problems in time-series sequence data (iv) Fast and effective relearning	(i) Takes a significantly long training time (ii) Only consider the forward information and ignore the backward information	[21], [58], [63], [70], [74], [107], [111], [114], [122]
3	DNN	(i) Accomplished success in different applications (ii) Express complex functions with fewer parameters (iii) Capable of facilitating tasks of feature extraction and representation learning	(i) The learning process could be time-consuming (ii) Require substantial labelled dataset for training (iii) Insufficient parameter selection techniques	[59], [67], [114]
4	MLP	(i) Able to learn non-linear models even in real-time or on-line learning using partial fit	(i) Model computationally expensive to solve a complex security model	[70]
5	RNN	(i) Suitable for time series/sequential data to maintain the continuity of information (ii) Provides promising results in sequential data, natural language and text processing (iii) Appropriate for text and pattern recognition (iv) Successful in next-word-in-a-sentence prediction, speech recognition, etc.	(i) Has an issue of vanishing or exploding gradients (ii) Unable to capture long-term dependencies (iii) More challenging to train than FFNN (iv) Require many resources and time to get trained	[20], [21], [58], [59], [61], [67], [70], [74], [107], [114]
6	GRU	(i) Lightweight version of LSTM (ii) Simpler than LSTM in computation and implementation (iii) Still effective in capturing both short-term and long-term dependencies	(i) Require long training time	[23], [48], [114]
7	DBM	(i) Can handle ambiguous inputs (ii) Solve issues arising from the complexity of BM (iii) Eliminate the connections among neurons in the same layer	(i) Performance sensitive to the parameter selection	[23], [75], [76]
8	RBM	(i) Feedback function facilitates extraction of important attributes (ii) Can be used as features extractor to train other models on top of it (iii) Play an important role in dimensionality reduction, classification, etc.	(i) Need high computational resources while implementing it on low-powered devices (ii) Single RBM lacks the capability of feature representation (iii) Hard to train well	[67], [70], [74]
9	DBN	(i) Applied for dimensionality reduction/ stand-alone classifier (ii) Efficient and fast, successfully used for pre-training tasks (iii) Suitable for vital feature extraction with training on unlabeled data (iv) Solve the problem of a single RBM, DNN	(i) Require a training phase, but with unlabeled datasets (ii) Require high computational cost (iii) Redundant information, easy to trap into local maximal	[20], [23], [61], [67], [70], [74], [107], [114]
10	AE	(i) Reduce the data into lower dimensions without losing its semantics (ii) Improve the system accuracy and reduce its computational complexity (iii) Learn and classify output automatically without the need for a labelled dataset (iv) Typically used for dimensionality reduction (v) Achieve better results on small datasets (vi) Can continuously extract useful features and filter the useless information (vii) Capable of learning potential representation of unknow attacks	(i) Require high computational power (ii) May not produce desired results if the training dataset is not representative of the testing dataset (iii) May suffer from extensive unlabeled data without enough prior knowledge (iv) Contain only one hidden layer (v) The ability to represent complex function is limited in the case of finite samples and computational units (vi) Generalization ability is constrained for complex problems	[20], [21], [23], [34], [61], [70], [114], [117]
11	DAE	(i) Applied to feature extraction or dimensionality reduction	(i) Need pre-training phase (ii) Does not have the capacity to figure out what data is pertinent	[67], [74]
12	GAN	(i) Not rely on any assumptions about the distribution (ii) Useful for detecting zero-day attacks (iii) Able to generate real and balanced dataset (iv) Discover inherent pattern of data to generate new samples	(i) Training is difficult (ii) Produce unstable results (iii) New in conception	[58], [70], [74], [107], [114]
13	DRL	(i) Can be used to solve complex problems that cannot be solved by conventional techniques (ii) Self-adaptive to the changes in the features	(i) Not yet been fully explored	[57]

TABLE 14. Convolutional neural network (CNN) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P1	[90]	Website phishing detection	✓	Sigmoid	Matlab	PhishTank Common Crawl	1 million URLs 10,000 images
2	P2	[91]	Website phishing detection		✓	Matlab	PhishTank Common Crawl	1 million URLs 10,000 images
3	P4	[120]	Website phishing detection	✓	✓	Python	NA	2,000 screenshots & URLs
4	P7	[22]	Website phishing detection	✓	NB, LR, RF, XGBoost, DNN	Python	PhishTank Common Crawl Yandex Alexa	D1: 318,642 D2: 73,575 D3: 83,857 D4: 82,888
5	P8	[38]	Website phishing detection	✓	Sigmoid	Keras Tensorflow	PhishTank Millersmiles Yahoo Starting point	2,456 instances
6	P9	[81]	Email phishing detection	✓	Sigmoid	Python Keras Tensorflow GoogleColab	PhishingCorpus SpamAssassin	7,315 6,047
7	P17	[79]	Email phishing detection	✓	ReLU	Python Keras	Enron APWG Non-public	84,111 30,776 4,048
8	P21	[121]	Website phishing detection	✓	XGBoost	JavaScript	NA	NA
9	P23	[122]	Website phishing detection	✓		NA	NA	NA
10	P27	[82]	Website phishing detection	✓	Sigmoid	Python	PhishTank Alexa/Amazon	21,303 24,800
11	P32	[46]	Website phishing detection	✓	Softmax	NA	PhishTank Openphish Alexa	4,820,940 URLs
12	P36	[102]	Website phishing detection	✓	Sigmoid	Python Keras Tensorflow	DMOZ Own dataset	3,816
13	P42	[116]	Website phishing detection	NA	NA	Tensorflow	ILSVRC-2012-CLS	1,2 million images
14	P44	[123]	Website phishing detection	✓	XGBoost	NA	NA	40,000
15	P47	[94]	Website phishing detection	✓	Sigmoid	NA	PhishTank	2,585,146
16	P53	[124]	Website phishing detection	✓	Softmax	NA	Ebbu2017	73,575
17	P55	[44]	Website phishing detection		✓	Python Tensorflow	PhishTank Alexa	2,119 1,407
18	P59	[101]	Website phishing detection	✓	Sigmoid	Tensorflow Keras	Alexa, DMOZ, etc., Sophos	611,894 124,574
19	P61	[125]	Website phishing detection	✓	Softmax/ XGBoost	NA	PhishTank DMOZ	1,021,758 989,021
20	P62	[87]	Email phishing detection	✓	Sigmoid	Tensorflow Keras	IWSPA-AP 2018	NA
21	P64	[47]	Website phishing detection	✓		Python	PhishTank Common Crawl	13,652 10,000
22	P67	[39]	Website phishing detection	✓	ReLU	Python	PhishTank Common Crawl	10,604 10,604
23	P68	[97]	Website phishing detection	✓	Sigmoid	Python Tensorflow	PhishTank Alexa	245,385 245,023
24	P70	[40]	Website phishing detection	✓	Sigmoid	NA	PhishTank 5000 Best Websites	43,984 45,000
25	P73	[37]	Website phishing detection	✓	XGBoost	NA	PhishTank DMOZ	1,021,758 989,021
26	P75	[99]	Website phishing detection	✓	Sigmoid	NA	PhishTank Virus Total Yandex	97,400 97,400
27	P76	[41]	Website phishing detection	✓	Sigmoid	Python Keras Tensorflow	UCI	11,055

knowledge. DL has become one of the top interested topics in the cybersecurity with the advent of new technologies and the rapid growth of data in the big data era, especially in the phishing detection field. As a result, this study provided a comprehensive review of DL for phishing detection through an in-depth SLR approach. The paper also offered a

TABLE 15. Long short-term memory (LSTM) for phishing detection

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P1	[90]	Website phishing detection	✓	Softmax	Matlab	PhishTank Common Crawl	1 million URLs 10,000 images
2	P2	[91]	Website phishing detection		✓	Matlab	PhishTank Common Crawl	1 million URLs 10,000 images
3	P17	[79]	Email phishing detection	✓	Sigmoid	Python Keras	Enron APWG Non-public	84,111 30,776 4,048
4	P21	[121]	Website phishing detection	✓	XGBoost	JavaScript	NA	NA
5	P25	[48]	Website phishing detection	✓	Sigmoid	NA	PhishTank Common Crawl	1.5 million URLs
6	P30	[80]	Email phishing detection	✓	Tanh		SpamAssassin Enron Narazio	6,951 10,000 14,534
7	P34	[111]	Email phishing detection	✓	Sigmoid	Python Keras Tensorflow	Own dataset	29,942,735
8	P46	[93]	Website phishing detection	✓	SVM	Python	PhishTank Openphish Alexa	153,788 7,212 170,552
9	P47	[94]	Website phishing detection	✓	Sigmoid	NA	PhishTank	2,585,146
10	P55	[44]	Website phishing detection		✓	Python Tensorflow	PhishTank Alexa	2,119 1,407
11	P59	[101]	Website phishing detection	✓	Sigmoid	Tensorflow Keras	Alexa, DMOZ, etc., Sophos	611,894 124,574
12	P60	[49]	Website phishing detection	✓	Logit	NA	Vaderetro Alexa	2,000 1,000,000
13	P61	[125]	Website phishing detection	✓	Softmax/ XGBoost	NA	PhishTank DMOZ	1,021,758 989,021
14	P62	[87]	Email phishing detection	✓	Sigmoid	Tensorflow Keras	IWSPA-AP 2018	NA
15	P72	[50]	Website phishing detection	✓	Sigmoid	Python Keras	PhishTank Yahoo Directory	2,000 2,000
16	P73	[37]	Website phishing detection	✓	XGBoost	NA	PhishTank DMOZ	1,021,758 989,021
17	P75	[99]	Website phishing detection	✓	Sigmoid	NA	PhisTank Virus Total Yandex	97,400 97,400
18	P9	[81]	Email phishing detection	✓	Softmax	Python Tensorflow	Phishing Corpus SpamAssassin	2,585,146

TABLE 16. Deep belief network (DBN) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P11	[78]	Email phishing detection		✓	Java	Enron UCI	NA
2	P51	[75]	Website phishing detection	✓	DNN	RStudio	UCI DMOZ	17,700 10,000
3	P77	[108]	Website phishing detection	✓		CUDA C++	ISP	1,982,005 URLs
4	P78	[100]	Website phishing detection	✓	SVM	Python	PhishTank DMOZ	864,753 224,259

TABLE 17. Deep boltzmann machine (DBM) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P58	[96]	Website phishing detection	✓	DNN	RStudio	PhishTank DMOZ PageRank WHOIS	17,000 20,000 480 480

significant insight into the current issues and challenges that DL faces in detecting phishing attacks by analyzing the trends and patterns of 81 selected articles from various sources. This research has drawn a taxonomy for phishing detection and DL to classify them into several classes based on a thorough analysis of the relevant studies. Phishing detection was

TABLE 18. Bidirectional long short-term memory (BILSTM) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P23	[122]	Website phishing detection		✓	NA	NA	NA
2	P24	[83]	Email phishing detection		✓	Tensorflow Keras	IWSPA-AP 2018 Enron SpamAssasin	8,780
3	P25	[48]	Website phishing detection	✓	Sigmoid	NA	PhishTank Common Crawl	1.5 million URLs
4	P27	[82]	Website phishing detection	✓	Sigmoid	Python	PhishTank Alexa/Amazon	21,303 24,800
5	P32	[46]	Website phishing detection	✓	Softmax	NA	PhishTank Openphish Alexa	4,820,940 URLs
6	P40	[86]	Email phishing detection	✓	Tanh	Python	IWSPA-AP 2018	5,721 4,585
7	P44	[123]	Website phishing detection	✓	XGBoost	NA	NA	40,000
8	P64	[47]	Website phishing detection	✓		Python	PhishTank Crawler	13,652 10,000
9	P65	[104]	Website phishing detection		✓	NA	UCI	2,456
10	P68	[97]	Website phishing detection	✓	Sigmoid	Python Tensorflow	PhishTank Alexa	245,385 245,023
11	P71	[98]	Website phishing detection	✓	Sigmoid	NA	PhishTank DMOZ	120,166 300,000

TABLE 19. Deep neural network (DNN) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P38	[42]	Website phishing detection	✓	Sigmoid	Python Tensorflow	UCI	11,055
2	P49	[43]	Website phishing detection	✓	Sigmoid	Tensorflow	PhishTank Yandex	73,575 URLs
3	P51	[75]	Website phishing detection	DBN	✓	RStudio	UCI DMO	17,700 10,000
4	P55	[44]	Website phishing detection		✓	Python Tensorflow	PhishTank Alexa	2,119 1,407
5	P56	[88]	Email phishing detection	✓	Sigmoid	NA	CSDMC2010 SPAM	4,327
6	P58	[96]	Website phishing detection	DBM, SAE	✓	RStudio	PhishTank DMOZ PageRank WHOIS	17,000 20,000 480 480
7	P63	[45]	Website phishing detection	✓	Softmax	Python	PhishTank Yahoo Own dataset	11,055 1,353 58,645 88,657

TABLE 20. Stacked autoencoder (SAE/DAE) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P26	[34]	Website phishing detection	✓		Matlab	PhishTank Alexa	11,321 8,848
2	P33	[103]	Website phishing detection	✓	Softmax	NA	UCI	NA
3	P58	[96]	Website phishing detection	✓	DNN	RStudio	PhishTank DMOZ PageRank WHOIS	17,000 20,000 480 480

classified according to different media and methods, while DL was classified by the application areas, techniques and datasets. Moreover, this paper also differentiated DL from traditional machine learning, and analyzed the strengths and weaknesses of several DL algorithms used in the previous studies. Finally, an empirical analysis was conducted to highlight the open issues discussed in the literature and identify possible research gaps for future directions. The

TABLE 21. Deep reinforcement learning (DRL) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P18	[77]	Website phishing detection		✓	Tensorflow	Ebbu2017 Yandex PhishTank	73,575

TABLE 22. Multilayer perceptron (MLP) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P5	[53]	Website phishing detection		✓	NA	UCI Kaggle	11,055 URLs 2,456 URLs
2	P11	[78]	Email phishing detection		✓	Java	Enron UCI	NA
3	P32	[46]	Website phishing detection	✓	Softmax	NA	PhishTank Openphish Alexa	4,820,940 URLs
4	P41	[54]	Website phishing detection	✓		Weka Python	PhishTank Millersmiles Google search	2,456
5	P43	[84]	Email phishing detection		✓	Python Tensorflow CUDA	Spam Enron Kyushu Own dataset	4,567,714 517,401 281 4,251
6	P48	[55]	Website phishing detection	✓		Python	Kaggle	10,000
7	P62	[87]	Email phishing detection	✓	Sigmoid	Tensorflow Keras	IWSPA-AP 2018	NA

TABLE 23. Recurrent neural network (RNN) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P56	[88]	Email phishing detection	✓	Sigmoid	NA	CSDMC2010 SPAM	4,327
2	P62	[87]	Email phishing detection	✓	Sigmoid	Tensorflow Keras	IWSPA-AP 2018	NA
3	P64	[47]	Website phishing detection	✓		Python	PhishTank Crawler	13,652 10,000

TABLE 24. Gated recurrent unit (GRU) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P25	[48]	Website phishing detection	✓	Sigmoid	NA	PhishTank Common Crawl	1.5 million URLs
2	P74	[51]	Website phishing detection	✓	Softmax	Python Keras	Own dataset	340,000 65,000

TABLE 25. Bidirectional gated recurrent unit (BIGRU) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P25	[48]	Website phishing detection	✓	Sigmoid	NA	PhishTank Common Crawl	1.5 million URLs
2	P79	[52]	Website phishing detection	✓	Sigmoid	NA	PhishTank Common Crawl	759,361

TABLE 26. Adversarial autoencoder (AAE) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P52	[95]	Website phishing detection	✓	DT, GB, KNN, RF, SVM	Python	PhishTank OpenPhish Alexa UCI Mendeley	1,000 3,013 2,000 3,850 10,000

results obtained from the empirical experiments indicated that the most common issues among DL techniques are manual parameter tunning, long training time and deficient performance accuracy. These findings imply that further efforts need to be taken to improve the state-of-the-art DL algorithms in terms of fine-tunning, training duration and detection accuracy, to ensure a robust and effective system for detecting phishing attacks in cyberspace. These outcomes also suggested that in addition to optimization techniques and ensemble methods, integrating DL with big data or cloud-based

TABLE 27. Generative adversarial network (GAN) for phishing detection.

No	ID	Reference	Application area	Feature extraction	Classification	Platform	Dataset	Dataset size
1	P16	[92]	Website phishing detection		✓	Keras	Amazon PhishTank Basic workstation	24,084 websites 11,267 websites

technologies in a hybrid approach are new research directions for phishing detection. Based on the above analysis, we believe that this study will serve as a valuable reference for researchers and developers in the field of cybersecurity.

As for future work, we will conduct extensive experiments by using different sets of parameters to obtain the highest possible detection accuracy. In addition, we also plan to include other DL techniques not yet been fully explored in phishing detection, such as GAN or DRL. Besides homogeneous architectures, we will implement heterogeneous ensemble DL models by integrating DL algorithms from different genres, for example, CNN-LSTM, DNN-AE, MLP-GRU, etc., to examine the effectiveness and efficiency of ensemble methods over individual techniques. Last but not least, instead of using a balanced dataset, we will use an imbalanced one in the experiment setup, owing to the fact that in real-life scenarios, phishing is an imbalanced classification problem, where the number of legitimate instances is much higher than the phishing ones.

APPENDIX

See Tables 11–27.

ACKNOWLEDGMENT

The authors sincerely thank the Ministry of Higher Education under the Fundamental Research Grant Scheme under Grant FRGS/1/2018/ICT04/UTM/01/1, Universiti Teknologi Malaysia (UTM) under Research University Grant Vot-20H04, Malaysia Research University Network (MRUN) Vot 4L876, for the completion of the research. They also grateful for the support of student Michal Dobrovolny in consultations regarding application aspects.

REFERENCES

[1] R. Zaimi, M. Hafidi, and M. Lamia, "Survey paper: Taxonomy of website anti-phishing solutions," in *Proc. 7th Int. Conf. Social Netw. Anal., Manage. Secur. (SNAMS)*, Dec. 2020, pp. 1–8, doi: 10.1109/SNAMS52053.2020.9336559.

[2] A. Odeh, I. Keshta, and E. Abdelfattah, "Machine LearningTechniquesfor detection of website phishing: A review for promises and challenges," in *Proc. IEEE 11th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2021, pp. 0813–0818, doi: 10.1109/CCWC51732.2021.9375997.

[3] L. Tang and Q. H. Mahmoud, "A survey of machine learning-based solutions for phishing website detection," *Mach. Learn. Knowl. Extraction*, vol. 3, no. 3, pp. 672–694, Aug. 2021, doi: 10.3390/make3030034.

[4] E. S. Aung, C. T. Zan, and H. Yamana. *A Survey of URL-Based Phishing Detection*. Accessed: Mar. 22, 2022. [Online]. Available: http://quadrodeofertas.com.br/www1

[5] N. Valiyaveedu, S. Jamal, R. Reju, V. Murali, and K. M. Nithin, "Survey and analysis on AI based phishing detection techniques," in *Proc. Int. Conf. Commun., Control Inf. Sci. (ICCISc)*, Jun. 2021, pp. 1–6, doi: 10.1109/ICCISc52257.2021.9484929.

[6] E. Benavides, W. Fuertes, S. Sanchez, and M. Sanchez, "Classification of phishing attack solutions by employing deep learning techniques: A systematic literature review," in *Developments and Advances in Defense and Security*. Singapore: Springer, 2020, pp. 51–64, doi: 10.1007/978-981-13-9155-2_5.

[7] A. El Aassal, S. Baki, A. Das, and R. M. Verma, "An in-depth benchmarking and evaluation of phishing detection research for security needs," *IEEE Access*, vol. 8, pp. 22170–22192, 2020, doi: 10.1109/ACCESS.2020.2969780.

[8] A. Basit, M. Zafar, X. Liu, A. R. 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, Jan. 2021, doi: 10.1007/s11235-020-00733-2.

[9] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL detection using machine learning: A survey," 2017, *arXiv:1701.07179*.

[10] A. Das, S. Baki, A. El Aassal, R. Verma, and A. Dunbar, "SoK: A comprehensive reexamination of phishing research from the security perspective," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 671–708, 1st Quart., 2020, doi: 10.1109/COMST.2019.2957750.

[11] R. von Solms and J. van Niekerk, "From information security to cyber security," *Comput. Secur.*, vol. 38, pp. 97–102, Oct. 2013, doi: 10.1016/j.cose.2013.04.004.

[12] G. Diksha and J. A. Kumar, "Mobile phishing attacks and defence mechanisms: State of art and open research challenges," *Comput. Secur.*, vol. 73, pp. 519–544, Mar. 2018, doi: 10.1016/j.cose.2017.12.006.

[13] *APWG | Phishing Activity Trends Reports*. Accessed: Apr. 8, 2021. [Online]. Available: https://apwg.org/trendsreports/

[14] Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha, and M. Guizani, "Systematization of knowledge (SoK): A systematic review of software-based web phishing detection," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2797–2819, 4th Quart., 2017, doi: 10.1109/COMST.2017.2752087.

[15] K. L. Chiew, K. S. C. Yong, and C. L. Tan, "A survey of phishing attacks: Their types, vectors and technical approaches," *Expert Syst. Appl.*, vol. 106, pp. 1–20, Sep. 2018, doi: 10.1016/j.eswa.2018.03.050.

[16] Y. Ding, N. Luktarhan, K. Li, and W. Slamu, "A keyword-based combination approach for detecting phishing webpages," *Comput. Secur.*, vol. 84, pp. 256–275, Jul. 2019, doi: 10.1016/j.cose.2019.03.018.

[17] B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, "Fighting against phishing attacks: State of the art and future challenges," *Neural Comput. Appl.*, vol. 28, no. 12, pp. 3629–3654, Dec. 2017, doi: 10.1007/s00521-016-2275-y.

[18] A. Abbasi, D. Dobolyi, A. Vance, and F. M. Zahedi, "The phishing funnel model: A design artifact to predict user susceptibility to phishing websites," *Inf. Syst. Res.*, vol. 32, no. 2, pp. 410–436, Jun. 2021, doi: 10.1287/isre.2020.0973.

[19] R. Talwar and A. Koury, "Artificial intelligence–The next frontier in IT security?" *Netw. Secur.*, vol. 2017, no. 4, pp. 14–17, Apr. 2017, doi: 10.1016/S1353-4858(17)30039-9.

[20] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the effectiveness of machine and deep learning for cyber security," in *Proc. 10th Int. Conf. Cyber Conflict (CyCon)*, May 2018, pp. 371–390, doi: 10.23919/CYCON.2018.8405026.

[21] R. Ahmad and I. Alsmadi, "Machine learning approaches to IoT security: A systematic literature review," *Internet Things*, vol. 14, Jun. 2021, Art. no. 100365, doi: 10.1016/j.iot.2021.100365.

[22] A. Aljofey, Q. Jiang, Q. Qu, M. Huang, and J.-P. Niyigena, "An effective phishing detection model based on character level convolutional neural network from URL," *Electronics*, vol. 9, no. 9, p. 1514, Sep. 2020, doi: 10.3390/electronics9091514.

[23] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowl.-Based Syst.*, vol. 189, Feb. 2020, Art. no. 105124, doi: 10.1016/j.knosys.2019.105124.

[24] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering–A systematic literature review," *Inf. Softw. Technol.*, vol. 51, no. 1, pp. 7–15, Jan. 2009, doi: 10.1016/j.infsof.2008.09.009.
[25] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement," *PLoS Med.*, vol. 6, no. 7, Jul. 2009, Art. no. e1000097, doi: 10.1371/journal.pmed.1000097.
[26] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proc. 18th Int. Conf. Eval. Assessment Softw. Eng.*, London, U.K., 2014, pp. 1–10, doi: 10.1145/2601248.2601268.
[27] X. Zhou, Y. Jin, H. Zhang, S. Li, and X. Huang, "A map of threats to validity of systematic literature reviews in software engineering," in *Proc. 23rd Asia–Pacific Softw. Eng. Conf. (APSEC)*, 2016, pp. 153–160, doi: 10.1109/APSEC.2016.031.
[28] A. Aleroud and L. Zhou, "Phishing environments, techniques, and countermeasures: A survey," *Comput. Secur.*, vol. 68, pp. 160–196, Jul. 2017, doi: 10.1016/j.cose.2017.04.006.
[29] I. Qabajeh, F. Thabtah, and F. Chiclana, "A recent review of conventional vs. automated cybersecurity anti-phishing techniques," *Comput. Sci. Rev.*, vol. 29, pp. 44–55, Aug. 2018, doi: 10.1016/j.cosrev.2018.05.003.
[30] R. S. Rao and A. R. Pais, "Detection of phishing websites using an efficient feature-based machine learning framework," *Neural Comput. Appl.*, vol. 31, no. 8, pp. 3851–3873, Aug. 2019, doi: 10.1007/s00521-017-3305-0.
[31] A. A. Zuraiq and M. Alkasassbeh, "Review: Phishing detection approaches," in *Proc. 2nd Int. Conf. New Trends Comput. Sci. (ICTCS)*, Oct. 2019, pp. 1–6, doi: 10.1109/ICTCS.2019.8923069.
[32] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, "PhishNet: Predictive blacklisting to detect phishing attacks," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–5, doi: 10.1109/INFCOM.2010.5462216.
[33] S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang. *An Empirical Analysis of Phishing Blacklists*. Accessed: Mar. 22, 2022. [Online]. Available: https://kilthub.cmu.edu/articles/An_Empirical_Analysis_of_Phishing_Blacklists/6469805/files/11898359.pdf
[34] J. Feng, L. Zou, and T. Nan, "A phishing webpage detection method based on stacked autoencoder and correlation coefficients," *J. Comput. Inf. Technol.*, vol. 27, no. 2, pp. 41–54, 2019, doi: 10.20532/cit.2019.1004702.
[35] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: A literature survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 2091–2121, 4th Quart, 2013, doi: 10.1109/SURV.2013.032213.00009.
[36] S. Patil and S. Dhage, "A methodical overview on phishing detection along with an organized way to construct an anti-phishing framework," in *Proc. 5th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, Mar. 2019, pp. 588–593, doi: 10.1109/ICACCS.2019.8728356.
[37] 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, doi: 10.1109/ACCESS.2019.2892066.
[38] N. Al-Milli and B. H. Hammo, "A convolutional neural network model to detect illegitimate URLs," in *Proc. 11th Int. Conf. Inf. Commun. Syst. (ICICS)*, Apr. 2020, pp. 220–225, doi: 10.1109/ICICS49469.2020.239536.
[39] W. Wei, Q. Ke, J. Nowak, M. Korytkowski, R. Scherer, and M. Woźniak, "Accurate and fast URL phishing detector: A convolutional neural network approach," *Comput. Netw.*, vol. 178, Sep. 2020, Art. no. 107275, doi: 10.1016/j.comnet.2020.107275.
[40] X. Xiao, D. Zhang, G. Hu, Y. Jiang, and S. Xia, "CNN–MHSA: A convolutional neural network and multi-head self-attention combined approach for detecting phishing websites," *Neural Netw.*, vol. 125, pp. 303–312, May 2020, doi: 10.1016/j.neunet.2020.02.013.
[41] S. Y. Yerima and M. K. Alzaylaee, "High accuracy phishing detection based on convolutional neural networks," in *Proc. 3rd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, Mar. 2020, pp. 1–6, doi: 10.1109/ICCAIS48893.2020.9096869.
[42] S. Mahdavifar and A. A. Ghorbani, "DeNNeS: Deep embedded neural network expert system for detecting cyber attacks," *Neural Comput. Appl.*, vol. 32, no. 18, pp. 14753–14780, Sep. 2020, doi: 10.1007/s00521-020-04830-w.
[43] O. K. Sahingoz, S. I. Baykal, and D. Bulut, "Phishing detection from urls by using neural networks," in *Computer Science & Information Technology (CS&IT)*. India: AIRCC Publishing Corporation, Dec. 2018, pp. 41–54, doi: 10.5121/csit.2018.81705.
[44] M. Somesha, A. R. Pais, R. S. Rao, and V. S. Rathour, "Efficient deep learning techniques for the detection of phishing websites," *Sādhanā*, vol. 45, no. 1, p. 165, Jun. 2020, doi: 10.1007/s12046-020-01392-4.
[45] G. Vrbančič, I. Fister, and V. Podgorelec, "Parameter setting for deep neural networks using swarm intelligence on phishing websites classification," *Int. J. Artif. Intell. Tools*, vol. 28, no. 6, Sep. 2019, Art. no. 1960008, doi: 10.1142/S021821301960008X.
[46] Y. Huang, Q. Yang, J. Qin, and W. Wen, "Phishing URL detection via CNN and attention-based hierarchical RNN," in *Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Communications/13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2019, pp. 112–119, doi: 10.1109/TrustCom/BigDataSE.2019.00024.
[47] H. Wang, L. Yu, S. Tian, Y. Peng, and X. Pei, "Bidirectional LSTM malicious webpages detection algorithm based on convolutional neural network and independent recurrent neural network," *Appl Intell*, vol. 49, no. 8, pp. 3016–3026, Aug. 2019, doi: 10.1007/s10489-019-01433-4.
[48] T. Feng and C. Yue, "Visualizing and interpreting RNN models in URL-based phishing detection," in *Proc. 25th ACM Symp. Access Control Models Technol.*, New York, NY, USA, Jun. 2020, pp. 13–24, doi: 10.1145/3381991.3395602.
[49] I. Torroledo, L. D. Camacho, and A. C. Bahnsen, "Hunting malicious TLS certificates with deep neural networks," in *Proc. 11th ACM Workshop Artif. Intell. Secur.*, New York, NY, USA, Jan. 2018, pp. 64–73, doi: 10.1145/3270101.3270105.
[50] Y. Su, "Research on website phishing detection based on LSTM RNN," in *Proc. IEEE 4th Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, vol. 1, Jun. 2020, pp. 284–288, doi: 10.1109/ITNEC48623.2020.9084799.
[51] W. Yang, W. Zuo, and B. Cui, "Detecting malicious URLs via a keyword-based convolutional gated-recurrent-unit neural network," *IEEE Access*, vol. 7, pp. 29891–29900, 2019, doi: 10.1109/ACCESS.2019.2895751.
[52] L. Yuan, Z. Zeng, Y. Lu, X. Ou, and T. Feng, "A character-level BiGRU-attention for phishing classification," in *Information and Communications Security*. Cham, Switzerland: Springer, 2020, pp. 746–762, doi: 10.1007/978-3-030-41579-2_43.
[53] S. Al-Ahmadi. (2020). *PDMLP: Phishing Detection Using Multilayer Perceptron*. Social Science Research Network, Rochester, NY, USA, SSRN Scholarly Paper ID. Accessed: May 12, 2021. [Online]. Available: https://papers.ssrn.com/abstract=3624621
[54] A. Odeh, I. Keshta, and E. Abdelfattah. (2020). *Efficient Detection of Phishing Websites Using Multilayer Perceptron*. International Association of Online Engineering. Accessed: Mar. 10, 2021. pp. 22–31. [Online]. Available: https://www.learntechlib.org/p/217754/
[55] I. Saha, D. Sarma, R. J. Chakma, M. N. Alam, A. Sultana, and S. Hossain, "Phishing attacks detection using deep learning approach," in *Proc. 3rd Int. Conf. Smart Syst. Inventive Technol. (ICSSIT)*, Aug. 2020, pp. 1180–1185, doi: 10.1109/ICSSIT48917.2020.9214132.
[56] B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, "Defending against phishing attacks: Taxonomy of methods, current issues and future directions," *Telecommun. Syst.*, vol. 67, no. 2, pp. 247–267, Feb. 2018, doi: 10.1007/s11235-017-0334-z.
[57] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018, doi: 10.1109/ACCESS.2018.2830661.
[58] D. Berman, A. Buczak, J. Chavis, and C. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, p. 122, Apr. 2019, doi: 10.3390/info10040122.
[59] S. Mahdavifar and A. A. Ghorbani, "Application of deep learning to cybersecurity: A survey," *Neurocomputing*, vol. 347, pp. 149–176, Jun. 2019, doi: 10.1016/j.neucom.2019.02.056.
[60] Z. Chen, "Deep learning for cybersecurity: A review," in *Proc. Int. Conf. Comput. Data Sci. (CDS)*, Aug. 2020, pp. 7–18, doi: 10.1109/CDS49703.2020.00009.
[61] R. Geetha and T. Thilagam, "A review on the effectiveness of machine learning and deep learning algorithms for cyber security," *Arch. Comput. Methods Eng.*, vol. 28, no. 4, pp. 2861–2879, Jun. 2021, doi: 10.1007/s11831-020-09478-2.
[62] I. Sohn, "Deep belief network based intrusion detection techniques: A survey," *Expert Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114170, doi: 10.1016/j.eswa.2020.114170.
[63] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *Appl. Sci.*, vol. 9, no. 20, p. 4396, Oct. 2019, doi: 10.3390/app9204396.

[64] G. Bovenzi, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescape, "A hierarchical hybrid intrusion detection approach in IoT scenarios," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2020, pp. 1–7, doi: 10.1109/GLOBECOM42002.2020.9348167.

[65] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," 2018, *arXiv:1802.09089*.

[66] A. Qamar, A. Karim, and V. Chang, "Mobile malware attacks: Review, taxonomy & future directions," *Future Gener. Comput. Syst.*, vol. 97, pp. 887–909, Aug. 2019, doi: 10.1016/j.future.2019.03.007.

[67] A. Naway and Y. Li, "A review on the use of deep learning in Android malware detection," 2018, *arXiv:1812.10360*.

[68] A. Barushka and P. Hajek, "Spam filtering using integrated distribution-based balancing approach and regularized deep neural networks," *Appl. Intell.*, vol. 48, no. 10, pp. 3538–3556, Oct. 2018, doi: 10.1007/s10489-018-1161-y.

[69] A. Barushka and P. Hajek, "Spam detection on social networks using cost-sensitive feature selection and ensemble-based regularized deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 9, pp. 4239–4257, May 2020, doi: 10.1007/s00521-019-04331-5.

[70] I. H. Sarker, "Deep cybersecurity: A comprehensive overview from neural network and deep learning perspective," *Social Netw. Comput. Sci.*, vol. 2, no. 3, p. 154, Mar. 2021, doi: 10.1007/s42979-021-00535-6.

[71] P. Dixit and S. Silakari, "Deep learning algorithms for cybersecurity applications: A technological and status review," *Comput. Sci. Rev.*, vol. 39, Feb. 2021, Art. no. 100317, doi: 10.1016/j.cosrev.2020.100317.

[72] T. Gangavarapu, C. D. Jaidhar, and B. Chanduka, "Applicability of machine learning in spam and phishing email filtering: Review and approaches," *Artif. Intell. Rev.*, vol. 53, no. 7, pp. 5019–5081, Oct. 2020, doi: 10.1007/s10462-020-09814-9.

[73] M. A. Amanullah, R. A. A. Habeeb, F. H. Nasaruddin, A. Gani, E. Ahmed, A. S. M. Nainar, N. M. Akim, and M. Imran, "Deep learning and big data technologies for IoT security," *Comput. Commun.*, vol. 151, pp. 495–517, Feb. 2020, doi: 10.1016/j.comcom.2020.01.016.

[74] J. Asharf, N. Moustafa, H. Khurshid, E. Debie, W. Haider, and A. Wahab, "A review of intrusion detection systems using machine and deep learning in Internet of Things: Challenges, solutions and future directions," *Electronics*, vol. 9, no. 7, p. 1177, Jul. 2020, doi: 10.3390/electronics9071177.

[75] S. G. Selvaganapathy, M. Nivaashini, and H. P. Natarajan, "Deep belief network based detection and categorization of malicious URLs," *Inf. Secur. J., Global Perspective*, vol. 27, no. 3, pp. 145–161, Apr. 2018, doi: 10.1080/19393555.2018.1456577.

[76] I. Bello, H. Chiroma, U. A. Abdullahi, A. Y. Gital, F. Jauro, A. Khan, J. O. Okesola, and S. M. Abdulhamid, "Detecting ransomware attacks using intelligent algorithms: Recent development and next direction from deep learning and big data perspectives," *J. Ambient Intell. Humanized Comput.*, vol. 12, pp. 8699–8717, Nov. 2020, doi: 10.1007/s12652-020-02630-7.

[77] M. Chatterjee and A.-S. Namin, "Detecting phishing websites through deep reinforcement learning," in *Proc. IEEE 43rd Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2019, pp. 227–232, doi: 10.1109/COMPSAC.2019.10211.

[78] M. Arshey and K. S. A. Viji, "An optimization-based deep belief network for the detection of phishing e-mails," *Data Technol. Appl.*, vol. 54, no. 4, pp. 529–549, Jul. 2020, doi: 10.1108/DTA-02-2020-0043.

[79] E. Castillo, S. Dhaduvai, P. Liu, K.-S. Thakur, A. Dalton, and T. Strzalkowski, "Email threat detection using distinct neural network approaches," in *Proc. 1st Int. Workshop Social Threats Online Conversations: Understand. Manage.*, Marseille, France, May 2020, pp. 48–55. Accessed: Mar. 10, 2021. [Online]. Available: https://www.aclweb.org/anthology/2020.stoc-1.8

[80] L. Halgaš, I. Agrafiotis, and J. R. C. Nurse, "Catching the phish: Detecting phishing attacks using recurrent neural networks (RNNs)," in *Information Security Applications*. Cham, Switzerland: Springer, 2020, pp. 219–233, doi: 10.1007/978-3-030-39303-8_17.

[81] R. Alotaibi, I. Al-Turaiki, and F. Alakeel, "Mitigating email phishing attacks using convolutional neural networks," in *Proc. 3rd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, Mar. 2020, pp. 1–6, doi: 10.1109/ICCAIS48893.2020.9096821.

[82] J. Feng, L. Zou, O. Ye, and J. Han, "Web2 Vec: Phishing webpage detection method based on multidimensional features driven by deep learning," *IEEE Access*, vol. 8, pp. 221214–221224, 2020, doi: 10.1109/ACCESS.2020.3043188.

[83] Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism," *IEEE Access*, vol. 7, pp. 56329–56340, 2019, doi: 10.1109/ACCESS.2019.2913705.

[84] S. Phomkeona and K. Okamura, "Zero-day malicious email investigation and detection using features with deep-learning approach," *J. Inf. Process.*, vol. 28, pp. 222–229, 2020, doi: 10.2197/ipsjjip.28.222.

[85] H. Zhu, "Online meta-learning firewall to prevent phishing attacks," *Neural Comput. Appl.*, vol. 32, no. 23, pp. 17137–17147, Dec. 2020, doi: 10.1007/s00521-020-05041-z.

[86] M. Nguyen, T. Nguyen, and T. Huu Nguyen, "A deep learning model with hierarchical LSTMs and supervised attention for anti-phishing," 2018, *arXiv:1805.01554*.

[87] R. Vinayakumar, H. B. B. Ganesh, M. A. Kumar, K. P. Soman, and P. Poornachandran, "DeepAnti-PhishNet: Applying deep neural networks for phishing email detection," in *Proc. CEUR Workshop*, vol. 2124, Mar. 2018, pp. 39–49.

[88] G. K. Soon, C. K. On, N. M. Rusli, T. S. Fun, R. Alfred, and T. T. Guan, "Comparison of simple feedforward neural network, recurrent neural network and ensemble neural networks in phishing detection," in *Proc. J. Phys., Conf.*, Mar. 2020, vol. 1502, no. 1, Art. no. 012033, doi: 10.1088/1742-6596/1502/1/012033.

[89] G. K. Soon, L. C. Chiang, C. K. On, N. M. Rusli, and T. S. Fun, "Comparison of ensemble simple feedforward neural network and deep learning neural network on phishing detection," in *Computational Science and Technology*, Singapore: Springer, 2020, pp. 595–604, doi: 10.1007/978-981-15-0058-9_57.

[90] M. A. Adebowale, K. T. Lwin, and M. A. Hossain, "Deep learning with convolutional neural network and long short-term memory for phishing detection," in *Proc. 13th Int. Conf. Softw., Knowl., Inf. Manage. Appl. (SKIMA)*, Aug. 2019, pp. 1–8, doi: 10.1109/SKIMA47702.2019.8982427.

[91] M. A. Adebowale, K. T. Lwin, and M. A. Hossain, "Intelligent phishing detection scheme using deep learning algorithms," *J. Enterprise Inf. Manage.*, Jun. 2020, doi: 10.1108/JEIM-01-2020-0036.

[92] P. Robic-Butez and T. Y. Win, "Detection of phishing websites using generative adversarial network," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 3216–3221, doi: 10.1109/BigData47090.2019.9006352.

[93] R. S. Rao, T. Vaishnavi, and A. R. Pais, "PhishDump: A multi-model ensemble based technique for the detection of phishing sites in mobile devices," *Pervas. Mobile Comput.*, vol. 60, Nov. 2019, Art. no. 101084, doi: 10.1016/j.pmcj.2019.101084.

[94] T. Rasymas and L. Dovydaitis, "Detection of phishing URLs by using deep learning approach and multiple features combinations," *Baltic J. Modern Comput.*, vol. 8, no. 3, pp. 471–483, Sep. 2020, doi: 10.22364/bjmc.2020.8.3.06.

[95] H. Shirazi, S. R. Muramudalige, I. Ray, and A. P. Jayasumana, "Improved phishing detection algorithms using adversarial autoencoder synthesized data," in *Proc. IEEE 45th Conf. Local Comput. Netw. (LCN)*, Nov. 2020, pp. 24–32, doi: 10.1109/LCN48667.2020.9314775.

[96] S. Sountharrajan, M. Nivashini, S. K. Shandilya, E. Suganya, A. Bazila Banu, and M. Karthiga, "Dynamic recognition of phishing URLs using deep learning techniques," in *Advances in Cyber Security Analytics and Decision Systems*, S. K. Shandilya, N. Wagner, and A. K. Nagar, Eds. Cham, Switzerland: Springer, vol. 2020, pp. 27–56, doi: 10.1007/978-3-030-19353-9_3.

[97] W. Wang, F. Zhang, X. Luo, and S. Zhang, "PDRCNN: Precise phishing detection with recurrent convolutional neural networks," *Secur. Commun. Netw.*, vol. 2019, Oct. 2019, Art. no. e2595794, doi: 10.1155/2019/2595794.

[98] J. Ya, T. Liu, P. Zhang, J. Shi, L. Guo, and Z. Gu, "NeuralAS: Deep word-based spoofed URLs detection against strong similar samples," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–7, doi: 10.1109/IJCNN.2019.8852416.

[99] B. Janet and S. Reddy, "Anti-phishing system using LSTM and CNN," in *Proc. IEEE Int. Conf. for Innov. Technol. (INOCON)*, Nov. 2020, pp. 1–5, doi: 10.1109/INOCON50539.2020.9298298.

[100] X. Yu, "Phishing websites detection based on hybrid model of deep belief network and support vector machine," in *Proc. IOP Conf., Earth Environ. Sci.*, vol. 602, Nov. 2020, Art. no. 012001, doi: 10.1088/1755-1315/602/1/012001.

[101] S. Srinivasan, R. Vinayakumar, A. Arunachalam, M. Alazab, and K. Soman, "DURLD: Malicious URL detection using deep learning-based character level representations," in *Malware Analysis Using Artificial Intelligence and Deep Learning*, M. Stamp, M. Alazab, and A. Shalaginov, Eds. Cham, Switzerland: Springer, 2021, pp. 535–554, doi: 10.1007/978-3-030-62582-5_21.

[102] D. Liu, J.-H. Lee, W. Wang, and Y. Wang, "Malicious websites detection via CNN based screenshot Recognition*," in *Proc. Int. Conf. Intell. Comput. Emerg. Appl. (ICEA)*, Aug. 2019, pp. 115–119, doi: 10.1109/ICEA.2019.8858300.

[103] S. Kurnaz and W. Gwad, "Deep auto-encoder neural network for phishing website classification," *Int. J. Comput. Sci. Mobile Comput.*, vol. 73, no. 3, pp. 68–72, 2018.

[104] 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, Sep. 2020, Art. no. e8694796, doi: 10.1155/2020/8694796.

[105] M. Orabi, D. Mouheb, Z. Al Aghbari, and I. Kamel, "Detection of bots in social media: A systematic review," *Inf. Process. Manage.*, vol. 57, no. 4, Jul. 2020, Art. no. 102250, doi: 10.1016/j.ipm.2020.102250.

[106] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," *Expert Syst. Appl.*, vol. 117, pp. 345–357, Mar. 2019, doi: 10.1016/j.eswa.2018.09.029.

[107] D. Chen, P. Wawrzynski, and Z. Lv, "Cyber security in smart cities: A review of deep learning-based applications and case studies," *Sustain. Cities Soc.*, vol. 66, Mar. 2021, Art. no. 102655, doi: 10.1016/j.scs.2020.102655.

[108] P. Yi, Y. Guan, F. Zou, Y. Yao, W. Wang, and T. Zhu, "Web phishing detection using a deep learning framework," *Wireless Commun. Mobile Comput.*, vol. 2018, Sep. 2018, Art. no. e4678746, doi: 10.1155/2018/4678746.

[109] R. Wason, "Deep learning: Evolution and expansion," *Cognit. Syst. Res.*, vol. 52, pp. 701–708, Dec. 2018, doi: 10.1016/j.cogsys.2018.08.023.

[110] S. Chen, L. Fan, C. Chen, M. Xue, Y. Liu, and L. Xu, "GUI-squatting attack: Automated generation of Android phishing apps," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 6, pp. 2551–2568, Nov./Dec. 2021, doi: 10.1109/TDSC.2019.2956035.

[111] Q. Li, M. Cheng, J. Wang, and B. Sun, "LSTM based phishing detection for big email data," *IEEE Trans. Big Data*, vol. 8, no. 1, pp. 278–288, Feb. 2022, doi: 10.1109/TBDATA.2020.2978915.

[112] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016, doi: 10.1016/j.neucom.2015.09.116.

[113] A. Gupta, H. K. Thakur, R. Shrivastava, P. Kumar, and S. Nag, "A big data analysis framework using apache spark and deep learning," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2017, pp. 9–16, doi: 10.1109/ICDMW.2017.9.

[114] Y. Wu, D. Wei, and J. Feng, "Network attacks detection methods based on deep learning techniques: A survey," *Secur. Commun. Netw.*, vol. 2020, Aug. 2020, Art. no. e8872923, doi: 10.1155/2020/8872923.

[115] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "DISTILLER: Encrypted traffic classification via multimodal multitask deep learning," *J. Netw. Comput. Appl.*, vols. 183–184, Jun. 2021, Art. no. 102985, doi: 10.1016/j.jnca.2021.102985.

[116] T. Phoka and P. Suthaphan, "Image based phishing detection using transfer learning," in *Proc. 11th Int. Conf. Knowl. Smart Technol. (KST)*, Jan. 2019, pp. 232–237, doi: 10.1109/KST.2019.8687615.

[117] R. Gupta, S. Tanwar, S. Tyagi, and N. Kumar, "Machine learning models for secure data analytics: A taxonomy and threat model," *Comput. Commun.*, vol. 153, pp. 406–440, Mar. 2020, doi: 10.1016/j.comcom.2020.02.008.

[118] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2921–2929. Accessed: Jan. 24, 2022. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/html/Zhou_Learning_Deep_Features_CVPR_2016_paper.html

[119] X. Zhou, K. Jin, Y. Shang, and G. Guo, "Visually interpretable representation learning for depression recognition from facial images," *IEEE Trans. Affect. Comput.*, vol. 11, no. 3, pp. 542–552, Jul. 2020, doi: 10.1109/TAFFC.2018.2828819.

[120] S. Al-Ahmadi. (2020). *A Deep Learning Technique for Web Phishing Detection Combined URL Features and Visual Similarity*. Social Science Research Network, Rochester, NY, USA, SSRN Scholarly Paper ID. Accessed: Mar. 10, 2021. [Online]. Available: https://papers.ssrn.com/abstract=3716033

[121] H. N. Digwal and N. P. Kavya, "Detection of phishing website based on deep learning," *Int. J. Res. Eng., Sci. Manage.*, vol. 3, no. 8, pp. 331–336, Aug. 2020.

[122] S. Elnagar and M. A. Thomas, "A cognitive framework for detecting phishing websites," in *Proc. Int. Conf. Adv. Appl. Cogn. Comput.*, Mar. 2019, pp. 60–64.

[123] A. S. S. V. L. Pooja and M. Sridhar, "Analysis of phishing website detection using CNN and bidirectional LSTM," in *Proc. 4th Int. Conf. Electron., Commun. Aerosp. Technol. (ICECA)*, Nov. 2020, pp. 1620–1629, doi: 10.1109/ICECA49313.2020.9297395.

[124] S. Singh, M. P. Singh, and R. Pandey, "Phishing detection from URLs using deep learning approach," in *Proc. 5th Int. Conf. Comput., Commun. Secur. (ICCCS)*, Oct. 2020, pp. 1–4, doi: 10.1109/ICCCS49678.2020.9277459.

[125] P. Vigneshwaran, A. S. Roy, and M. L. Chowdary, "Multidimensional features driven phishing detection based on deep learning," *Int. Res. J. Eng. Technol.*, vol. 7, no. 6, pp. 3062–3067, Jun. 2020.

NGUYET QUANG DO received the B.S. degree in electrical and electronics engineering and the master's degree in electrical engineering from Universiti Tenaga Nasional (UNITEN), Malaysia, in 2011 and 2014, respectively. She is currently pursuing the Ph.D. degree with the Malaysia–Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM). Prior to that, she was working as a Design Engineer at Sony EMGS Sdn. Bhd. She was also a Research Engineer at Universiti Tenaga Nasional Research and Development (UNITEN R&D), Malaysia. During her master's degree, she has published several conference and journal articles. Her research interests include wireless communication networks, smart grid, networks testing, cyber security, machine learning, and artificial intelligence. She received a scholarship for her bachelor's degree from the Electricity of Vietnam (EVN). She is also an awardee of the Malaysia International Scholarship (MIS) from the Ministry of Higher Education (MOHE), Malaysia, for her postgraduate studies.

ALI SELAMAT (Member, IEEE) received the B.Sc. degree (Hons.) in IT from Teesside University, U.K., in 1997, the M.Sc. degree in distributed multimedia interactive systems from Lancaster University, U.K., in 1998, and the Dr.Eng. degree from Osaka Prefecture University, Japan, in 2003. He is currently the Dean of the Malaysia–Japan International Institute of Technology (MJIIT), which is an educational institute that is established by the Ministry of Higher Education, Malaysia, to enhance Japanese oriented engineering education in Malaysia and Asia with the support from the Government of Japan through the Japanese International Cooperation Agency (JICA) and Universiti Teknologi Malaysia (UTM) together with 29 Japanese University Consortium (JUC). Prior to that, he was a Chief Information Officer (CIO) and the Director of Communication and Information Technology at UTM. He was elected as the Chair of IEEE Computer Society, Malaysia Section, under the Institute of Electrical and Electronics Engineers (IEEE), USA. He was previously assuming the position of Research Dean on the knowledge economy research alliance at UTM. He was a Principal Consultant of big data analytics at the Ministry of Higher Education, in 2010, a member of the Malaysia Artificial Intelligence Roadmaps, from 2020 to 2021, and a keynote speaker in many international conferences. He was a Visiting Professor at Kuwait University and few other universities in Japan, Saudi Arabia, and Indonesia. Currently,

he is a Visiting Professor with the University of Hradec Králové, Czech Republic, and the Kagoshima Institute of Technology, Japan. His research interests include data analytics, digital transformations, knowledge management in higher education, key performance indicators, cloud-based software engineering, software agents, information retrievals, pattern recognition, genetic algorithms, neural networks, and soft computing. He is also currently serving on the Editorial Boards of the international journal of *Knowledge-Based Systems* (Elsevier, The Netherlands), the *International Journal of Intelligent Information and Database Systems* (IJIIDS) (Inderscience Publications, Switzerland), and *Vietnam Journal of Computer Science* (Springer Publications). He is the Program Co-Chair of IEA/AIE 2021: The 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems in Kuala Lumpur, Malaysia.

ONDREJ KREJCAR received the Ph.D. degree in technical cybernetics from the Technical University of Ostrava, Czech Republic, in 2008. From 2016 to 2020, he was the Vice-Dean for science and research at the Faculty of Informatics and Management, University of Hradec Králové (UHK), Czech Republic, where he has been a Vice-Rector for science and creative activities, since June 2020. He is a Full Professor in systems engineering and informatics with the Center for Basic and Applied Research, Faculty of Informatics and Management, UHK, and a Research Fellow at the Malaysia–Japan International Institute of Technology, University of Technology Malaysia, Kuala Lumpur, Malaysia. He is also the Director of the Center for Basic and Applied Research, UHK. At UHK, he is responsible for the Doctoral Study Program in applied informatics, where he is focusing on lecturing on smart approaches to the development of information systems and applications in ubiquitous computing environments. His H-index is 21, with more than 1800 citations received in the Web of Science, where more than 120 IF journal articles are indexed in JCR index. Currently, he is on the Editorial Board of the *Sensors* (MDPI) IF journal (Q1/Q2 at JCR), and several other ESCI indexed journals. He has been a Management Committee Member Substitute of Project COST CA16226, since 2017. He has also been the Vice-Leader and a Management Committee Member of WG4 at Project COST CA17136, since 2018. In 2018, he was the 14th Top Peer Reviewer in Multidisciplinary in the World according to Publons and a Top Reviewer in the Global Peer Review Awards 2019 by Publons. Since 2019, he has been the Chairperson of the Program Committee of the KAPPA Program, Technology Agency of the Czech Republic, and a Regulator of the EEA/Norwegian Financial Mechanism in the Czech Republic (2019–2024). Since 2020, he has also been the Chairperson of Panel 1 (Computer, Physical and Chemical Sciences) of the ZETA Program, Technology Agency of the Czech Republic. From 2014 to 2019, he was the Deputy Chairperson of Panel 7 (Processing Industry, Robotics, and Electrical Engineering) of the Epsilon Program, Technology Agency of the Czech Republic.

ENRIQUE HERRERA-VIEDMA (Fellow, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, in 1993 and 1996, respectively. He is currently a Professor of computer science and AI and the Vice-President for research and knowledge transfer with the University of Granada. His H-index is 69 (more than 17 000 citations received in the Web of Science and 85 in Google Scholar), with more than 29 000 cites received. He has been identified as one of the World's most influential researchers by the Shanghai Centre and Thomson Reuters/Clarivate Analytics in both the scientific categories of computer science and engineering, from 2014 to 2018. His current research interests include group decision making, consensus models, linguistic modeling, aggregation of information, information retrieval, bibliometric, digital libraries, web quality evaluation, recommender systems, blockchain, smart cities, and social media.

HAMIDO FUJITA (Life Senior Member, IEEE) received the Doctor Honoris Causa degrees from Óbuda University, Budapest, Hungary, in 2013, and from Politehnica University Timisoara, Timişoara, Romania, in 2018. He received the title of Honorary Professor from Óbuda University, in 2011. He is an Emeritus Professor with Iwate Prefectural University, Takizawa, Japan. He is currently the Executive Chairperson at i-SOMET Incorporated Association, Morioka, Japan. He is a Highly Cited Researcher in cross-field and in the filed of computer science by Clarivate Analytics, in 2019 and 2020, respectively. He is a Distinguished Research Professor at the University of Granada and an Adjunct Professor with Stockholm University, Stockholm, Sweden; the University of Technology Sydney, Ultimo, NSW, Australia; and the National Taiwan Ocean University, Keelung, Taiwan. He has jointly supervised Ph.D. students at Laval University, Quebec City, QC, Canada; the University of Technology Sydney; Oregon State University, Corvallis, OR, USA; the University of Paris 1 Pantheon-Sorbonne, Paris, France; and the University of Genoa, Italy. He has four international patents in software systems and several research projects with Japanese industry and partners. He headed a number of projects including the intelligent HCI, a project related to mental cloning for healthcare systems as an intelligent user interface between human users and computers, and the SCOPE project on virtual doctor systems for medical applications. He collaborated with several research projects in Europe, and recently he is collaborating in the OLIMPIA Project supported by the Tuscany region on therapeutic monitoring of Parkinson's disease. He has published more than 400 highly cited papers. He was the recipient of the Honorary Scholar Award from the University of Technology Sydney, in 2012. He is the Emeritus Editor-in-Chief of *Knowledge-Based Systems* and currently the Editor-in-Chief of *Applied Intelligence* (Springer).