applied sciences

MDPI

Article

Detecting Phishing URLs Based on a Deep Learning Approach to Prevent Cyber-Attacks

Qazi Emad ul Haq [1,*], Muhammad Hamza Faheem [1] and Iftikhar Ahmad [2]

[1] Centre of Excellence in Cybercrime and Digital Forensics, Naif Arab University for Security Sciences, Riyadh 14812, Saudi Arabia
[2] Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
* Correspondence: qabdulrab@nauss.edu.sa

Abstract: Phishing is one of the most widely observed types of internet cyber-attack, through which hundreds of clients using different internet services are targeted every day through different replicated websites. The phishing attacker spreads messages containing false URL links through emails, social media platforms, or messages, targeting people to steal sensitive data like credentials. Attackers generate phishing URLs that resemble those of legitimate websites to gain these confidential data. Hence, there is a need to prevent the siphoning of data through the duplication of trustworthy websites and raise public awareness of such practices. For this purpose, many machine learning and deep learning models have been employed to detect and prevent phishing attacks, but due to the ever-evolving nature of these attacks, many systems fail to provide accurate results. In this study, we propose a deep learning-based system using a 1D convolutional neural network to detect phishing URLs. The experimental work was performed using datasets from Phish-Tank, UNB, and Alexa, which successfully generated 200 thousand phishing URLs and 200 thousand legitimate URLs. The experimental results show that the proposed system achieved 99.7% accuracy, which was better than the traditional models proposed for URL-based phishing detection.

Keywords: phishing; deep learning; convolutional neural network; deep neural network

check for updates

Citation: Haq, Q.E.u.; Faheem, M.H.; Ahmad, I. Detecting Phishing URLs Based on a Deep Learning Approach to Prevent Cyber-Attacks. *Appl. Sci.* **2024**, *14*, 10086. https://doi.org/10.3390/app142210086

Academic Editor: Gianluca Lax

Received: 20 August 2024
Revised: 29 September 2024
Accepted: 4 October 2024
Published: 5 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The vast web of the internet has brought huge changes to human life. It facilitates access to every necessity with just a few clicks, from online shopping to online bill paying, and it significantly influences human life. Using millions of websites, people complete millions of transactions online every day for various purposes. Most online services offer their clients a chance to store their valuable data on their servers for reuse and easy access. However, alongside these advantages, there are many disadvantages associated with cybercriminals. By using different types of methods and techniques, the personal data of clients can be stolen from trustworthy websites and can be misused in different ways [1]. One of the most common techniques of cybercrime is phishing.

Phishing attacks represent a rapidly growing crime in the cyber world. The major cause identified is the careless behavior of users while engaged in social activities. Phishing is defined as a social-engineering-based method of attack that involves stealing credentials and sensitive information to carry out financial fraud, theft, etc. [2–4]. A phishing attack scenario is shown in Figure 1. Online fraudsters often steal sensitive information by using phishing URLs to direct victims to scam pages that impersonate legitimate websites and appear very similar to the originals [5]. Phishing URLs quickly deceive people, who then become victims of fraud. This succeeds as people do not pay sufficient attention to URLs, either due to a lack of knowledge or careless behavior. By making copies of legitimate websites, criminals can easily deceive people.

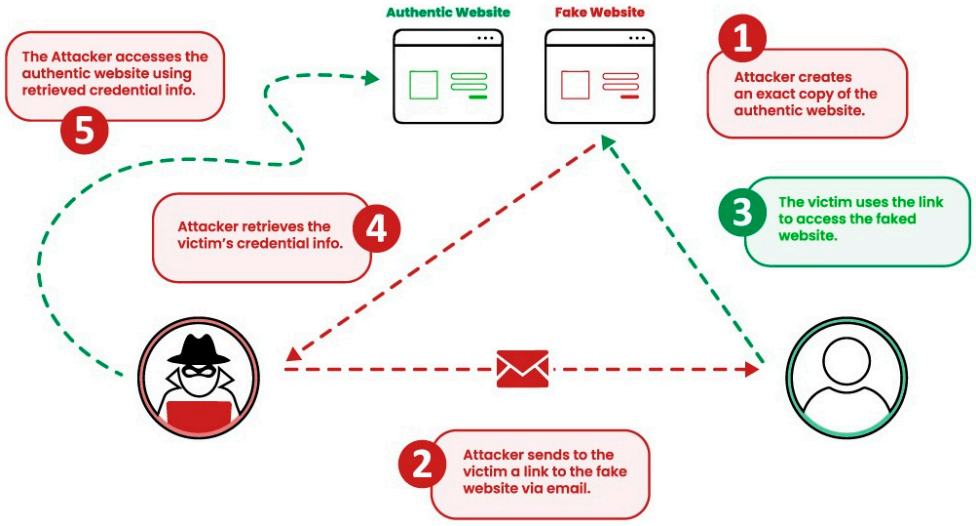


Figure 1. An overview of phishing websites.

Phishing websites are nearly identical to actual websites. Scammers, without any margin of error, duplicate legitimate websites, making differences between the real and copy websites indistinguishable from the naked eye. The scammers make sure that the interface and user functionalities provided by the phishing websites are identical to those provided by the legitimate websites [5].

The Anti-Phishing Working Group (APWG) has generated a report showing that phishing attacks in the fourth quarter of 2023 involved 1,077,501 sites. It observed that there were almost five million phishing attacks in 2023, the highest recorded year for phishing on record [6]. In the fourth quarter of 2023, according to the APWG, the industries most frequently targeted were those shown in Figure 2. It is clear that the number of phishing attacks has increased significantly, and there is a need to take crucial steps to prevent these attacks by using efficient approaches. The demand for effective countermeasures has made phishing detection an important area of research in the past two decades, from which different categories of phishing detection have emerged. The main categories of phishing detection are as follows: (a) blacklist- and whitelist-based approaches; (b) web-page-based visual similarity; and (c) URL- and web-content-based feature extraction. Many machine learning- and deep learning-based approaches have been employed to extract URL- and web-based features to improve the detection of phishing attacks [7,8]. These approaches have also been used to improve the performance of detection systems.

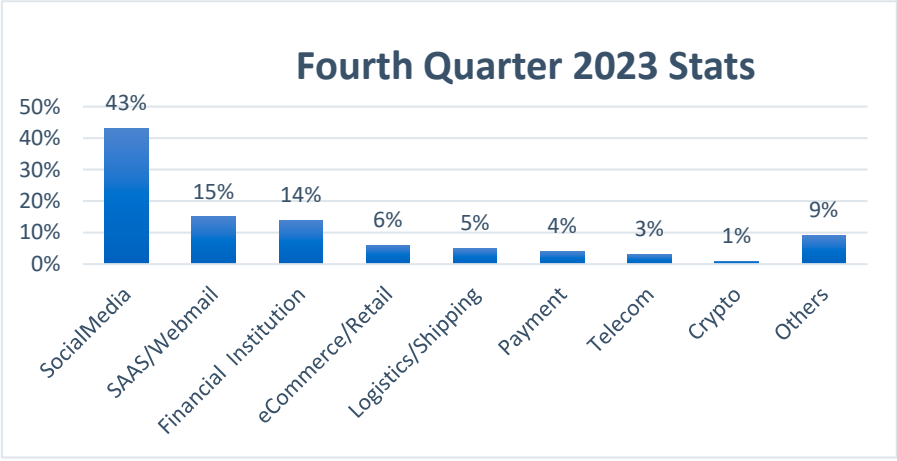


Figure 2. The industries most targeted by phishing attacks in 2023.

Due to the varying limitations of these studies and the continuous evolution of phishing attacks, no proposed approach has been sufficient to provide a comprehensive mechanism that detects and prevents phishing attacks. By using a deep-learning-based approach, this study proposes a one-dimensional convolutional neural network (1D CNN) for highly robust phishing URL detection. The 1D CNN is low-cost and well-suited to real-world scenarios, as it has a shallow architecture that can learn challenging tasks quickly and as it can run on any common CPU and hardware. However, 1D CNNs sometimes struggle to capture complex hierarchical features compared with other architectures like recurrent neural networks (RNNs) or transformers, especially when the data contain long-range dependencies. Accordingly, the study applies a system that uses URL-based feature extraction to detect phishing attacks. The key contributions of this paper are given below:

- This study developed and presents a user-friendly end-to-end web-based system that detects whether a URL is phishing or legitimate;
- It presents a deep learning model using a 1D convolutional neural network to detect URL-based phishing attacks by determining whether a URL is phishing or legitimate;
- It evaluates the proposed system using diverse datasets obtained from PhishTank, UNB, and Alexa;
- This study presents a detailed analysis of existing phishing detection methods, highlighting their limitations and our proposed model's advantages.

The remainder of this paper is organized as follows. Section 2 includes the research work performed about phishing detection, in the form of a literature review. In Section 3, the methodology is presented. Section 4 explains the details of the datasets. Section 5 explains the evaluation measures and gives a comparison with existing approaches. Section 6 discusses limitations and the future application of the proposed methodology, and Section 7 presents the conclusions.

2. Literature Review

Through extensive study, numerous techniques have been developed for the detection of phishing URLs in recent years. Many authors have presented their approaches to protect people from illegitimate sites. Many studies have used machine learning and deep learning methods to achieve highly accurate detection results.

2.1. Traditional Methods

Many traditional phishing detection techniques have been proposed previously, which can be classified into five categories: whitelist, blacklist, content list, visual similarity, and URL-based strategies.

2.1.1. Whitelist Approach

In 2021, an approach was proposed based on whitelist sites, in which they performed URL-based similarity checks to distinguish a phishing site by comparing the domain name system (DNS) query to overcome the phishing attack [9]. Later, in [10], proposed an approach in which an automated individual whitelist in the system maintained the user's previous login and warned them when unfamiliar access occurred. Although these techniques seem effective in phishing detection, there are many limitations faced by legitimate sites on the web.

2.1.2. Blacklist Approach

Google browsers defend against phishing attacks by updating their list of blacklisted sites. A technique was proposed by [11] for blacklist generation to solve common issues in maintaining and updating a list. To compare its results with top-level results, the proposed system used a third-party service like Google to search the domain name, causing performance issues in blacklisting. Furthermore, it encountered its main issues with zero-hour phishing attacks because newly created phishing sites were not on the list [12]. An approach named PhisNet was proposed by [13], based on a blacklisting scheme. The top five

heuristic domains, IP address, directory structure, query string, and brand name were used for the prediction of blacklisted phishing-attack websites. Although zero-hour phishing sites could not be detected, it still achieved 95.0% accuracy in its positive results [14].

2.1.3. Content List Approach

A novel content-list approach called CANTINA was proposed by [15]. A renowned information retrieval algorithm, frequency-inverse document frequency (TF-IDF), was used to detect phishing websites. However, CANTINA had the limitation that the search engine provided many false results, which increased the false positive rate. To overcome this, many heuristic methods were used to improve the results' accuracy. It then attained 97% more accurate results compared with other state-of-the-art tools [16]. CANTINA was then upgraded and renamed CANTINA+ by [17], and this was thought to be the most thorough and feature-rich solution for content-based phishing detection. Over 92% of positive results were accurate, and the false positive rate was improved to 0.4%. However, because both strategies rely on search engines and outside services, DNS compromise has become a difficult challenge [18]. Readers can find related works in [17].

2.1.4. Visual-Similarity-Based Methods

In [19], used a straightforward strategy based on visual similarities. Three levels of similarity matrices, block-level similarity, layout similarity, and overall style similarity, were used to detect phishing. Later, it was reported that the most representative work on visual similarity, by [11], used the Earth Mover Distance (EMD). The signatures of two photographs were computed using the EMD to determine their visual resemblance. With 89% true positive and 0.71% false positive rates, their method was accurate, but it demonstrated lower performance than previous methods due to the laborious processing of two images [20]. In another study, a heuristic anti-phishing method to mimic perceptual similarities was introduced. The researchers used a logistic regression approach to normalize the attributes of the page content. Despite having a 100% true positive rate, the suggested approach had a 0.74% false positive rate, which could be reduced. A study based on aesthetic resemblance was published by [21] and there are numerous comparable works.

2.1.5. URL-Based Methods

Several approaches based on machine learning and deep learning to detect URL-based phishing attacks are discussed below.

Machine Learning-Based Methods

In this study, an approach based on the survey and structural detection of malware-based URL detection techniques using machine learning is proposed. By detecting malicious URLs using different machine learning techniques like feature extraction, feature representation, algorithm design, blacklisting, and heuristic approaches are used to detect malicious URLs and binary classification is applied to classify the results as "malicious" or "benign". Machine learning aims to maximize the prediction accuracy via feature representation followed by classification, using a data-driven optimization approach [22]. In this study, 19,066 phishing attacks were monitored, and over 90% of those attacks were real. Several techniques were used against the phishing attacks, in which quick and efficient detection was achieved using the prevention tool. A phishing detection website was created via a hierarchical clustering approach in which vector numbers of tags were grouped; these numbers were generated from the DOMs of attacked websites according to their proportional distances. Machine learning algorithms were used for the classification of the detected websites [23].

A survey was conducted on the phishing detection approaches examined in this study and malicious URL detection techniques using machine learning. Many formulations have been provided for the detection task, and by utilizing machine learning models, the study identified the categorization, contributions, and different problems that caused a gap in

malicious URL detection. Furthermore, the article reviewed a timely and comprehensive range of studies that are also helpful for practitioners in the cyber-security industry. Moreover, much open research and many open-ended challenges to work on in the future were mentioned [24]. Elsewhere, a real-time anti-phishing tool system was proposed for the detection of malicious URLs in a study that used seven different classification algorithms and features based on natural language processing (NLP). That work was based on previous studies. The study proposed a different system, which was language-independent, using a huge amount of phishing and legitimate data and real item execution from the detection of websites. Feature-rich machine learning classifiers were used for independent servicer detection. A random forest algorithm with NLP-based features was used for the evolution of performance, with a 97.89% accuracy rate [8]. In another case, in a study on the detection of phishing URLs, the researchers adopted a distributed representation of words using a given URL. Seven different machine learning algorithms were used for the prediction of whether websites were phishing sites or not. The algorithms provided a satisfactory performance level, beyond that of past studies. Unseen characters were identified, and loss of semantic information was detected using a bag-of-words model, which retained semantic information [25].

In another study, an anti-phishing protection system was introduced that consisted of an email web browser extension and a machine learning-based phishing detection server. The browser extension was used to extract the URL, capture the screenshot, and store the user visiting history as a profile on the client side. Phishing detection was performed in the following steps: (a) using a machine learning model to predict whether the URL was phishing or not based on 13 different features, (b) using blacklisting and whitelisting on a third-party server to filter the new URL, and (c) analyzing website logs using computer vision technology to detect and compare the similarity of screenshots of web pages. The results showed that the model achieved 95% accuracy using a random forest classifier [26]. Beyond this, another study presented a machine-learning-based browser extension for the detection of URL-based phishing. The researchers trained their UCI dataset using a random forest model and normalized the 30 features, which meant it required extra features for its training in the URL-based string real-time environment. The study found that of 30 features, 16 features did not rely on a third-party service; however, the results showed that the system attained an 89.6% higher accuracy rate than state-of-art systems for the Chrome-based detection of phishing URLs [27].

In another previous study, a website was developed to detect URL-based phishing attacks. The JSoup HTML Parser (JHP) library in Java was used for detection based on three stages: (a) using JSoup to parse the DOM structure of the website to be detected, (b) analyzing the number of links from the DOM structure and analyzing the values of the attributes, and (c) using the linked calculator to figure out the indicated value between 0 and 1. Here, 300 URLs were used for the experimental work to evaluate the performance. The results showed a 99.97% accuracy in URL-based phishing detection using the linked calculator. However, using such a big dataset makes detection much riskier, and many characteristics were overlooked [28].

Deep Learning-Based Methods

A deep learning model using an exposed neural network was proposed. In this model, a raw short character string was taken as an input, and features were extracted and classified via character-level embedding using a convolutional neural network for the detection of phishing URLs. The string included a security input, potential malicious URLs, file paths, etc. Using a deep learning model, deep automated features were designed, and malicious URLs were extracted, yielding 5% to 10% better performance than the state-of-art results [29]. In another study, several new malware classification architectures were proposed, using a long short-term memory (LSTM) language model and a gated recurrent unit (GRU) language model. A single-stage malware classifier was proposed based on a character-level convolutional neural network (CNN). The researchers compared

the results and showed that LSTM attained better results with temporal max pooling and that logistic regression offered 31.3% improved results. It was better at capturing long-term dependencies, and neural language models helped increase the performance of malware detection [30]. GNNs have shown great potential by leveraging graph structures, where key properties such as connectivity and diagnosability are crucial for the performance of GNN models, as they impact the structural integrity of the input graphs. Research on the diagnosability of various graphs provides valuable insights into the robustness and fault tolerance of GNN-based models. Relevant studies that could enhance the models' performance in phishing detection include work on the diagnosability of bubble-sort star graphs under the PMC and MM* models [1]. Additionally, iterative processes for entity relationship and business process model extraction, as discussed in Javed and Lin's work on iMER [2], offer valuable techniques for improving data pattern extraction, which can be adapted to enhance the efficiency of phishing detection models.

In a previous study, an approach was proposed to automatically extract URL features from phishing websites using deep neural networks. A CNN model was employed to extract the character-level spatial features from the embedded URL, bi-directional LSTM was used to classify them and generate them for the detection of the word level, and a temporal attention-based hierarchical RNN model was employed for the representation of the URL. For the detection of phishing URLs, a fused feature representation of a three-layer CNN and a multilayer perceptron (MLP) were used [31]. A deep-learning-based framework was proposed for phishing URL detection, in which a browser plug-in is capable of determining whether the phishing risk in real-time is high or low. Based on the user's visits to web pages, it provides the user with a warning message. Real-time prediction servers were combined to generate multiple strategies to improve the detection of threats to website users through URLs. The used framework was capable of recognizing false alarms, reducing the calculation time of prediction, and detecting the whitelist and blacklist based on machine learning predictions using multiple datasets. The RNN-GRU-based model helped the framework attain an accuracy rate of 99.18%, demonstrating its feasibility [1].

In another previous study, an adaptive, self-structuring artificial neural network was applied for the classification of true and false URL detection. Phishing URL-based features are continuous problems in detection and determination, and the types of web pages are constantly changing. The researchers tried to resolve these issues by automatically processing the structure of the network, which showed a high acceptance of noisy data, fault tolerance, and a high prediction accuracy, achieved by setting different epoch values in the several experiments performed [32]. In a final study from the literature, the authors proposed a system for the detection of malicious URL-based phishing sequences for proxy logs based on a system named event de-noising of the convolution neural network (ED-CNN). The researchers sought to remove benign negative effects from the proposed system by comparing the URL-based websites with other websites. The system helped reduce the implementation cost by about 47% by using a fast CNN for URL phishing detection [33].

3. Methodology

The goal associated with the proposed model was to detect phishing URLs using a 1D CNN based on deep learning. The fundamental premise of this approach was based on taking URLs as inputs and training the model to detect phishing URLs.

3.1. Deep Learning

In 2017, the deep learning concept was first introduced by the Canadian Institute for Advanced Research (CIFAR) [34]. Deep learning is based on learning and improvement through work based on computational algorithms, and it is considered one of the fundamental branches of machine learning [35]. Deep learning enables humans and computers to perform sophisticated computations using hundreds and thousands of input nodes in a few seconds. Machine learning uses simple logic, while deep learning is based on artificial neural networks (ANNs) similar to the human brain. The model has multiple processing

layers, each of which performs special operations at an abstract level. There are three types of layers in deep neural networks based on artificial neural networks: the input layer, hidden layers, and output layers. The structure of a deep neural network is shown below in Figure 3.

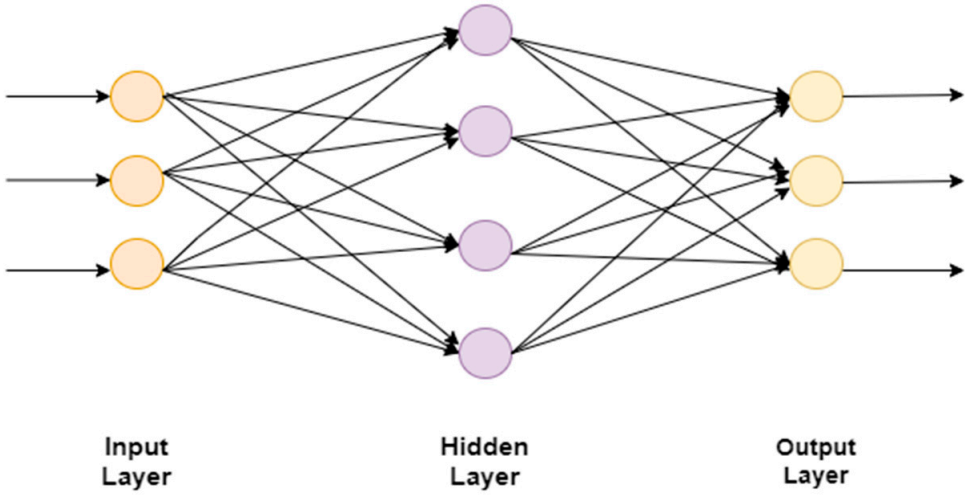


Figure 3. Structure of a deep neural network.

Deep learning provides services in different fields of science with efficient results, such as fraud detection, language translation, healthcare systems, virtual assistants, information systems, computing, and IT [36–38]. Different deep learning algorithms have been proposed so far, of which the most popular are convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory networks (LSTMs), generative adversarial networks (GANs), multilayer perceptrons (MLPs), radial basis function networks (RBFNs), restricted Boltzmann machines (RBMs), auto-encoders, and deep belief networks (DBNs) [39].

3.2. CNN and 1D CNN

In 1988, Yann LeCun proposed the CNN model for the recognition of characters [40]. The CNN is one of the most popular algorithms in deep learning and is a regularized version of a multilayer perceptron. A CNN specifies a deep feed-forward network that works on an artificial neural network (ANN). A CNN model is shown in Figure 2. The workings of a 1D CNN model consist of a CNN with multiple layers inside that process the data and extract features from the data [41]. Among these is the convolutional layer that performs convolutional operations. The ReLU layer is used to perform operations on the elements that provide the output of the rectified feature maps. Those rectified feature maps then feed into the pooling layer, which performs down-sampling operations that reduce the dimensions of the feature maps. These pooling layers convert the results into two-dimensional layers using the pooling features by flattening them. When these flatting matrices are fed in as inputs, fully connected layers are then formed, which identify and classify the results [42]. In a 1D CNN, a single layer of a CNN block connects to train the provided data based on the workflow described above. Details of the 1D CNN model are given in Figure 4.

3.3. Proposed Architecture

The proposed model for URL-based phishing detection is divided into four major sections. The following are the major tasks performed for the detection of legitimate and phishing URLs in the proposed model:

- Data collection;
- Data preprocessing;

- Classification using the proposed deep learning algorithms;
- Web application.

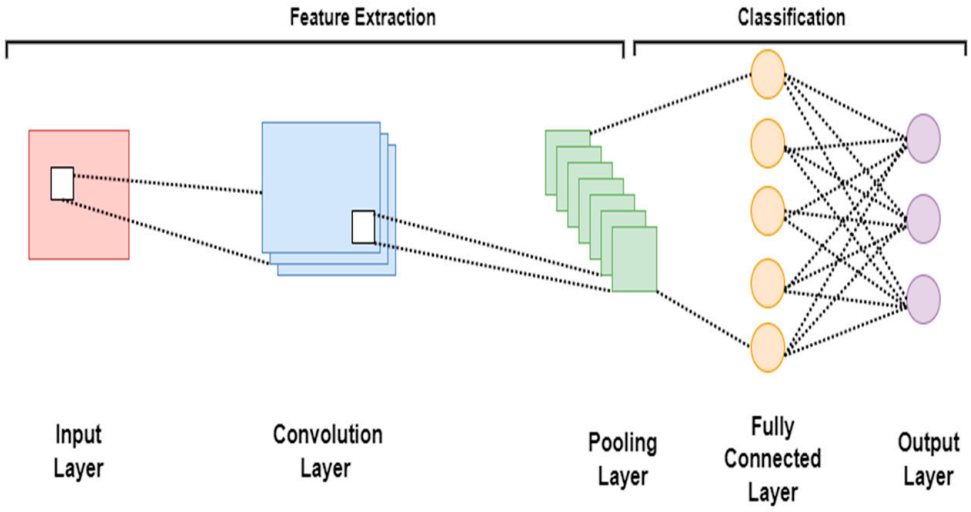


Figure 4. 1D CNN architecture diagram.

All these major tasks are described in detail in Figure 5.

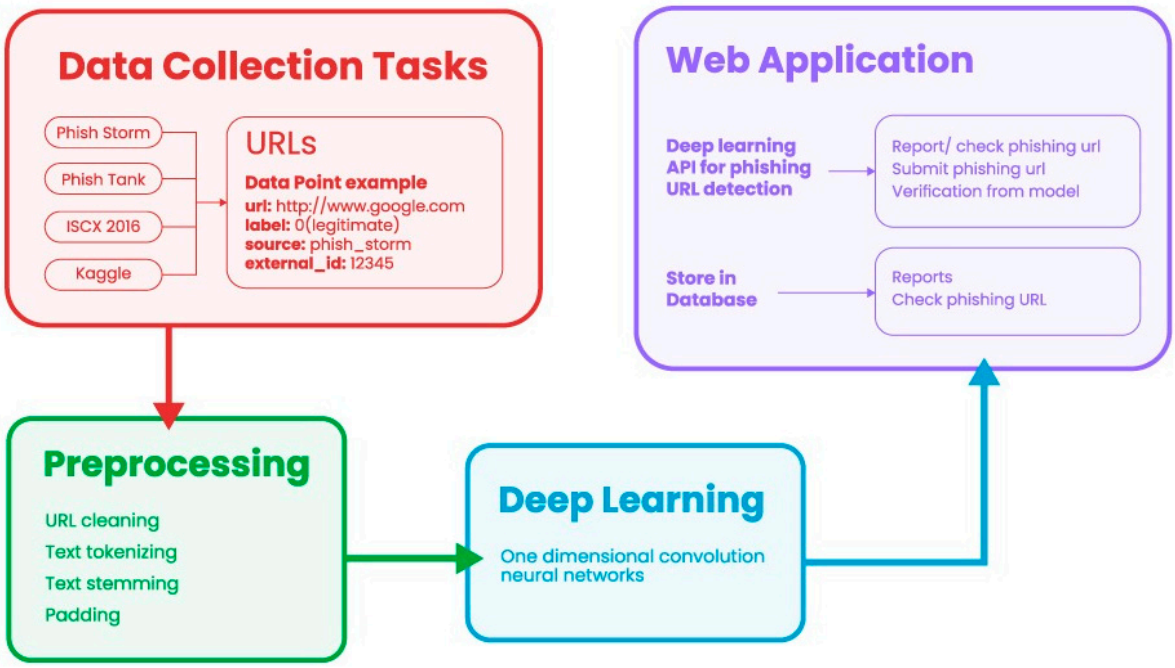


Figure 5. Proposed architecture diagram.

3.3.1. Data Collection

During data collection, data were collected from four open-source datasets, and the important features of URLs were extracted based on parameters, also called the data points. The datasets used in this work were PhishTank, UNB, and Alexa, which are the most popular datasets used for detecting phishing through URLs. The data points used to extract the URLs were the following: URL, labels, source, and external IDs.

3.3.2. Preprocessing

After data collection, data preprocessing was performed, involving the cleaning, tokenization, stemming, and padding of the data from the above-mentioned dataset, as mentioned in Figure 4.

- Data Cleaning: All details are extracted from the URLs and only important features remain to capture information;
- Text Tokenization: The important features of the text are tokenized;
- Text Stemming: Multiple forms of text are converted into one form or stem to simplify the task of analyzing the data;
- Data Padding: P keeps the size of the vector aligned.

3.3.3. Deep Learning Model

A 1D CNN is used to train the model, where the preprocessed data are used for training to detect the phishing URLs. The details of the model are given in the next section.

3.3.4. Web Application

A web application is introduced to identify legitimate and phishing URLs. The main purpose of this website is to protect the user from visiting phishing-based websites, and it provides a mechanism through which the user can check the legitimacy of a suspicious website before actually visiting it. The user can use this website to verify phishing links. The backend of the web application is based on the deep learning algorithm mentioned in the previous section. Once the website verifies that a URL is a phishing URL, the links to that website are stored in the database to prevent its future use.

3.4. 1D CNN Architecture Diagram

The input of preprocessed data from the datasets used is fed into the model for model training and to verify whether a website URL is a phishing site or not. A detailed diagram of the 1D CNN architecture and the workflow is shown in Figure 6. The architecture consists of the following sections:

i. Input Data: Clean URLs are provided as input data for preprocessing;
ii. Preprocessing: Stop words are removed from the URLs, and tokenization is performed as in Figure 4;
iii. Embedding Layers: Data are transferred to the embedding layer, where data dimensioning is performed based on the length of the URL. Suppose that in our model, the length of the URL is 120. Then, 120 dimensions are provided in the embedding layer;
iv. Convolutional Blocks: After embedding the layer, the data are entered into the convolutional block, where seven convolutional layers that have 1D-CNN blocks and one ReLU function have been mapped. Feature mapping and feature extraction of the URL are performed in this layer. One after the other, the inputs are passed through each block to filter out the most important features of the URL;
v. Global Max Pooling: After the convolutional layer, global pooling is performed on the URL's features, where the input size of the matrix is taken with the input of the dimension, and the max value is selected for the computation. Then, the URLs are moved to a deep neural architecture, where dropout is applied;
vi. Drop Out: This is used to prevent the model from overfitting;
vii. Sigmoid: In this step, based on the features identified, the URL is classified as phishing or legitimate.

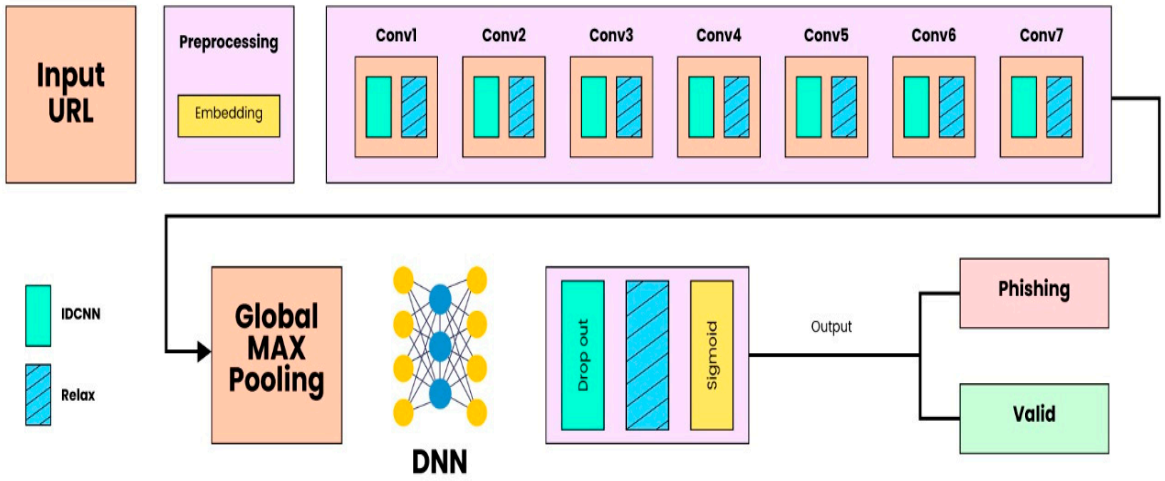


Figure 6. Workflow of 1D CNN model.

4. Datasets

In this study, we used the benchmark datasets PhishTank [32] and UNB [43] to train and test the proposed model of a deep learning system for phishing website detection. PhishTank is a benchmark open-source dataset for phishing URL detection models. It is available for modifications by developers and researchers and is operated by the Cisco Talos Intelligence Group (Talos). The other dataset was taken from the UNB repository operated by the Canadian Institute of Cyber Security, which has various open-source datasets available for the research community to use and test models. These two benchmarks contain both phishing and legitimate website URLs in their repositories. Two major operations were performed on the dataset to normalize it for the best results:

- Dataset preprocessing;
- Dataset splitting.

4.1. Data Preprocessing

During dataset preprocessing, we used different steps to filter out uncertainty from the data, as shown in Figure 1. The dataset was cleaned, during which noise was filtered out and essential features were highlighted. In the next step, we labeled those features and created a new version of the dataset.

4.2. Data Splitting

In this step, the data were divided into training (80%) and test (20%) data [44].

5. Performance Evaluation and Comparison with Existing Approaches

To measure the accuracy of the results of the proposed model, we used four types of evaluation measures: true positive, true negative, false positive, and false negative values. We used the accuracy, recall, precision, and F1-score as evaluation measures to obtain the results, which we compared with those of traditional models. The reason for using these traditional evaluation measures is defined briefly below, where TP indicates a true positive, TN is a true negative, FP is a false positive, and FN is a false negative:

- *Accuracy*: This represents the number of correctly classified data instances over the total number of data instances and is defined as follows:

$$Accuracy = \frac{TP + TN}{FP + FN + TP + TN} \quad (1)$$

- *Recall*: This helped us identify true positive values by giving true positives divided by actual positives:

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

- *Precision*: It gives you the proportion of true positives to the number of total positives that the model predicts and is defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

- *F1 Score*: It takes into account both precision and recall and is defined as follows:

$$F1Score = \frac{2\ (Recall \times Precision)}{Recall + Precision} \tag{4}$$

All the generated results for the testing and the training of the models are given in Table 1, below, which presents the Accuracy, Recall, Precision, and F1 score values.

Table 1. Training and testing.

Metrics	Training	Testing
Accuracy	99.7%	99.3%
Recall	99.8%	99.5%
Precision	99.6%	99.2%
F1 score	99.76%	99.34%

Figures 7 and 8 illustrate the proposed model's accuracy and loss during the validation and the training, achieved using epochs of up to 100.

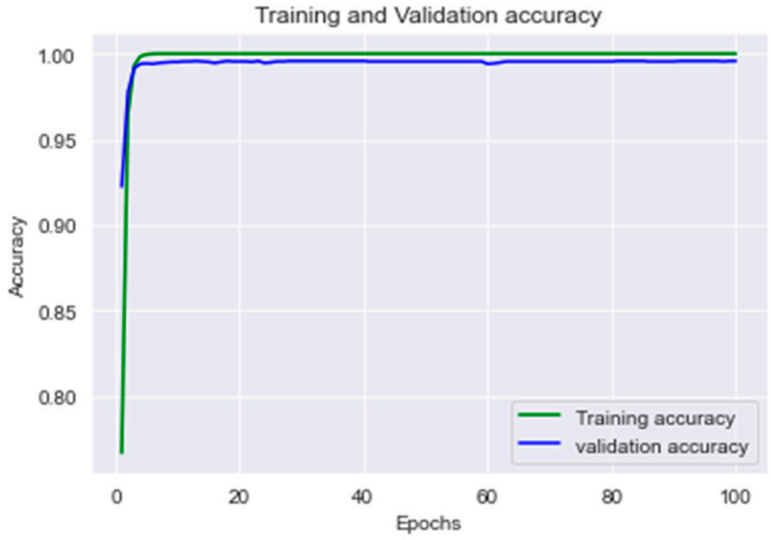


Figure 7. Training and validation accuracy.

In this section, we discuss the experimental work involving our proposed architecture. Due to the widespread use of the internet and social media, there is a great risk of cyber-attacks, and the proposed model is intended to detect one of the most common cyber-attacks, phishing, which can cause massive losses to clients in many ways when criminals steal sensitive information such as credentials and bank details. In this study, we mainly focused on the URL-based detection of phishing attacks via the 1D CNN approach, one of the popular deep learning algorithms. In the proposed model, URLs were taken as inputs and preprocessed using the embedding layer. In the next step, seven convolutional blocks were used to detect the features of the URLs. Max pooling was performed in the next

layer to fetch the matching height and reduce the size and dimensions of the URL, and the dropout and the sigmoid functions were then used to prevent the model from overfitting and to evaluate the results, determining whether the URL was phishing or legitimate. The proposed deep learning-based system was trained on various numbers of hyperparameters and under different settings. The optimal parameters after fine-tuning are listed in Table 2.

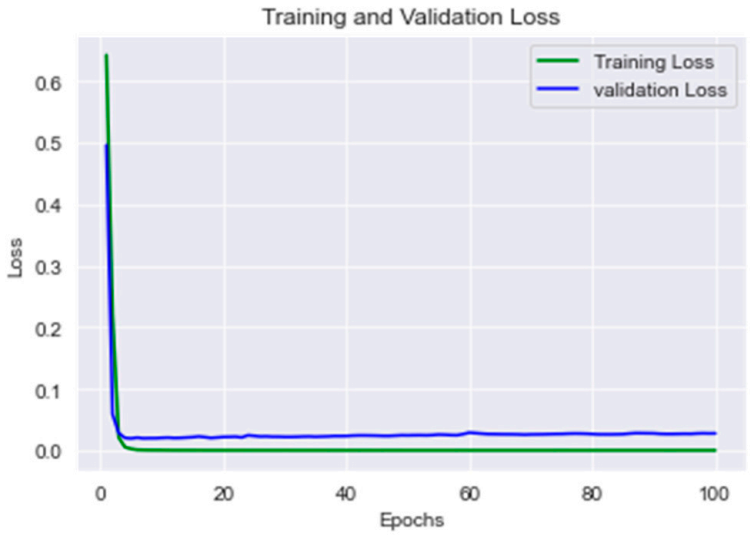


Figure 8. Training and validation loss.

Table 2. Parameter details.

Settings	Parameters
Epochs	500
Loss function	Binary cross-entropy
Optimizer	Adam
Activation function	ReLU
Batch size	500
Dropout	0.2

As shown in Figure 6, the training and the validation accuracies of the model were relatively close to each other, meaning that the model's training did not result in overfitting. The general loss was noted during the experimental work, where the validation and the training loss were relatively close. Some studies on phishing URL detection that we have mentioned are listed in Table 3. In [45] proposed a hybrid deep learning technique that detects public image frames and uses textual information for URL detection via both the CNN and LSTM methods; however, the proposed technique focused more on URL image detection, and the model attained an accuracy of only 93.28% [43]. In [46] proposed an NN based on a dense forward-backward long short-term term memory (LSTM) model (d-FBLSTM) for phishing detection, which was successful at detecting homepage URLs, with the model attaining an accuracy of 98.3% [44]. In 2022, Ahmadi et al. proposed a hybrid model based on LSTM and a CNN for the detection of phishing URLs, and the model attained an accuracy of 97.58%, but the results were not compared to any deep learning or machine learning models [5]. In [47], proposed phishing URL detection using a quadruplet deep neural network based on combining several n-gram embeddings and word embeddings using a gram-embedded dataset, and they attained an accuracy of 98%, but the experimental results were not discussed in detail [48].

Table 3. Comparison of the proposed model with existing approaches.

Ref.	Author, Year	Methodology	Datasets	Limitations	Accuracy
[48]	Adebowale et al., 2023	Hybrid deep learning technique that detects the public image frame and textual information for URL detection utilizing both the CNN and LSTM methods.	Phishing website datasets	The proposed technique is focused more on URL image detection.	93.28%
[45]	Tenis et al., 2023	A dense forward-backward long short-term memory (LSTM) model (d-FBLSTM) was proposed for the detection of phishing URLs.	MUPD	The proposed model detects only home page URLs.	98.5%
[5]	Ahmadi et al., 2022	URL-based phishing detection based on LTSM and CNN models.	PhishTank and DomCop	The results were not compared with a deep learning or machine learning model.	97.58%
[46]	Bozkir et al., 2023	Phishing URL detection using a quadruplet deep neural network based on combining several n-gram embeddings and word embeddings.	Gram Embedding	Not all the evaluation measures mentioned were applied to evaluate the performance of the model.	98%
[47]	Dhanavanthini et al., 2023	A Phish-armor phishing detection model using deep recurrent neural networks that match SSL and website content for the detection of false URLs.	PhishTank, Common Crawl, and Open-phish	A complex and time-consuming computation in Raspberry Pi.	90.50%
[49]	Kumar et al., 2023	An SI-BBA technique with a deep learning model for identifying phishing websites and successfully classifying them.	Phishing URL EDU	The results of the black box phishing attacks can be improved.	94.8%
[50]	Velpula et al., 2023	A CNN is used for detection with random forest models to determine the feature significance at various levels.	5000 phishing emails dataset	The detection results can be improved.	98.68%
[51]	Said et al., 2023	This model combines the multi-head self-attention mechanism with a CNN and generative adversarial network model to create a URL detector.	UCI phishing domains	The results can be improved further.	97.83%
[52]	Saha et al., 2020	A data-driven framework for detecting phishing webpages.	Based on ten thousand web pages	The framework focused on phishing web pages.	93%
[53]	Rasymas et al., 2020	Proposes a deep neural network architecture.	Phishing URLs and benign URLs	The results can be improved further.	94.4%
Proposed model	-	A model is proposed for the detection of phishing URLs based on 1D CNN architecture.	PhishTank, UNB, and Alexa	It takes a long time to train the model.	99.7%

In [49] proposed a Phish-armor phishing detection model using deep recurrent neural networks that matched SSL and website content for the detection of false URLs, but the complex model involved time-consuming computation in Raspberry Pi and attained an accuracy of only 90.50% [45]. In [49] proposed an SI-BBA technique with a deep learning model for identifying phishing websites and successfully classifying them using the dataset Phishing URL EDU; they attained an accuracy of 94.8%, but the model showed low precision [46]. In [49] proposed a CNN to be used for detection with random forest models to determine feature significance at various levels; they attained 98.68% accuracy, but the results and model could have been improved [47]. In 2023, [51] proposed combining a multi-head self-attention mechanism with a CNN and generative adversarial network model to create a URL detector, and the results for UCI phishing domains showed that the model attained an accuracy of 97.83% [49]. Extending these findings, our proposed 1D CNN model has shown 99.7% accuracy; moreover, it has a lightweight architecture and a complete end-to-end web-based system that can distinguish phishing from legitimate websites.

6. Limitations and Future Work

The limitations of the proposed model include the following. A long time is required to train the model, the link can be auto-updated, which can cause changes in results, and it is difficult to detect genetic links that are not easily distinguishable; for example, a phishing attacker may make a fake version of https://www.pizzahut.com (accessed on 18 July 2024), using the URL https://www.pizz.ahut.com (accessed on 18 July 2024), to trap clients.

In the future, we plan to work on content-based phishing detection, in which the whole web page will be examined to detect and identify phishing links present on the page, which could redirect clients away from the website to another phishing page. We aim to propose a bot scroller that will automatically scan the website's content, including its text content, imagery, and links.

7. Conclusions

Instances of internet fraud are increasing nowadays and occur via different methods, among which phishing is one of the most popular. Scammers use fake websites to steal data, with URLs resembling those of original and legitimate websites. To address this issue, in the present study, a model was developed to detect phishing and legitimate websites. For this purpose, we designed a website at which users can enter a URL to differentiate between fake and legitimate URLs. The proposed model is composed of a 1D convolutional neural network (1D CNN). When tested on large-scale datasets from PhishTank, UNB, and Alexa, the proposed model achieved good results and was tested on 200k phishing URLs and 200k legitimate URLs. The proposed model was compared with state-of-the-art architectures, and in the comparison, it achieved the highest accuracy of 99.7%.

Author Contributions: Conceptualization, Q.E.u.H. and M.H.F.; data curation, Q.E.u.H., M.H.F. and I.A.; formal analysis, Q.E.u.H., M.H.F. and I.A.; funding acquisition, Q.E.u.H.; methodology, Q.E.u.H., M.H.F. and I.A.; project administration, Q.E.u.H.; resources, Q.E.u.H.; software, Q.E.u.H. and M.H.F.; supervision, Q.E.u.H.; validation, Q.E.u.H., M.H.F. and I.A.; visualization, Q.E.u.H., M.H.F. and I.A.; writing—original draft, Q.E.u.H. and M.H.F.; writing—review and editing. Q.E.u.H., M.H.F. and I.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Naif Arab University for Security Sciences under grant no. NAUSS-23-R11.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available at https://phishtank.org/ (accessed on 18 July 2024) and https://www.unb.ca/cic/datasets/url-2016.html (accessed on 18 July 2024).

Acknowledgments: The authors would like to express their deep thanks to the Vice Presidency for Scientific Research at Naif Arab University for Security Sciences for their kind encouragement of this work.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tang, L.; Mahmoud, Q.H. A Deep Learning-Based Framework for Phishing Website Detection. *IEEE Access* **2022**, *10*, 1509–1521. [CrossRef]
2. Yerima, S.Y.; Alzaylaee, M.K. High Accuracy Phishing Detection Based on Convolutional Neural Networks. In Proceedings of the ICCAIS 2020–3rd International Conference on Computer Applications and Information Security, Riyadh, Saudi Arabia, 19–21 March 2020.
3. Jakobsson, M.; Myers, S. (Eds.) *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*; John Wiley & Sons: Hoboken, NJ, USA, 2006.
4. Hong, J. The state of phishing attacks. *Commun. ACM* **2012**, *55*, 74–81. [CrossRef]
5. Al-Ahmadi, S.; Alotaibi, A.; Alsaleh, O. PDGAN: Phishing Detection With Generative Adversarial Networks. *IEEE Access* **2022**, *10*, 42459–42468. [CrossRef]
6. Rajitha, K.; Vijayalakshmi, D. Suspicious URLs Filtering Using Optimal RT-PFL: A Novel Feature Selection Based Web URL Detection. In Proceedings of the Smart Innovation, Systems and Technologies, Queensland, Australia, 20–22 June 2018.
7. APWG | Phishing Activity Trends Reports. Available online: https://apwg.org/trendsreports/ (accessed on 19 August 2024).
8. Sahingoz, O.K.; Buber, E.; Demir, O.; Diri, B. Machine Learning Based Phishing Detection from URLs. *Expert Syst. Appl.* **2019**, *117*, 345–357. [CrossRef]
9. Bu, S.J.; Cho, S.B. Deep Character-Level Anomaly Detection Based on a Convolutional Autoencoder for Zero-Day Phishing Url Detection. *Electron* **2021**, *10*, 1492. [CrossRef]
10. Kang, J.M.; Lee, D.H. Advanced White List Approach for Preventing Access to Phishing Sites. In Proceedings of the 2007 International Conference on Convergence Information Technology (ICCIT 2007), Gwangju, Republic of Korea, 21–23 November 2007.
11. Fu, A.Y.; Liu, W.; Deng, X. Detecting Phishing Web Pages with Visual Similarity Assessment Based on Earth Mover's Distance (EMD). *IEEE Trans. Dependable Secur. Comput.* **2006**, *3*, 301–311. [CrossRef]
12. Cao, Y.; Han, W.; Le, Y. Anti-Phishing Based on Automated Individual White-List. In Proceedings of the ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 27–31 October 2008.
13. Oest, A.; Safei, Y.; Doupe, A.; Ahn, G.J.; Wardman, B.; Warner, G. Inside a Phisher's Mind: Understanding the Anti-Phishing Ecosystem through Phishing Kit Analysis. In Proceedings of the 2018 APWG Symposium on Electronic Crime Research (eCrime), San Diego, CA, USA, 15–17 May 2018. [CrossRef]
14. Sharifi, M.; Siadati, S.H. A Phishing Sites Blacklist Generator. In Proceedings of the AICCSA 08–6th IEEE/ACS International Conference on Computer Systems and Applications, Doha, Qatar, 31 March–4 April 2008.
15. Zhang, Y.; Hong, J.I.; Cranor, L.F. Cantina: A Content-Based Approach to Detecting Phishing Web Sites. In Proceedings of the 16th International World Wide Web Conference (WWW2007), Banff, AB, Canada, 8–12 May 2007.
16. Prakash, P.; Kumar, M.; Rao Kompella, R.; Gupta, M. PhishNet: Predictive Blacklisting to Detect Phishing Attacks. In Proceedings of the Proceedings IEEE INFOCOM, San Diego, CA, USA, 14–19 March 2010.
17. Xiang, G.; Hong, J.; Rose, C.P.; Cranor, L. CANTINA+: A Feature-Rich Machine Learning Framework for Detecting Phishing Web Sites. ACM Trans. *Inf. Syst. Secur.* **2011**, *14*, 1–28. [CrossRef]
18. Keivanloo, I.; Roy, C.K.; Rilling, J. SeByte: Scalable Clone and Similarity Search for Bytecode. *Sci. Comput. Program.* **2014**, *95*, 426–444. [CrossRef]
19. Ozker, U.; Sahingoz, O.K. Content Based Phishing Detection with Machine Learning. In Proceedings of the 2020 International Conference on Electrical Engineering (ICEE 2020), Istanbul, Turkey, 25–27 September 2020.
20. Liu, W.; Huang, G.; Liu, X.; Zhang, M.; Deng, X. Detection of Phishing Webpages Based on Visual Similarity. In Proceedings of the 14th International World Wide Web Conference (WWW2005), Chiba, Japan, 10–14 May 2005.
21. Abdelnabi, S.; Krombholz, K.; Fritz, M. VisualPhishNet: Zero-Day Phishing Website Detection by Visual Similarity. In Proceedings of the ACM Conference on Computer and Communications Security, Virtual Event, 9–13 November 2020.
22. Chen, J.L.; Ma, Y.W.; Huang, K.L. Intelligent Visual Similarity-Based Phishing Websites Detection. *Symmetry* **2020**, *12*, 1681. [CrossRef]
23. Nair, S.M. Detecting Malicious URL Using Machine Learning: A Survey. *Int. J. Res. Appl. Sci. Eng. Technol.* **2020**, *8*, 2670–2677. [CrossRef]
24. Cui, Q.; Jourdan, G.V.; Bochmann, G.V.; Couturier, R.; Onut, I.V. Tracking Phishing Attacks over Time. In Proceedings of the 26th International World Wide Web Conference (WWW 2017), Perth, Australia, 3–7 April 2017.
25. Alfouzan, N.A.; Narmatha, C. A Systematic Approach for Malware URL Recognition. In Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT 2022), Tabuk, Saudi Arabia, 25–27 January 2022.
26. Orunsolu, A.A.; Sodiya, A.S.; Akinwale, A.T. A Predictive Model for Phishing Detection. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 232–247. [CrossRef]

27. Atimorathanna, D.N.; Ranaweera, T.S.; Devdunie Pabasara, R.A.H.; Perera, J.R.; Abeywardena, K.Y. NoFish; Total Anti-Phishing Protection System. In Proceedings of the ICAC 2020 2nd International Conference on Advancements in Computing, Colombo, Sri Lanka, 10–11 December 2020.
28. Shah, B.; Dharamshi, K.; Patel, M.; Gaikwad, D.; Professor, A. Chrome Extension for Detecting Phishing Websites. *Int. Res. J. Eng. Technol.* **2020**, *7*, 2958–2962.
29. Abiodun, O.; Sodiya, A.S.; Kareem, S.O. LINKCALCULATOR–AN EFFICIENT LINK-BASED PHISHING DETECTION TOOL. *Acta Inform. Malaysia* **2020**, *4*, 37–44. [CrossRef]
30. Wu, J.; Yang, Z.; Guo, L.; Li, Y.; Liu, W. Convolutional Neural Network with Character Embeddings for Malicious Web Request Detection. In Proceedings of the Proceedings–2019 IEEE Intl Conf on Parallel and Distributed Processing with Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking, ISPA/BDCloud/SustainCom/SocialCom 2019, Xiamen, China, 16–18 December 2019; pp. 622–627.
31. Athiwaratkun, B.; Stokes, J.W. Malware Classification with LSTM and GRU Language Models and a Character-Level CNN. In Proceedings of the ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing–Proceedings, New Orleans, LA, USA, 5–9 March 2017; pp. 2482–2486.
32. Huang, Y.; Yang, Q.; Qin, J.; Wen, W. Phishing URL Detection via CNN and Attention-Based Hierarchical RNN. In Proceedings of the Proceedings–2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE 2019, Rotorua, New Zealand, 5–8 August 2019; pp. 112–119.
33. Mohammad, R.M.; Thabtah, F.; McCluskey, L. Predicting Phishing Websites Based on Self-Structuring Neural Network. *Neural Comput. Appl.* **2014**, *25*, 443–458. [CrossRef]
34. Shibahara, T.; Yamanishi, K.; Takata, Y.; Chiba, D.; Akiyama, M.; Yagi, T.; Ohsita, Y.; Murata, M. Malicious URL Sequence Detection Using Event De-Noising Convolutional Neural Network. In Proceedings of the IEEE International Conference on Communications, Paris, France, 21–25 May 2017; pp. 1–7.
35. Janiesch, C.; Zschech, P.; Heinrich, K. Machine Learning and Deep Learning. *Electron. Mark.* **2021**, *31*, 685–695. [CrossRef]
36. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *J. Big Data* **2021**, *8*, 1–74. [CrossRef]
37. Bolhasani, H.; Mohseni, M.; Rahmani, A.M. Deep Learning Applications for IoT in Health Care: A Systematic Review. Informatics Med. *Unlocked* **2021**, *23*, 100550. [CrossRef]
38. Hassani, H.; Huang, X.; Silva, E.; Ghodsi, M. Deep Learning and Implementations in Banking. *Ann. Data Sci.* **2020**, *7*, 433–446. [CrossRef]
39. Alahmari, S.S.; Goldgof, D.B.; Mouton, P.R.; Hall, L.O. Challenges for the Repeatability of Deep Learning Models. *IEEE Access* **2020**, *8*, 211860–211868. [CrossRef]
40. Guo, T.; Dong, J.; Li, H.; Gao, Y. Simple convolutional neural network on image classification. In Proceedings of the 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), Beijing, China, 10–12 March 2017; pp. 721–724.
41. Singh, K.; Scholar, R.; Mahajan, A.; Mansotra, V. 1D-CNN Based Model for Classification and Analysis of Network Attacks. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 0121169. [CrossRef]
42. Xiao, X.; Xiao, W.; Zhang, D.; Zhang, B.; Hu, G.; Li, Q.; Xia, S. Phishing Websites Detection via CNN and Multi-Head Self-Attention on Imbalanced Datasets. *Comput. Secur.* **2022**, *108*, 102372. [CrossRef]
43. Atrees, M.; Ahmad, A.; Alghanim, F. Enhancing Detection of Malicious Urls Using Boosting and Lexical Features. *Intell. Autom. Soft Comput* **2022**, *31*, 1405–1422. [CrossRef]
44. Pawluszek-Filipiak, K.; Borkowski, A. On the Importance of Train-Test Split Ratio of Datasets in Automatic Landslide Detection by Supervised Classification. *Remote Sens.* **2020**, *12*, 3054. [CrossRef]
45. Tenis, A.A.; Santhosh, R. Modelling an Efficient URL Phishing Detection Approach Based on a Dense Network Model. *Comput. Syst. Sci. Eng.* **2023**, *47*, 2625–2641. [CrossRef]
46. Bozkir, A.S.; Dalgic, F.C.; Aydos, M. GramBeddings: A New Neural Network for URL Based Identification of Phishing Web Pages Through N-Gram Embeddings. *Comput. Secur.* **2023**, *124*, 102964. [CrossRef]
47. Dhanavanthini, P.; Chakkravarthy, S.S. Phish-Armour: Phishing Detection Using Deep Recurrent Neural Networks. *Soft Comput.* **2023**. [CrossRef]
48. Adebowale, M.A.; Lwin, K.T.; Hossain, M.A. Intelligent Phishing Detection Scheme Using Deep Learning Algorithms. *J. Enterp. Inf. Manag.* **2023**, *36*, 747–766. [CrossRef]
49. Kumar, P.P.; Jaya, T.; Rajendran, V. SI-BBA–A Novel Phishing Website Detection Based on Swarm Intelligence with Deep Learning. *Mater. Today Proc.* **2021**, *80*, 3129–3139. [CrossRef]
50. Siva Satya Sreedhar, P.; Velpula, S.; Parise, R.; Vamsi, N.K.; Chaitanya, S.K. Phishing Attack Detection Using Convolutional Neural Networks. In Proceedings of the 2023 9th International Conference on Advanced Computing and Communication Systems, ICACCS 2023, Coimbatore, India, 17–18 March 2023.
51. Said, Y.; Alsheikhy, A.A.; Lahza, H.; Shawly, T. Detecting Phishing Websites through Improving Convolutional Neural Networks with Self-Attention Mechanism. *Ain Shams Eng. J.* **2024**, *15*, 102643. [CrossRef]

52. Saha, I.; Sarma, D.; Chakma, R.J.; Alam, M.N.; Sultana, A.; Hossain, S. Phishing attacks detection using deep learning approach. In Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 20–22 August 2020; pp. 1180–1185.
53. Rasymas, T.; Dovydaitis, L. Detection of phishing URLs by using deep learning approach and multiple features combinations. *Balt. J. Mod. Comput.* **2020**, *8*, 471–483. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.