NIPES - Journal of Science and Technology Research
www.nipes.org

Development of a Hybridized CNN-BiGRU Framework for Detection of Website Phishing Attacks

Abdullahi Raji Egigogo[1,2]*, Ismaila Idris[1], Morufu Olalere[3], Opeyemi Aderike Abisoye[4], Joseph Adebayo Ojeniyi[1]

[1] Cyber Security Science Department, Federal University of Technology Minna, Nigeria.
[2] Department of Software Engineering and Cyber Security, Al-Qalam University, Katsina, Nigeria.
[3]Cyber Security Science Department, National Open University Abuja, Nigeria
[4]Computer Science Department, Federal University of Technology Minna, Nigeria
*Corresponding Author: rajiaegigogo@auk.edu.ng; abdullahirajiegigogo@gmail.com

Article Info

Keywords: *Phishing detection, Website, Convolutional Neural Networks (CNN), Bidirectional Gated Recurrent Units (BiGRU), Hybrid framework*

Received 13 February 2025
Revised 08 March 2025
Accepted 22 April 2025
Available online 20 May 2025

https://doi.org/10.37933/nipes/7.2.2025.18

eISSN-2682-5821, pISSN-2734-2352

Abstract

Phishing remains a major cybersecurity challenge, with attackers using deceptive tactics to trick users into disclosing confidential data. Traditional detection systems, which often rely on fixed features or predefined rules, struggle to keep up with rapidly evolving phishing strategies. This research introduces a deep learning-based solution that combines Convolutional Neural Networks (CNN) and Bidirectional Gated Recurrent Units (BiGRU) to improve phishing website detection. The CNN component is responsible for learning spatial patterns from web data, while the BiGRU layer captures sequential relationships, providing a more complete understanding of the underlying threats. The framework involves meticulous preprocessing steps such as data cleaning, normalization through MinMax scaling, and optimal feature selection using the SelectKBest, CNN and BiGRU methods. The model was trained and tested on large-scale, publicly available datasets from IEEE Data Port and Mendeley, consisting of over 250,000 URL entries. Through train-test split and cross-validation techniques, the model consistently achieved outstanding results: 99.96% accuracy, 99.92% precision, 100% recall, and a 99.92% F1 score. When compared to existing solutions, this hybrid approach sets a new performance benchmark, underscoring the power of combining spatial and temporal deep learning methods in defending against phishing threats.

1.0 Introduction

Phishing attacks represent a serious and ever-evolving danger in the digital landscape. These schemes deceive individuals into sharing confidential details, such as passwords, credit card information, and usernames, by masquerading as real and reliable sources [1], [2]. Web-based phishing, in particular, involves creating fraudulent websites that imitate those of reputable organizations, tricking users into entering their confidential information. The increasing sophistication of these deceptive practices necessitates advanced detection mechanisms to protect users from phishing scams [3]. Traditional detection methods rely on static features and predefined rules and often struggle to identify newly created phishing sites. As phishing schemes continue to grow, there is a pressing need for more robust and intelligent detection systems [4].

The advent of deep learning (DL) has significantly advanced the development of more effective phishing detection methods. Their ability to learn hierarchical representations of data provides substantial improvements over traditional machine-learning techniques [5], [6]. These models can autonomously extract features from raw data, significantly reducing the dependence on manual feature engineering and the detection system to adapt to new and evolving phishing tactics [7].

From the phishing detection perspective, the proposed framework leverages CNN and BiGRU to form a robust hybrid model. CNNs are renowned for their ability to extract spatial features from data automatically. They identify patterns and structures within images and text, making them ideal for analyzing website elements such as URLs, HTML content, and metadata[8], [9]. Utilizing CNNs, the framework can effectively capture the intricate details that differentiate phishing websites from genuine ones.

Complementing the CNNs, BiGRUs are a Recurrent Neural Network (RNN) type that excels in processing sequential data. They are particularly adept at recognizing temporal dependencies and patterns within sequences. BiGRUs, with their bidirectional processing capability, can examine data in both forward and reverse directions, offering a thorough comprehension of the temporal relationships within the input data [10], [11]. This bidirectional analysis is crucial for detecting phishing websites, as the order and context of elements on a web page can reveal malicious intent.

Integrating CNNs and BiGRUs in a hybridized framework enhances the detection system's ability to identify phishing websites. The CNN component extracts detailed spatial features from the website data. In contrast, the BiGRU component analyzes the temporal sequences of these features to detect patterns indicative of phishing. This combined approach enables the system to capture both spatial and temporal characteristics of phishing websites, improving detection accuracy and robustness.

Conventional phishing detection systems, which frequently depend on static features or simple heuristics, fall short in the face of increasingly sophisticated and adaptive phishing strategies. These traditional methods struggle to identify newly created phishing sites and adapt to evolving tactics. This necessitates a more dynamic and comprehensive approach to enhance detection capabilities. The advent of deep learning has significantly advanced phishing detection, offering models that can learn hierarchical representations of data and adapt to new phishing methods [12]. This study aims to create a hybrid solid framework that utilizes the advantages of CNN and BiGRU to enhance performance and improve the reliability of the detection of phishing websites.

2. Related Works

The literature presents numerous machine learning and deep learning frameworks for detecting phishing attacks (Table 1). Subba [13] developed a security framework utilizing a diverse stacking ensemble approach, incorporating three base classifiers and a meta-classifier. The model processes 44 extracted features from URLs and web pages, combining the results for the final prediction. The framework demonstrated high accuracy (99% for binary, 98% for multiclass) on benchmark datasets. Tenis & Santhosh [14] presented a real-time phishing detection system using a deep learning approach, including whitelisting and blacklisting mechanisms. The adaptive RNN (a− RNN) model showed a superior accuracy of 99.18% across different datasets.

Alsharaiah et al [15] proposed a novel framework integrating random forest classifiers with k-means clustering (RM-KmC) to improve feature correlation detection. Tested on a 5,000-sample dataset, the model achieved an accuracy of 98.64% with solid precision and recall metrics. Tang & Mahmoud [16] introduced a browser plug-in-based deep learning framework for real-time phishing detection, achieving 99.18% accuracy with the RNN-GRU model through a blend of whitelist and blacklist filtering.

Liu et al. [17] proposed a multistage detection model using the CASE framework, which exhibited high efficiency and performance and low false alarms in extensive evaluations. Kumar & Subba [18] introduced a lightweight framework for phishing detection, analyzing URLs to extract key features, resulting in high precision and minimal false positives. Similarly, Zeng et al. [11] introduced PhishBench 2.0, a robust benchmarking platform for phishing detection systems with extensive features, classifiers, and metrics. It is set to be released on GitHub for community use.

Rendall et al. [19] worked on a multi-layered detection framework that classifies phishing domains multiple times, achieving performance on par with leading detection systems. Sadique et al. [20] presented a real-time phishing URL detection framework that achieved 87% accuracy, suggesting incremental learning techniques to improve detection effectiveness.

Gowda et al. [21] presented a browser-embedded anti-phishing system using a rule-extraction method paired with Random Forest Classification, reaching 99.36% accuracy for real-time phishing detection. Saravanan & Subramanian [22] developed a framework for phishing detection that effectively extracts and selects features from websites, enhancing classification accuracy and outperforming current methods in experimental evaluations.

Elnagar & Thomas [23] introduced a cognitive detection framework combining BLSTM-RNN and CNN models, incorporating image recognition to enhance the identification of phishing websites. Rao & Pais [24] discussed a machine learning-based framework employing heuristic features from URLs and source code, achieving 99.31% accuracy of the Random Forest algorithm for phishing detection.

Cuzzocrea et al. [25] suggested a decision tree-based machine learning framework identifying and evaluating phishing assaults, exhibiting good performance in experimental evaluations.

A PhishMon framework based on machine learning was created utilizing fifteen unique features, achieving 95.4% accuracy in detecting phishing sites with a low false positive rate of 1.30% [26]. Yi et al. [27] examined a Deep Belief Network (DBN) based framework for phishing detection, achieving a 90% true positive rate and minimizing false positives to 0.6%. Park et al. [28] developed Phishing-Detective. This framework uses web scraping and data mining to

detect phishing websites through heuristic analysis, though its performance may be affected by changing phishing strategies.

Table 1: Summary of the Literature Review

Author & Year	Methodology/Algorithm	Results	Strengths	Limitations
Subba [13]	Heterogeneous stacking ensemble; 3 base classifiers, 1 meta-classifier (FCNN)	99.00% accuracy (binary), 98.00% accuracy (multiclass)	High accuracy, comprehensive feature extraction	Increased computational complexity
Tenis & Santhosh [14]	adaptive Recurrent Neural Networks (a-RNN)	99.18% accuracy	Reduced false positives, high detection accuracy	Complexity in real-time implementation
Alsharaiah et al. [15]	Random Forest integrated with k-means clustering(RM-KmC)	98.64% accuracy, high precision and recall	Enhanced feature correlation detection	It may require significant computational resources
Tang & Mahmoud [16]	Deep learning with whitelist/blacklist filtering; RNN-GRU model	99.18% accuracy	Effective real-time detection	Reliance on blacklisting could miss new phishing sites
Liu et al. [17]	Multistage detection model; CASE framework	High efficiency, low false alarms	Short execution times	It may not handle all phishing attack types
Kumar & Subba [18]	Lightweight machine learning; URL feature extraction	High precision, low false positive rate	Efficient with minimal resources	May miss more sophisticated phishing attacks
Sadique et al. [20]	Real-time phishing URL detection	87.00% accuracy	Real-time capability suggests incremental learning	Moderate accuracy; further improvement is needed
Zhao et al. [11]	PhishBench 2.0: benchmarking framework for phishing detection	Offers over 250 features, 12 classifiers, 17 metrics	Comprehensive feature and classifier set	High complexity for users unfamiliar with the framework
Gowda et al. [21]	Browser-embedded system; rule-extraction; Random Forest	99.36% accuracy in real-time detection	Real-time phishing detection	Potential browser compatibility issues
Rendall et al. [19]	Multi-layered detection; supervised machine learning	Comparable to state-of-the-art systems	Multi-tiered classification improves accuracy	Added complexity in classification processes
Saravanan & Subramanian [22]	Feature selection and extraction; phishing detection module	Outperformed existing classifiers in experimental tests	Efficient feature selection enhances detection accuracy	Potential generalization issues with unseen d
Elnagar & Thomas [23]	BLSTM-RNN and CNN with image recognition	Enhanced phishing detection	Cognitive approach with dual models	Computationally intensive
Rao & Pais [24]	Feature-based machine learning; Random Forest	99.31% accuracy	Effective heuristic feature extraction	Dependence on third-party services
Cuzzocrea et al. [25]	Decision tree-based machine learning	High accuracy in detecting phishing attacks	Simple and effective machine learning approach	May struggle with very dynamic phishing methods
Niakanlahiji et al. [26]	PhishMon: machine learning with 15 novel features	95.40% accuracy, 1.30% false positive rate	Low false positives, novel feature set	Potentially complex implementation

Yi et al. [27]	Deep Belief Networks (DBN)	90.00% true positive rate, 0.60% false positive rate	High accuracy in identifying phishing sites	Limited testing environment
Park et al. [28]	Web scraping and data mining; heuristic analysis	Effective in detecting phishing websites	Dynamic approach	May be impacted by evolving phishing tactics

3. Proposed Framework

3.1 Overview

The proposed hybrid framework integrates CNN and BiGRU networks to leverage their complementary strengths. CNNs are employed for feature extraction, and BiGRUs are used for sequence modeling, resulting in a robust detection system. The framework consists of four (4) states, namely input, preprocessing, deep learning, and detection stage, each having distinct features tailored towards the same purpose. The following is a thorough explanation of the stages depicted in Figure 1.

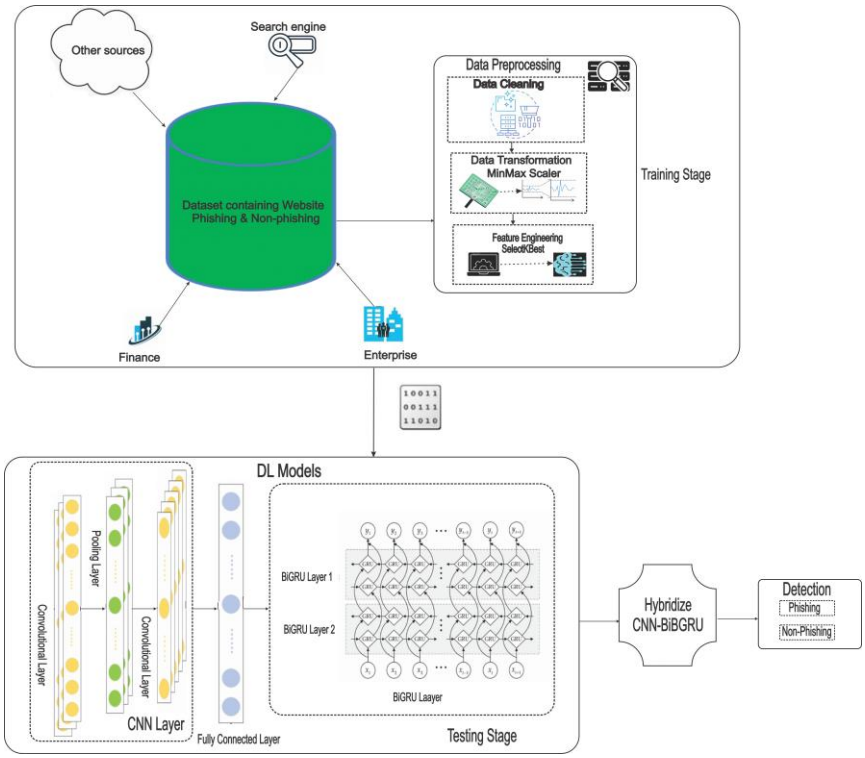

Figure 1: Hybridized CNN-BiGRU Framework for the Detection of Website Phishing Attack

In the first stage of this framework, the data input comprises phishing websites from different angles, such as enterprise, finance, search engine, and other sources, which will be fetched in the convolutional layer. This study employed datasets collected from IEEE Data Port because they are publicly available and contain information on phishing attacks reported by users, the largest data science community in the world, offering strong tools and resources to support researchers in achieving their data science objectives. A community of security experts verifies it, and it has been widely used by various authors in their research[30][31] [32][33][34].

Data Preprocessing Stage: During the data preprocessing stage, three crucial phases were carried out: data cleaning, transformation, and feature engineering. Data cleaning, an indispensable aspect of preprocessing, involved identifying and rectifying inconsistencies, errors, and irrelevant data within the dataset to enhance its quality and prepare it for utilization in deep learning models. Following data cleaning, a Min-max scaler was applied to transform the data, aiming to improve its compatibility with deep learning algorithms, preserve the original distribution's shape, and mitigate the influence of feature scales on the optimization process. Subsequently, the dataset was divided into two groups, with 20% set aside for testing and 80% for training and validation. Using Equation 1 below, the values were converted to the testing set and scaled between 0 and 1 using the min-max scaler fitted to the training set [35][36].

$$MinMaxSclaer\ (v'i) = \frac{x_i - min_A}{max_A - min_A}\ (new\_max_A - new\_min_A) + new\_min_A \qquad (1)$$

Where x_i represents the ith value, max_A and min_A Denote a feature's maximum and minimum values and $new\_.max_A$ and new_min_A are the values 0 and 1, respectively.
Feature engineering ensued, incorporating a feature selection technique known as SelectKBest to enhance model performance, generalization, and computational efficiency and mitigate the impact of irrelevant or noisy features. The score was determined utilizing equation 2, as proposed by [37][38].

$$x^2 = \sum_{i=1}^{n} \frac{(OFi - EFi)}{EFi} \quad (2)$$

Where OFi is the frequency observed for the feature $F's\ i$ −th value, and EFi is the frequency anticipated for feature $F's\ i$ −th value. After the features were scaled and selected, CNN and BiGru algorithms were also used to extract data, the CNN component extracted spatial features while Bigru temporal across different dataset segments which were then fused to form the feature vector that was passed to the fully connected layer.
The dataset's description and correlation heatmap are presented in Table 2 and Figure 2, respectively.

Table 2: Distribution of the Datasets

Dataset Source	Phishing attack type	Total number of features	Non-phishing	Phishing	Total
IEEE Data Port	Website URLs	80	7781	7586	15367
Mendeley	Website URLs	55	100,945	134,850	235,795

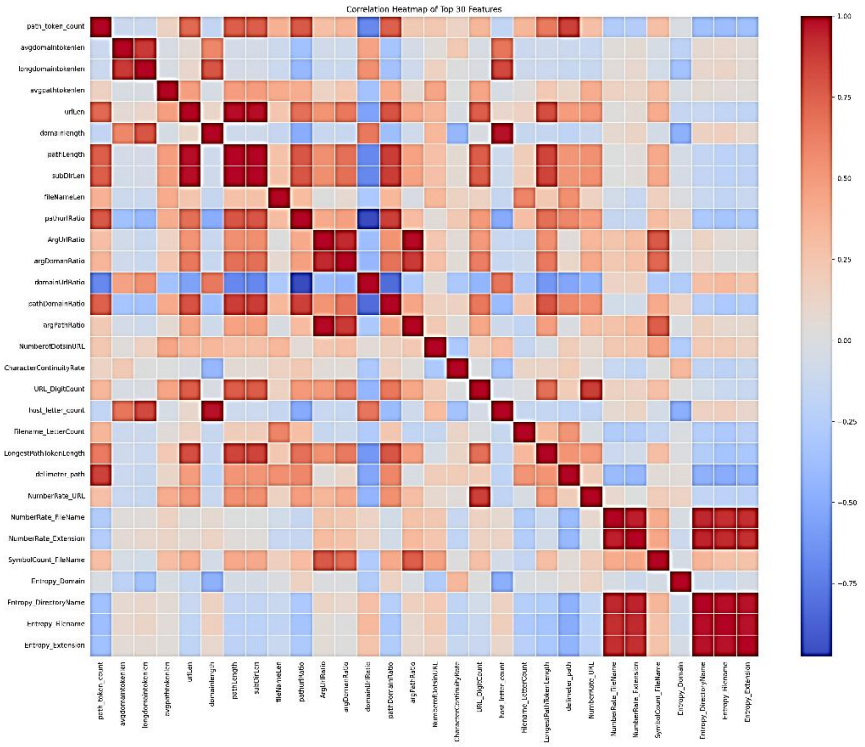

Figure 2: The Correlation Heatmap Dataset Employed

DL Stage: A subset of machine learning has garnered significant attention in recent years due to advancements in processing power and expanded data storage capacities. These developments have greatly facilitated the application of DL methodologies, which have demonstrated remarkable efficacy across various domains, including image processing, natural language processing, and machine translation, particularly when handling large datasets. Leveraging these advantages, our study adopts two prominent DL algorithms: CNN and BiGRU. The selection of these algorithms is based on the belief that combining different approaches enhances overall accuracy, as demonstrated by [39], [40].
Also, based on the review of the existing literature and to the best of our knowledge, no author has combined these algorithms for the detection of phishing attack. Within this framework, the CNN component is tasked with extracting

high-level features from the input dataset. It is adept at capturing local patterns and features. Subsequently, the BiGRU component sequentially processes these features, considering sequential dependencies and temporal features across different dataset segments. The integration of a fully connected layer atop the BiGRU facilitates final detection. This framework is poised to enhance the accuracy and effectiveness of phishing attack classification by harnessing the complementary strengths of the CNN-BiGRU architectures.

Let $X = \{x_1, x_2, \dots, x_n, \}$ represent the input sequence, where $x_i,$ is the word embedding of the $i - th$ feature in the sequence

$$z_i^l = f\left(\sum_{j=1}^{m} w_j^l . x_{i+j-1} + b^l\right) \quad (3)$$

Where z_i^l is the output of the $l - th$ convolutional filter at position i, w_j^l are the filter weights, b^l is the bias term, and f is the activation function ReLU.

$$p^l = max\left(z_1^l, z_2^l, \dots, z_{n-m+1}^l\right) \quad (4)$$

where $p^l$is the output of the max-pooling layer for the $l - th$ filter

$$\vec{h}_t = GRU(x_t, \vec{h}_{t-1}) \quad (5)$$

$$\overleftarrow{h}_t = GRU(x_t, \overleftarrow{h}_{t+1}) \quad (6)$$

$$h_t = [\vec{h}_t \; ; \; \overleftarrow{h}_t] \quad (7)$$

Where $[.;.]$ denotes concatenation

$$h = h^1, h^2, \dots, h^L \quad (8)$$

where h^l is the output feature vector from the $l - th$ convolutional filter or BiGRU layer.

$$z = hW + b \quad (9)$$

Where W is the weight matrix and b is the bias vector.

$$\hat{y} = softmaz(z) \quad (10)$$

Where $\hat{y}$ is the predicted probability distribution over the classes.

4. Experimental Setup

To evaluate the suggested hybrid model, a simulation is run on a desktop computer running Windows 11 Pro with a 64-bit operating system on an Intel(R) Core(TM) i5-6300U CPU running at 2.40GHz and 8GB of RAM. The notebook is a Jupiter Notebook 6.4.8. The parameters used for the study are shown in Table 3.

Table 3: Parameters

Parameters	Values
Activation Function	Relu
Epochs	20
Batch size	32
Optimizer	Adam
Dropout	0.2
Loss	Binary Cross Entropy

4.1 Dataset

A publicly accessible phishing dataset with features taken from both legitimate and phishing websites was used for the studies. The dataset used for evaluating the model, sourced from IEEE Data Port and Mendeley, consists of 15,367 instances, with 7,781 non-phishing and 7,586 phishing, 235,795 entries, with 134,850 legitimate URLs and 100,945 phishing URLs samples, respectively, which is presented in Table 2. The 80 extracted features include URL attributes, HTML content, and metadata critical for the CNN-BiGRU architecture, where CNN captures spatial features and BiGRU models temporal patterns.

4.2 Evaluation Metrics

The framework's performance was appraised using recall, accuracy, F1-score, and precision. These metrics offer an ample assessment of the model's efficiency in distinguishing between phishing and legitimate websites [1] [41] these metrics are represented in the following equation.

$$Precision = \frac{TP}{TP + FP} = + \frac{True\ Positive}{Total\ Predicted\ Positive} \quad (11)$$

$$Recall = \frac{TP}{TP + FN} = + \frac{True\ Positive}{Total\ Predicted\ Positive} \quad (12)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} = + \frac{True\ Positive}{Total\ Predicted\ Positive} \quad (13)$$

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} = + \frac{True\ Positive}{Total\ Predicted\ Positive} \quad (14)$$

$$Specificity = \frac{True\ Negatives\ (TN)}{True\ Negatives\ (TN) + False\ Positive\ (FP)} \quad (15)$$

5. Results and Discussion

The proposed development of a hybridized CNN-BiGRU framework for detecting website phishing attacks was evaluated through an experiment with the Python programming language using the Jupyter Notebook tool, a cooperative web application for producing and sharing documents that combine Code, Rich Text, and Visualizations. This was chosen because of its Ease of Use, Flexibility, and Reproducibility. The experimentation of the model was performed with a baseline dataset using train-test split and cross-validation methods. Also, to ensure model generalization, an independent dataset was used to test and train the framework's performance. To produce a robust hybridized framework, 80% of the datasets were used for training and 20% for testing in each dataset. Also, SelectKbest was used to select the best features passed into the model for training and testing. Figure 3 depicts the performance of the model using a train-test split and cross-validation on the baseline dataset.

The performance of the hybridized CNN-BiGRU framework for detecting website phishing attacks exhibits remarkable effectiveness when evaluated using two methods: Train-Test Split and Cross-validation. The methods yield exceptionally high scores, each exceeding 99.9%. Cross-validation stands out for its consistently superior and balanced results across all metrics, including accuracy, precision, F1 score, and specificity. Although the Train-Test Split achieves a perfect recall of 100%, the framework ensures all phishing attempts are detected, leaving no malicious sites undetected. These metrics underscore the model's reliability and efficacy in practical cybersecurity applications. Figure 4 shows the confusion matrices using the two approaches, and Figure 5 shows the performance of the standalone dataset.

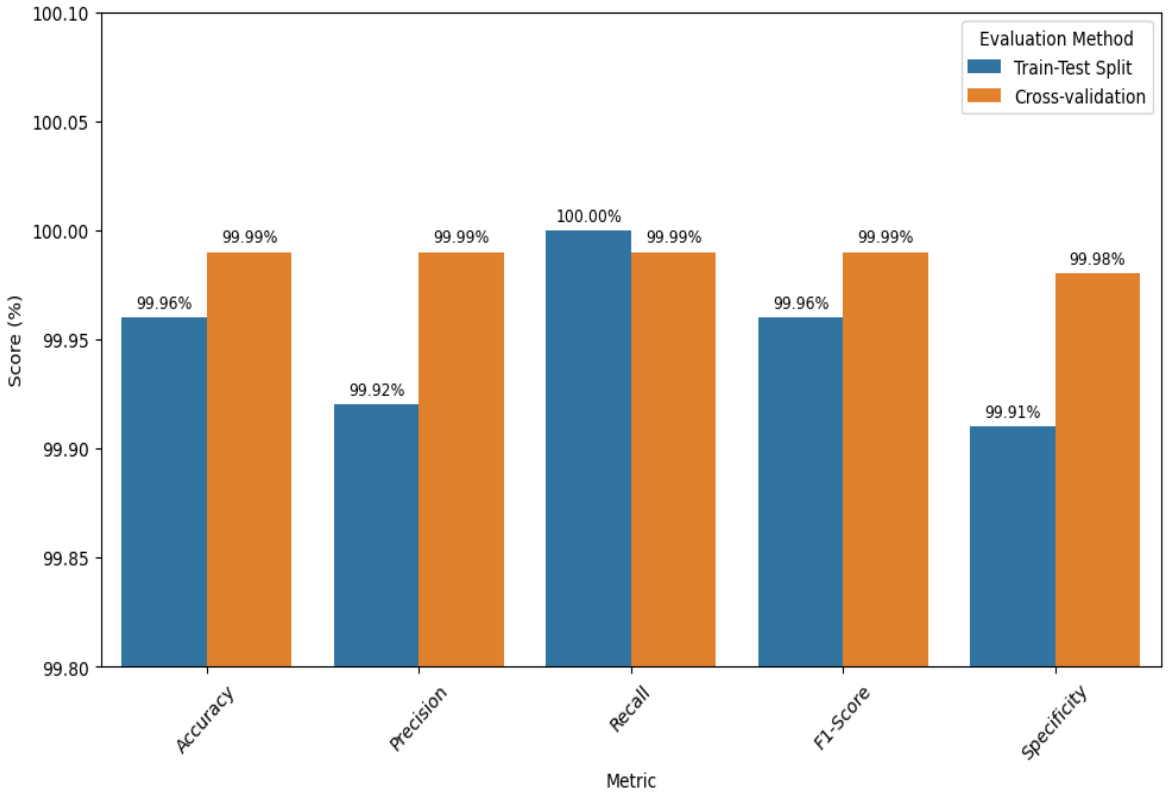

Figure 3: Visualization of the Performance of the Proposed CNN-BiGRU Framework using Train-Test Split and Cross-validation

5.2 Comparative Analysis of this Study's Accuracy with Existing Study Accuracy, Precision, Recall and F1-score

Comparison of various studies' accuracy, precision, recall, and F1-score with the current research. These studies were analyzed comparatively, focusing on the trends and overall progress presented in Tables 4 and 5, respectively.

Table 4 compares the accuracy rates of several studies on phishing detection systems, revealing a consistent upward trajectory in performance over time. Early research, such as studies by Yi et al. [27] and Sadique et al. [20], reported lower accuracies of 90.00% and 87.00%, highlighting the initial challenges in effectively identifying phishing threats.

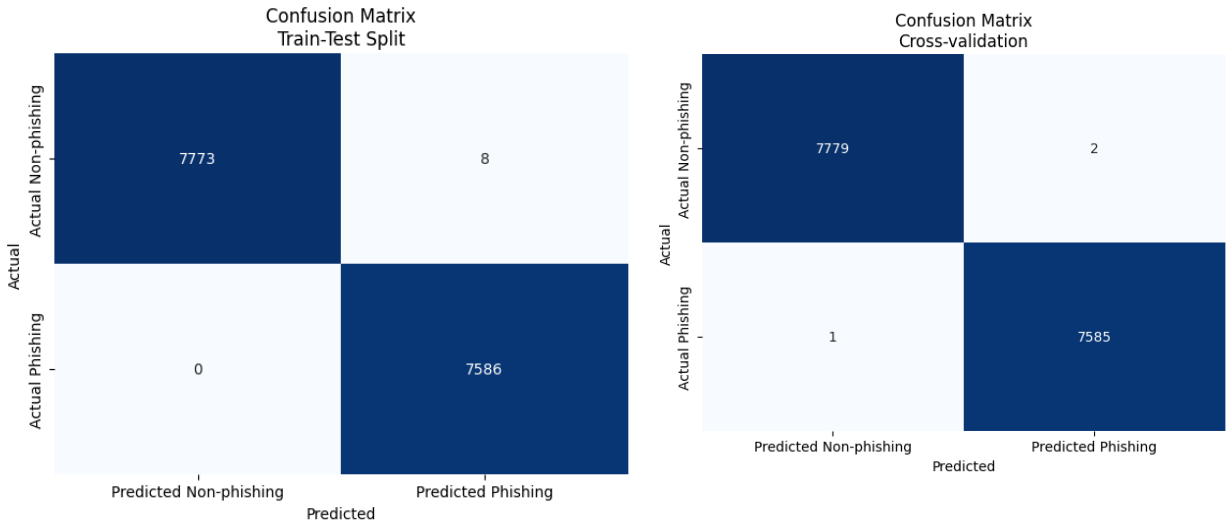

Figure 4: Confusion Matrices of the proposed CNN-BiGRU Framework

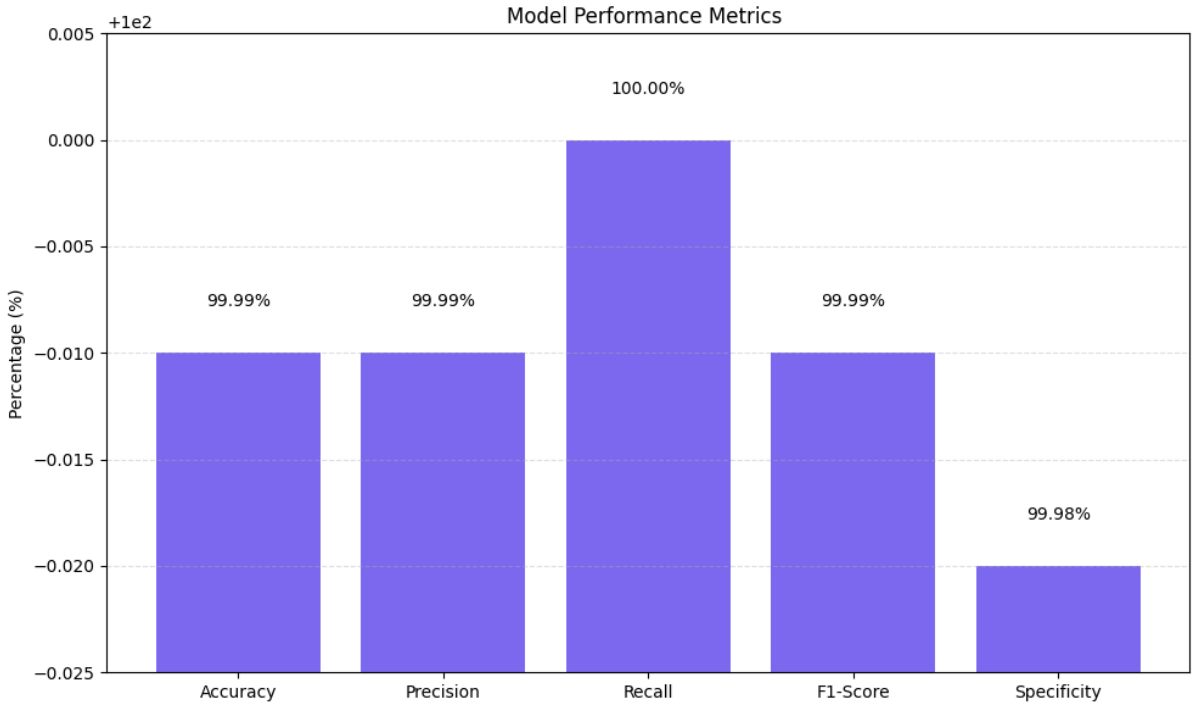

Figure 5: Performance of the Standalone Dataset

As detection techniques advanced, studies like Niakanlahiji et al. [26] and Alsharaiah et al. [15] demonstrated notable improvements, achieving accuracies of 95.40% and 98.64%, respectively. More recent studies—including Rao & Pais [24], Gowda et al. [21], Tang & Mahmoud [16], Tenis & Santhosh [14], and Subba [13] consistently surpassed the 99.00% mark, showcasing significant progress in detection efficiency. The current study achieves an exceptional accuracy of 99.96%, setting a new benchmark and underscoring the continued refinement and effectiveness of modern phishing detection approaches.

Table 4: Comparison of the Accuracy of Various Studies with the Current Research

Studies	Accuracy (%)
Yi et al. [27]	90.00
Niakanlahiji et al. [26]	95.40
Rao & Pais [24]	99.31
Gowda et al. [21]	99.36
Sadique et al. [20]	87.00
Tang & Mahmoud [16]	99.18
Alsharaiah et al. [15]	98.64
Tenis & Santhosh [14]	99.18
Subba [13]	99.00
This study	99.96

Table 5: Comparison of the Accuracy, Precision, Recall, and F1-score of various studies with the current research

Author (Year)	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
[21]	99.36	98.87	100.00	99.43
[17]	Nil	98.86	89.23	93.80
[14]	99.20	99.00	99.00	90.00
[15]	98.64	98.60	98.70	98.60
[13]	98.80	99.20	99.10	99.10
This Study (CNN-BiGRU)	99.96	99.92	100.00	99.92

Table 5 compares studies focused on phishing detection frameworks, evaluating their accuracy, precision, recall, and F1 score. The current research, employing a CNN-BiGRU model, achieves the highest accuracy at 99.96%, near-perfect precision F1-scores of 99.92%, and a recall of 100.00%. This highlights a significant advancement over prior studies, showcasing a highly effective model in accurately identifying phishing attacks while minimizing false positives.
Other notable studies, such as Gowda et al. [21] and Subba [13], also demonstrated strong performances, with accuracies exceeding 99.00% and well-balanced metrics. However, researchers like Liu et al. [17] displayed lower precision and F1 scores, indicating potential compromises in their detection approaches. Overall, while each study contributes valuable insights to the field, the current research sets a new benchmark with its superior results, as shown in Table 6.
Table 6 offers a comparative evaluation of this study alongside various deep learning models applied to phishing detection, highlighting notable advancements across core performance metrics. The a-RNN model from Gowda et al. [21] and the RFk-mC model from Alsharaiah et al. [15] achieved accuracies of 99.20% and 98.64%, respectively, with slightly lower precision, recall, and F1-scores. The LSTM-CNN model from Alshingiti et al. [42] recorded a lower accuracy of 97.60%, while the CNN-BiLSTM approach from Zhang et al. [43] showed stronger results with 98.84% accuracy and an impressive precision of 99.71%. In contrast, the CNN-BiGRU framework proposed in this study outperforms all previous deep learning frameworks, achieving outstanding results with 99.96% accuracy, 99.92% precision, perfect recall at 100.00%, and a 99.92% F1-score demonstrating its clear superiority and effectiveness in phishing detection.

Table 6: Comparison of this Study with other Deep Learning Architectures

Author&Year	Adopted Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
[21]	a-RNN	99.20	99.00	99.00	90.00
[15]	RFk-mC	98.64	98.60	98.70	98.60
[42]	LSTM-CNN	97.60	96.90	98.20	97.60
[43]	CNN-BiLSTM	98.84	99.71	98.04	98.87
This Study	CNN-BiGRU	99.96	99.92	100	99.92

6. Conclusion

This paper presents a novel deep learning architecture that integrates CNN and BiGRU to tackle the increasingly sophisticated landscape of phishing attacks. The CNN layer extracts meaningful spatial features, while the BiGRU layer effectively analyzes the sequential flow of website elements, enabling the model to capture both structural and contextual indicators of phishing. Rigorous preprocessing, including normalization and feature selection, ensures data quality and enhances learning efficiency. Experimental evaluations using large-scale phishing datasets confirm the model's exceptional performance, achieving 99.96% accuracy and 100% recall. These results significantly outperform existing frameworks, affirming the potential of the CNN-BiGRU architecture in practical cybersecurity applications. Furthermore, the error analysis reveals a consistent pattern in misclassified samples, suggesting possible data-related limitations such as ambiguous or underrepresented phishing patterns. Future work will explore expanding the feature set, statistical significance testing, and addressing biases through dataset augmentation and continuous learning to ensure adaptability against emerging phishing tactics.

Acknowledgments

The authors sincerely thank the Cyber Security Science and Security Science Departments at the Federal University of Technology, Minna, Nigeria, and the National Open University, Abuja, Nigeria, for their support, resources, and academic guidance throughout this research.

Funding: No funding was received for this research

Credit authorship contribution statement

Abdullahi Raji Egigogo: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing. **Ismaila Idris:** Methodology, Software, Validation, Formal analysis, Data curation, Writing – original draft, Writing – review & editing, Supervision. **Morufu Olalere:** Methodology, Software, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Supervision. **Abisoye Opeyemi Aderike** Abisoye**:** Methodology, Formal analysis, Data curation, Writing – original draft, Writing – review & editing, Supervision. **Joseph Adebayo Ojeniyi:** Methodology, Supervision, Project administration, Validation, Writing – review & editing, Formal analysis, Investigation.

References

[1] M. A. Ivanov, B. V. Kliuchnikova, I. V. Chugunkov, and A. M. Plaksina, "Phishing Attacks and Protection against Them," *Proceedings of the 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2021*, pp. 425–428, 2021, doi: 10.1109/ElConRus51938.2021.9396693.

[2] M. D. Abdulrahman, J. K. Alhassan, O. S. Adebayo, J. A. Ojeniyi, and M. Olalere, "Phishing Attack Detection Based on Random Forest with Wrapper Feature Selection Method," *International Journal of Information Processing and Communication (IJIPC*, vol. 7, no. 2, pp. 209–224, 2019, [Online]. Available: http://repository.futminna.edu.ng:8080/jspui/handle/123456789/3109

[3] A. Redi and N. Ernasari, "Efforts to Overcome Web-Based Phishing Crimes in the World of Cyber Crime," 2023, doi: 10.4108/eai.28-10-2023.2341807.

[4] M. Rivki, A. M. Bachtiar, T. Informatika, F. Teknik, and U. K. Indonesia, "Automated Phishing Detection using URLs and Webpages," no. 112, 2024, doi: https://doi.org/10.48550/arXiv.2408.01667.

[5] P. Lara-Benítez, M. Carranza-García, and J. C. Riquelme, "An Experimental Review on Deep Learning Architectures for Time Series Forecasting," *International Journal of Neural Systems*, vol. 31, no. 3, 2021, doi: 10.1142/S0129065721300011.

[6] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Computer Science Review*, vol. 40, 2021, doi: 10.1016/j.cosrev.2021.100379.

[7] M. A. Adebowale, K. T. Lwin, and M. A. Hossain, "Intelligent phishing detection scheme using deep learning algorithms," *Journal of Enterprise Information Management*, vol. 36, no. 3, pp. 747–766, 2023, doi: 10.1108/JEIM-01-2020-0036.

[8] W. Wei, Q. Ke, J. Nowak, M. Korytkowski, R. Scherer, and M. Woźniak, "Accurate and fast URL phishing detector: A convolutional neural network approach," *Computer Networks*, vol. 178, p. 107275, Sep. 2020, doi: 10.1016/J.COMNET.2020.107275.

[9] A. D. Kulkarni, "Convolution Neural Networks for Phishing Detection," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 4, pp. 15–19, 2023, doi: 10.14569/IJACSA.2023.0140403.

[10] P. Khandelwal, J. Konar, and B. Brahma, "Training RNN and its Variants Using Sliding Window Technique," *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science, SCEECS 2020*, 2020, doi: 10.1109/SCEECS48394.2020.93.

[11] H. Zhao, Z. Lai, H. Leung, and X. Zhang, "Neural-Network-Based Feature Learning: Recurrent Neural Network," pp. 253–275, 2020, doi: 10.1007/978-3-030-40794-0_12.

[12] S. Sawant, R. Savakhande, O. Sankhe, and S. Tamboli, "Phishing Detection by integrating Machine Learning and Deep Learning," in *Proceedings of the 18th INDIAcom; 2024 11th International Conference on Computing for Sustainable Global Development, INDIACom 2024*, 2024, pp. 1078–1083. doi: 10.23919/INDIACom61295.2024.10499100.

[13] B. Subba, "A heterogeneous stacking ensemble-based security framework for detecting phishing attacks," *2023 National Conference on Communications, NCC 2023*, 2023, doi: 10.1109/NCC56989.2023.10068026.

[14] A. Tenis and R. Santhosh, "Modelling of an Adaptive Network Model for Phishing Website Detection Using Learning Approaches," *Fusion: Practice and Applications*, vol. 12, no. 2, pp. 159–171, 2023, doi: 10.54216/FPA.120213.

[15] M. A. Alsharaiah, A. A Abu-Shareha, M. Abualhaj, L. H. Baniata, O. Adwan, A. Al-saaidah and M. Oraiqat, "A new phishing-website detection framework using ensemble classification and clustering," *International Journal of Data and Network Science*, vol. 7, no. 2, pp. 857–864, 2023, doi: 10.5267/j.ijdns.2023.1.003.

[16] L. Tang and Q. H. Mahmoud, "A Deep Learning-Based Framework for Phishing Website Detection," *IEEE Access*, vol. 10, pp. 1509–1521, 2022, doi: 10.1109/ACCESS.2021.3137636.

[17] D.-J. Liu, G.-G. Geng, X.-B. Jin, and W. Wang, "An efficient multistage phishing website detection model based on the CASE feature framework: Aiming at the real web environment," *Computers and Security*, vol. 110, 2021, doi: 10.1016/j.cose.2021.102421.

[18] Y. Kumar and B. Subba, "A lightweight machine learning based security framework for detecting phishing attacks," in *2021 International Conference on COMmunication Systems & Networks (COMSNETS)*, 2021, pp. 184–188. doi: 10.1109/COMSNETS51098.2021.9352828.

[19] K. Rendall, A. Nisioti, and A. Mylonas, "Towards a Multi-Layered Phishing Detection," *SENSORS*, vol. 20, no. 16, 2020, doi: 10.3390/s20164540.

[20] F. Sadique, R. Kaul, S. Badsha, and S. Sengupta, "An Automated Framework for Real-time Phishing URL Detection," in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 335–341. doi: 10.1109/CCWC47524.2020.9031269.

[21] M. H. R. Gowda, M. Adithya V, G. S. Prasad, and S. Vinay, "Development of anti-phishing browser based on random forest and rule of extraction framework," *CYBERSECURITY*, vol. 3, no. 1, 2020, doi: 10.1186/s42400-020-00059-1.

[22] P. Saravanan and S. Subramanian, "A Framework for Detecting Phishing Websites using GA-based Feature Selection and ARTMAP-based Website Classification," *Procedia Computer Science*, vol. 171, pp. 1083–1092, 2020, doi: 10.1016/j.procs.2020.04.116.

[23] S. Elnagar and M. A. Thomas, "A Cognitive Framework for Detecting Phishing Websites," no. March, 2019.

[24] R. S. Rao and A. R. Pais, "Detection of phishing websites using an efficient feature-based machine learning framework," *NEURAL COMPUTING & APPLICATIONS*, vol. 31, no. 8, pp. 3851–3873, 2019, doi: 10.1007/s00521-017-3305-0.

[25] A. Cuzzocrea, F. Martinelli, and F. Mercaldo, "A machine-learning framework for supporting intelligent web-phishing detection and analysis," *ACM International Conference Proceeding Series*, 2019, doi: 10.1145/3331076.3331087.

[26] A. Niakanlahiji, B. T. Chu, and E. Al-Shaer, "PhishMon: A machine learning framework for detecting phishing webpages," *2018 IEEE International Conference on Intelligence and Security Informatics, ISI 2018*, pp. 220–225, 2018, doi: 10.1109/ISI.2018.8587410.

[27] P. Yi, Y. Guan, F. Zou, Y. Yao, W. Wang, and T. Zhu, "Web phishing detection using a deep learning framework," *Wireless Communications and Mobile Computing*, vol. 2018, 2018, doi: 10.1155/2018/4678746.

[28] A. J. Park, R. N. Quadari, and H. H. Tsang, "Phishing website detection framework through web scraping and data mining," *2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2017*, pp. 680–684, 2017, doi: 10.1109/IEMCON.2017.8117212.

[29] M. G. Hr, A. Mv, S. Gunesh Prasad, and S. Vinay, "Development of anti-phishing browser based on random forest and rule of extraction framework," *Cybersecurity*, vol. 3, no. 1, 2020, doi: 10.1186/s42400-020-00059-1.

[30] D. Liu, J.-H. Lee, W. Wang, and Y. Wang, "Malicious Websites Detection via CNN-based Screenshot Recognition," in *2019 International Conference on Intelligent Computing and its Emerging Applications (ICEA)*, 2019, pp. 115–119. doi: 10.1109/ICEA.2019.8858300.

[31] W. Wang, F. Zhang, X. Luo, and S. Zhang, "PDRCNN: Precise Phishing Detection with Recurrent Convolutional Neural Networks," *Security and Communication Networks*, vol. 2019, 2019, doi: 10.1155/2019/2595794.

[32] N. Q. Do, A. Selamat, O. Krejcar, T. Yokoi, and H. Fujita, "Phishing webpage classification via deep learning-based algorithms: An empirical study," *Applied Sciences (Switzerland)*, vol. 11, no. 19, 2021, doi: 10.3390/app11199210.

[33] S. Srinivasan and D. P, "Enhancing the security in cyber-world by detecting the botnets using ensemble classification-based machine learning," *Measurement: Sensors*, vol. 25, 2023, doi: 10.1016/j.measen.2022.100624.

[34] S. A. Sajidha, "ISCX-URL-2016," *IEEE Dataport.*, 2023, doi: https://dx.doi.org/10.21227/xngk-3p42.

[35] C. Y. Chou, D. Y. Hsu, and C. H. Chou, "Predicting the Onset of Diabetes with Machine Learning Methods," *Journal of Personalized Medicine*, vol. 13, no. 3, 2023, doi: 10.3390/jpm13030406.

[36] J. Huang, Z., Li, Y., Peng, H., & Wu, "An ensemble learning approach for predicting the grade of brain glioma using MRI images," in *Biomedical Signal Processing and Control*, 2021. doi: . https://doi.org/10.1016/j.bspc.2020.102529.

[37] G. A. Sharifai and Z. Zainol, "Feature selection for high-dimensional and imbalanced biomedical data based on robust correlation based redundancy and binary grasshopper optimization algorithm," *Genes*, vol. 11, no. 7, pp. 1–26, 2020, doi: 10.3390/genes11070717.

[38] N. N. Thuy and S. Wongthanavasu, "A Novel Feature Selection Method for High-Dimensional Mixed Decision Tables," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 7, pp. 3024–3037, 2022, doi: 10.1109/TNNLS.2020.3048080.

[39] B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, "Defending against phishing attacks: taxonomy of methods, current issues and future directions," *Telecommunication Systems*, vol. 67, no. 2, pp. 247–267, Feb. 2018, doi: 10.1007/S11235-017-0334-Z.

[40] N. Q. Do, A. Selamat, O. Krejcar, T. Yokoi, and H. Fujita, "Phishing webpage classification via deep learning-based algorithms: An empirical study," *Applied Sciences (Switzerland)*, vol. 11, no. 19, 2021, doi: 10.3390/app11199210.

[41] S. O. S. H. A. W. J. O. I Idris, "Stack Ensemble Model For Detection Of Phishing Website," *IEEE International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*, pp. 1–6, 2024.

[42] Z. Alshingiti, R. Alaqel, J. Al-Muhtadi, Q. E. U. Haq, K. Saleem, and M. H. Faheem, "A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN," *Electronics (Switzerland)*, vol. 12, no. 1, 2023, doi: 10.3390/electronics12010232.

[43] Q. Zhang, Y. Bu, B. Chen, S. Zhang, and X. Lu, "Research on phishing webpage detection technology based on CNN-BiLSTM algorithm," *Journal of Physics: Conference Series*, vol. 1738, no. 1, 2021, doi: 10.1088/1742-6596/1738/1/012131.