Journal of Future Artificial Intelligence and Technologies

E-ISSN: 3048-3719

Research Article

BERTPHIURL: A Teacher-Student Learning Approach Using DistilRoBERTa and RoBERTa for Detecting Phishing Cyber URLs

Payman Hussein Hussan * and Syefy Mohammed Mangj

Department of Computer Networks and Software Techniques, Babylon Technical Institute, Al-Furat Al-Awsat Technical University, Kufa, Iraq; email: inb.beman10@atu.edu.iq; inb.syffi10@atu.edu.iq

* Corresponding Author : Payman Hussein Hussan

Abstract Phishing is a fraudulent activity wherein an attacker impersonates a trusted individual or organization to acquire sensitive information from an online user. Phishing websites have become a major cybersecurity issue in the contemporary digital landscape. As online activities expand in e-commerce, banking, and social media, the hazards presented by these fraudulent websites have intensified. Deep learning-based Natural Language Processing (NLP) approaches offer an effective solution for detecting phishing URLs. However, deploying large models like BERT or RoBERTa for real-time detection poses computational challenges. This study proposes BERTPHIURL, a Teacher-Student Learning framework that leverages RoBERTa as the Teacher model and DistilRoBERTa as the Student model to improve phishing detection efficiency while reducing computational overhead. By applying knowledge distillation, the Student model learns from the Teacher, preserving high detection accuracy with significantly lower resource consumption. The proposed approach effectively captures contextual relevance and local features in malicious URL detection tasks. The experiments were conducted on a dataset exceeding 50,000 URLs to evaluate performance. Results indicate that BERTPHIURL achieves a 94.22% accuracy, outperforming existing phishing detection methods while maintaining efficiency suitable for real-time applications.

Keywords: BERT; DistilRoBERTa; Natural Language Processing; NLP phishing detection; Phishing URL Detection; RoBERTa; Teacher-Student learning; Transformer.

1. Introduction

Phishing is a deceptive and cyber-attack aimed at collecting sensitive information by impersonating legitimate websites. Phishers deceive individuals by imitating authentic content to extract personal information, including social security numbers, credit card details, and passwords. Phishing attacks manifest in several ways and typically utilize multiple communication channels, including email, instant messaging, and social media[1]. Malefactors frequently impersonate reputable banks, credit card companies, or prominent websites to coerce customers into accessing the phishing site and divulging information they may later regret. In the 2020 report by "the Internet Crime Complaint Centre", the FBI documented 791,790 allegations of suspected Internet crime, reflecting an increase of over Three hundred thousand relatives to 2019 statistics[2].

Phishing URL websites are a common linguistic manipulation strategy in phishing and text-based cyberattacks. They are a primary tool for attackers seeking to obtain users' sensitive information. The impact of these attacks extends beyond individual victims, potentially disrupting organizations, eroding public trust, and undermining societal stability. Social engineering tactics are crucial in phishing attacks, where counterfeit websites are created to resemble legitimate ones closely. These deceptive sites lure victims into submitting sensitive information before redirecting them to the authentic website[3]. Figure 1 illustrates the structured phases of a URL-based phishing assault, which progresses through deliberate planning

Received: February, 1st 2025
Revised: February, 20th 2025
Accepted: February, 23rd 2025
Published: February, 24th 2025
Curr. Ver.: February, 24th 2025

Copyright: © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY SA) license (https://creativecommons.org/licenses/by-sa/4.0/)

and exploitation. The process begins with the attacker crafting a fraudulent website that mimics a legitimate platform. The phishing URL is then strategically disseminated to unsuspecting users. Once users interact with the URL, they are directed to the counterfeit site, which is meticulously designed to appear genuine, increasing the likelihood of data compromise.

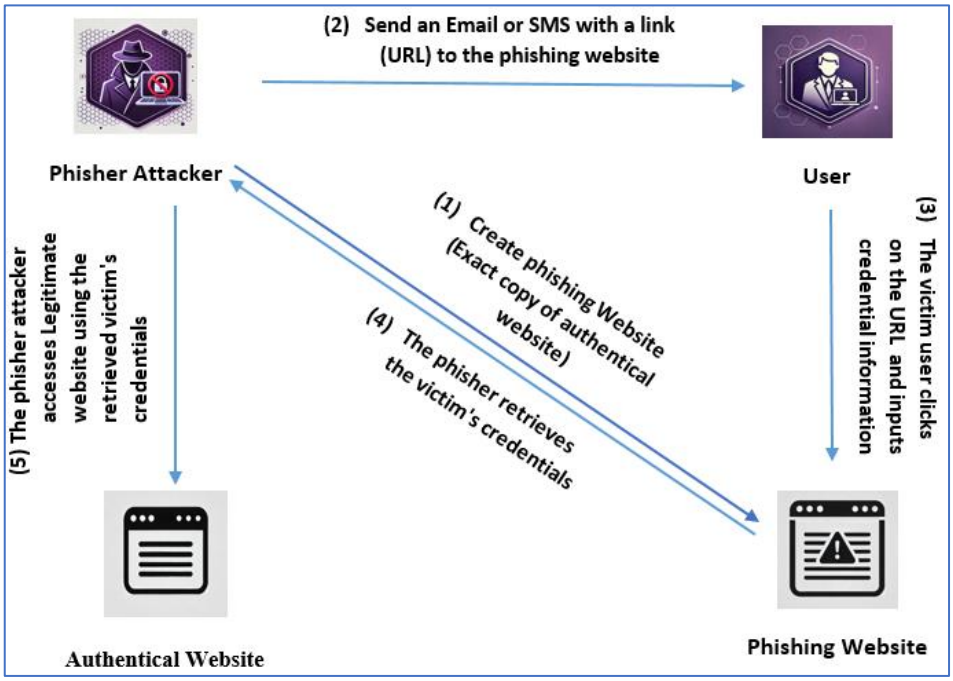

Figure 1. Phishing cyber-URL attack lifecycle

In this hypothetical scenario, the phishing attacker adeptly obtains the user's login credentials, thereby gaining unauthorized access to sensitive information. Utilizing the acquired login credentials, the perpetrator may get unauthorized access to legitimate websites, impersonating the actual user. This clandestine strategy allows the phisher to exploit the user's identity and potentially engage in illicit activities for personal gain, underscoring the gravity and intricacy of phishing attacks in the digital domain[4].

Given the significance of identifying phishing websites, it is essential to develop a powerful detector that filters and obstructs their activities on the Internet. Numerous an-ti-phishing techniques have been developed, including blacklist and whitelist methods, heuristic feature analysis, and visual similarity approaches. Contemporary browsers are designed to mitigate the risk of users becoming ensnared in malicious schemes, yet consumers remain susceptible to phishing attacks and disclose their confidential information[5].

As technology continues to grow, phishing techniques have commenced to progress expeditiously, and this has been averted by utilizing anti-phishing mechanisms to detect phishing. Some approach-based phishing detection methods use tabular features like WHOIS data, domain age, and page rank to classify websites[6]. However, some limitations are faced when analyzing URL text. It relies on external data sources, introduces high latency, and struggles to detect newly registered phishing domains without historical data. Additionally, feature extraction and normalization steps increase computational costs and can be rate-limited or blocked.

The efficacy of phishing and text-based cyber-attacks relies on psychological manipulation, linguistic complexity, and the ability to evade standard detection mechanisms. Understanding these inherent properties is crucial for developing effective countermeasures. Natural Language Processing (NLP) models provide a robust alternative, enabling phishing detection without relying on external data sources and reducing computational overhead. By analyzing linguistic patterns, grammatical structures, and contextual cues, NLP approaches can identify subtle indicators of malicious intent that traditional methods often miss[7]–[9]. These systems also leverage contextual understanding to distinguish between benign and harmful content, even when textual similarities are significant.

However, current phishing detection methodologies face challenges in precisely identifying phishing attempts, particularly against adversarial techniques such as URL shortening, dynamic URL modifications, and DNS manipulation. Many conventional approaches rely on manually engineered features, such as URL length and domain reputation, which sophisticated attackers can easily bypass. Additionally, deep learning-based models, while effective, often require high computational resources, making them less practical for real-time applications.

To overcome these limitations, this study adopts a Teacher-Student Learning approach, where a larger model (RoBERTa) serves as the Teacher to guide a more lightweight model (DistilRoBERTa) as the Student[10], [11]. RoBERTa, known for its robust contextual embedding capabilities, provides superior phishing detection accuracy. However, its computational demands limit its deployment in resource-constrained environments[12], [13]. By employing knowledge distillation[14], DistilRoBERTa learns from RoBERTa, retaining its high-level performance while significantly reducing computational complexity. This trade-off enables real-time phishing detection without compromising accuracy, making the approach highly suitable for cybersecurity applications.

Therefore, to develop a high-performance method that accurately and efficiently classifies websites as legitimate or phishing, this study addresses the following questions: What website attributes are commonly used to identify phishing attempts? What key indicators help detect illegitimate phishing websites? How does NLP contribute to phishing domain detection? What role does lexical analysis play in phishing URL identification? Which techniques can effectively detect malicious URLs? Which deep learning (DL) architecture is best suited for phishing detection, and why? This study addresses the aforementioned questions., which are significant for detecting malicious URLs. The main contributions that this paper brings out are:

1. Enhancing phishing webpage identification by developing a model that leverages URL-based features for improved detection accuracy.
2. Proposing BERTPHIURL, a Teacher-Student Learning architecture where RoBERTa (Teacher) transfers knowledge to DistilRoBERTa (Student) to improve phishing URL detection while reducing computational costs.
3. Utilizing DistilRoBERTa to derive word-based contextual representations, enabling efficient low-level textual feature extraction for detecting phishing URLs.
4. Demonstrating the superiority of the proposed Teacher-Student approach by comparing its detection performance against existing methods, achieving higher accuracy and efficiency in phishing URL classification.

The subsequent sections of the paper are structured as follows: Section 2: Relevant Literature. Section 3 outlines the technique utilized for the investigation. Section 4 delineates the findings and conducts a comprehensive series of comparison assessments among the utilized networks, followed by a thorough discussion and analysis of the outcomes in Section 5. Conclusions are presented in Section 6.

2. Literature Review

Several research endeavors have concentrated on developing robust detection methodologies to counteract phishing attempts utilizing machine learning and DL. Saleem [15] provided a lightweight method that utilized solely lexical aspects of URLs, and the findings demonstrated enhanced accuracy of the random forest classifier. Despite machine learning demonstrating commendable efficacy in identifying fraudulent URLs, it faces inherent limitations and obstacles. Tang and Mahmoud [16] utilized machine learning models, such as RNN, Random Forest, and Logistic Regression, to create a browser-based plugin that detects phishing websites and alerts users upon visiting these sites. The study's author determined that the RNN model attained the highest accuracy rate among the assessed models.

Karim et al. [17] used several ML models such as SVM, Decision Tree, Linear Regression, Gradient Boosting Classifier, and K-nearest Neighbor on a dataset including benign and phishing URLs. Using evaluation criteria, the authors attained the best accuracy with Random Forest. Prabakaran et al. [18] enhanced the accuracy of URL detection tasks by integrating VAE with DNN. Their research demonstrates superior performance compared to traditional methods by focusing on the amalgamation of Variational Autoencoders (VAE) with deep neural networks (DNN) to enhance URL identification precision.

Amanullah et al. [19] developed a CNN-based approach for detecting phishing attempts based on URL attributes. Their methodology enhanced the model's efficacy by applying the Optimised Multi Centric Feature Selection strategy. Sahingoz et al. [20] examined DL algorithms for detecting phishing URLs, emphasizing language-agnostic techniques. Their technique demonstrates the adaptability and efficacy of DL in tackling phishing issues through the use of distinct algorithms such as BRNN, CNN, and Attention Networks.

Vaitkevicius et al. [21] evaluated traditional machine learning techniques, determining that ensembles and neural networks performed better. Their research elucidates the efficacy of traditional ML methods in phishing detection, emphasizing ensembles and neural networks as the most effective options for countering phishing threats. Nabila et al. [22] integrated textual and visual information, attaining significant accuracy via multi-modal analysis. Their novel methodology utilizes DL algorithms to extract features from textual URLs and image screenshots, demonstrating the efficacy of multi-modal analysis in enhancing detection accuracy.

Alshingiti et al. [23] examined the identification of phishing URLs using a distributed representation of words obtained from a specific URL. Seven distinct machine learning methodologies were employed to categorize websites as phishing or non-phishing. The algorithms demonstrated a performance level above previous studies. Unrecognized characters were detected, leading to a loss of semantic information while utilizing a bag-of-words model that preserved semantic content.

Although these suggested methodologies are focused on detecting phishing URLs, they are not always accurate, could lead to false positives, and require large amounts of training data to achieve outstanding results. Moreover, many DL approaches involve high computational costs, limiting real-time deployment. To address these challenges, this study employs a Teacher-Student Learning framework, where knowledge from RoBERTa (Teacher) is distilled into DistilRoBERTa (Student), reducing computational complexity while maintaining high detection accuracy.

3. Proposed Method

This section describes the BERTPHIURL approach for detecting phishing cyber-attacks. The proposed framework consists of some main stages, i.e., dataset collection, preprocessing, teacher-student learning model design and training, detection, and performance evaluation. Each stage is explained in Sections 3.1 to 3.4, with an illustration in Figure 2.

3.1. Dataset Collection

This proposed methodology employed a publicly accessible "Phishing Site URLs" dataset from Kaggle [24] to train and test the proposed model for detecting phishing sites. The raw dataset had 549,346 records distributed across two columns. The website links (URLs) are denoted as the primary attribute, whilst labels are indicated as the secondary attribute. The website's labels were categorized as either 'good' or 'bad', with legitimate URLs designated as 'good' and phishing URLs categorized as 'bad'. A sample of the dataset is presented in Table 1.

Table 1. Dataset sample records.

ID	URL	Label
547122	arslan.pro/auuy3to00	Bad
274049	amazon.com/Bande-Sonore-Miniserie-Alys-Robi/dp...	Good
85513	rzr.online.fr/contribs.htm#	Good
423629	riversof.com/	Good
10078	www.plasticoscarmen.com/wp-content/plugins/wor...	Bad

3.2. Preprocessing

Various text preprocessing steps have been employed in the BERTPHIURL approach to enhance the reliability and robustness of the DL model. Firstly, removing duplicate and null URLs, the "Phishing Site URLs" dataset would comprise 507,195 distinct samples. Figure 3 shows the dataset distribution before and after removing duplicate and null URLs.

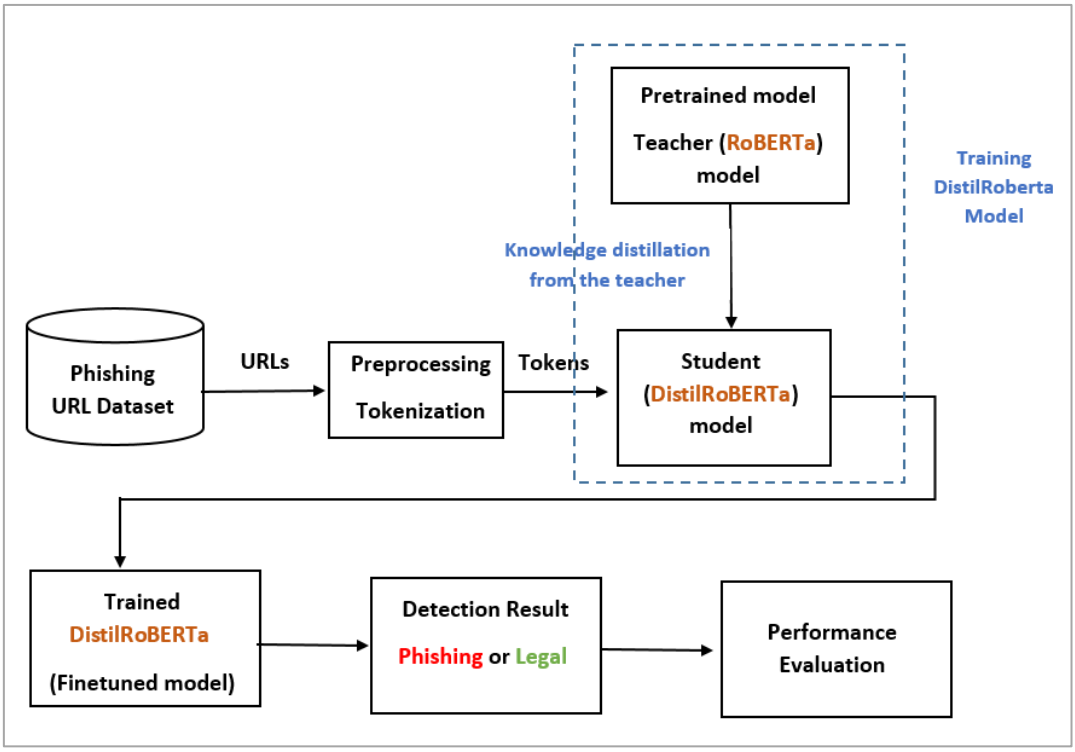

Figure 2. The framework of the BERTPHIURL approach.

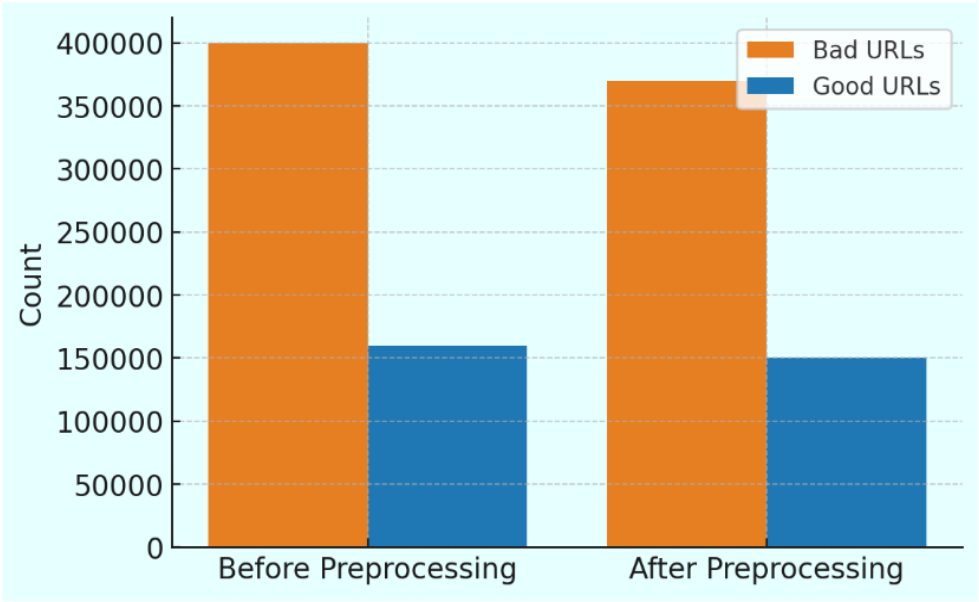

Figure 3. Phishing Site URLs Dataset Distribution after and before removing duplicate and null URLs

After these steps, tokenization is applied to deal with tokens separately. Tokenization is turning a paramount piece of data into a desultory string of characters with no consequential value. The tokenization process has been utilized to break a URL, which is given as a string, and then the given URL has been broken into some tokens by removing numbers and symbols such as (- , _ , @, ets). These tokens have been designated as having significant worth. The stemming process has produced morphological variants of these created tokens. Upon producing the root words, two instances ('Tokenized_text' and 'Stemmed_text') have been established to retain the tokens and root words within the raw dataset. Table 2 presents a sample of tokenization and stemming.

Table 2. Sample of the dataset after tokenization and stemming.

ID	URL	Label	text_tokenized	text_stemmed
155632	cain.ulst.ac.uk/sutton/chron/1969.html	good	[cain, ulst, ac, uk, sutton, chron, html]	[cain, ulst, ac, uk, sutton, chron, html]
492114	user.infernomush-roomrr.org/civis/index.php?PHP...	bad	[user, infernomush-roomrr, org, civis, index, p...	[user, infernomush-roomrr, org, civi, index, ph...
259279	wiki.teamfortress.com/wiki/Killing_Gloves_of_B...	good	[wiki, teamfortress, com, wiki, Killing, Glove...	[wiki, teamfortress, com, wiki, kill, glove, o...
496746	77.222.54.202/upload/_dispatch.php	bad	[upload, dispatch, php]	[upload, dispatch, php]
122051	thenewwrold.net/JAMGD/verification.php	bad	[thenewwrold, net, JAMGD, verification, php]	[thenewwrold, net, jamgd, verif, php]
60129	journals.cambridge.org/action/displayJournal?j...	good	[journals, cambridge, org, action, displayJour...	[journal, cambridg, org, action, displayjourn,...
412211	past-to-present.com/photos.cfm?reference=D12368	good	[past, to, present, com, photos, cfm, referenc...	[past, to, present, com, photo, cfm, refer, d]
386661	mediafire.com/?5tiuldkzju2	good	[mediafire, com, tiuldkzju]	[mediafir, com, tiuldkzju]
3711	bo.lt/d0643:67kgfo2hq7	bad	[bo, lt, d, kgfo, hq]	[bo, lt, d, kgfo, hq]

Finally, The extracted root words of both good and bad URLs have been embedded through the word cloud. This word cloud visualization has been utilized to show the root words. Figures 5 and 6 show the most frequent words in both good and bad URLs after processing.

Figure 5. Word cloud of good website URLs

Figure 6. Word cloud of bad website URLs

3.3. The BERTPHIURL Architecture

Semantic analysis is a technique that focuses on understanding the deeper meaning of textual content by examining word relationships, contextual embeddings, and sentence structures. The DistilRoBERTa transformer has been employed in the proposed BERTPHIURL approach due to its efficient contextual embedding capabilities to extract diverse contextual features from the text. It captures nuanced textual features through self-attention mechanisms, enhancing the model's understanding of complex linguistic patterns.

DistilRoBERTa depends on RoBERTa [25], which stands for Robustly Optimized BERT approach. It is a variant of the BERT model (Bidirectional Encoder Representation from Transformers) [26], which Google developed. DistilRoBERTa can be considered a lightweight model of RoBERTa that mimics the performance, making it faster and more efficient due to its less computational. Table 3 shows the RoBERTa (Teacher) architecture.

Table 3. The RoBERTa (Teacher) architecture.

Layer Type	Description
Input Layer	Tokenized text converted into numerical representations (subwords)
Embedding Layer	Integrates token embeddings and positional embeddings and excludes segment embeddings
Multi-Head Self-Attention	Include 12 heads for Base and 16 for Large for Acquires long-range dependencies.
Feed-forward Layer	It utilizes a fully connected feed-forward network (FFN) and employs ReLU activation.
Layer Normalization & Dropout	Normalizes activations and prevents overfitting
Masked Language Modeling (MLM) Layer	Trains using dynamic masking
Output Layer	Produces contextual embeddings

DistillRoberta is similar to DistilBERT [27] and applies the same idea of knowledge distillation. Still, it depends on the RoBERTa model instead of BERT, simplifying the intricacies of its forerunner, RoBERTa, while preserving its contextual embedding functionalities. The concept utilizes transformer blocks that incorporate self-attention processes, facilitating profound contextual comprehension of token representations essential for discerning the nuances of language. The DistilRoBERTa model is implemented from the transformer library using Hugging Face. Comparatively RoBERTa model performs better than BERT and Distill-BERT. By using DistilRoBERTa, we get the same architecture of the RoBERTa model but with a low computational footprint and better performance. Table 4 shows the Distill-RoBERTa architecture.

Table 4. The DistillRoBERTa (Student) architecture.

Layer Type	Description
Input Layer	Tokenized text converted into numerical representations (subwords)
Embedding Layer	Converts tokens into dense vectors (size 768)
Positional Encoding	Adds positional information to word embeddings
Transformer Encoder (6 Layers)	Core processing layers with self-attention mechanisms. This is used to help the model understand the complex relationship between the words from the input sequence.
Multi-Head Self-Attention (12 heads)	Allows the model to focus on different words in the sequence simultaneously
Feed-Forward Network (FFN)	Two linear layers with activation (size: 3072)
Layer Normalization and Dropout	Normalizes activations and prevents overfitting
Output Layer	Produces contextual embeddings or classification logits

The key difference from RoBERTa is that DistilRoBERTa attains expedited inference by decreasing the transformer layers from 12 to 6 while preserving roughly 95% of its correctness with fewer parameters. Rather than training from the ground up, it uses knowledge distillation, drawing from RoBERTa to improve efficiency. Like RoBERTa, it discards the Next Sentence Prediction (NSP) task and exclusively utilizes Masked Language Modelling (MLM) for training. DistilRoBERTa is trained using knowledge distillation, allowing it to learn from RoB-ERTa instead of training from scratch. In this process, The term teacher model " RoBERTa" refers to the large, pre-trained RoBERTa on large datasets, and the student model "Distil-RoBERTa" refers to a smaller version of RoBERTa that is trained to mimic the teacher's outputs while using fewer parameters and less computational power.

After collecting the dataset and transforming input sequences into contextual representations during the implementation phase, the dataset was partitioned into training, validation, and testing sets, which were then stored in a data dictionary. The tokenizer was subsequently initialized with the DistilRoBERTa pre-trained tokenizer, as both DistilRoBERTa and RoB-ERTa employ BPE tokenization, indicating that their tokenizers function identically. BPE is a subword tokenization method that merges commonly occurring character pairings to create subwords. This helps the model handle uncommon words, misspellings, and morphological variations efficiently. The tokenizer is employed to tokenize the data within the data dictionary, transforming each word or sub-word into unique identifiers. Subsequently, the computational metrics are established to assess the suggested methodology through several indications that gauge the model's efficacy in identifying phishing websites. The output forecast is classified as either valid or phishing for the webpage URLs.

3.4. Evaluation Metrics

Several metrics were used to evaluate the performance of the proposed approach and assess the model's effectiveness, as shown in Table 5.

Table 5. Evaluation Metrics.

Metrics	Formula	Description
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	The proportion of accurately identified fake and safe websites relative to the total cases.
Precision	$\frac{TP}{TP + FP}$	The ratio of accurately identified malicious websites to those classified as phishing.
Recall	$\frac{TP}{TP + FN}$	The ratio of accurately identified phishing websites to the total number of genuine ones.
F1-Score	$2 \times \frac{precision \times recall}{precision + recall}$	The harmonic mean of Precision and recall offers a balanced metric that accounts for both erroneous positives and false negatives.

Where TP represents True Positives, FN represents False Negatives, TN represents True Negatives and FP represents False Positives.

4. Results and Discussion

This section outlines the experimental assessment of the proposed methodology for evaluating the system's performance. The analysis compares the proposed model's performance with previous works using their approaches.

4.1. Model Training and Testing

In this work, the public dataset was used, namely the "Phishing Site URLs dataset," for training and evaluating the DistillRoBERTa model, where 70% of the dataset has been used for training, while the remaining 30% has been used pristinely for validation and testing after preprocessing. When dealing with a complex deep neural network such as the DistillRoB-ERTa model with many layers and parameters, using a larger training set (70% of the data) can have several benefits, such as Improved Generalization and avoiding overfitting. During the training phase, the RoBERTa and DistilRoBERTa models were configured with the hyperparameters in Tables 6 and 7.

Table 6. Training parameters for the RoBERTa (Teacher) model.

Parameter	Value
Loss functions	Masked Language Modeling (MLM) loss
Number of epochs	100
Learning rate	0.00001
batch size	128
Weight Decay	0.01
Optimizer	adam

Table 7. Training parameters for the DistilRoBERTa (Student) model.

Parameter	Value
Loss functions	Masked Language Modeling (MLM) loss and distillation loss
Number of epochs	30
Temperature that Softens logits for knowledge transfer	2
Alpha (α) that Balances MLM loss and distillation loss	0.5
Learning rate	0.00002
batch size	32
Weight Decay	0.01

Our Python code is developed utilizing TensorFlow. The research employs a system equipped with 16 gigabytes of RAM, a 2.30 GHz Intel Core i7 processor, and an NVIDIA GeForce RTX graphics processing unit featuring 8 gigabytes of memory.

4.2. Results of the proposed BERTPHIURL approach

The comprehensive analysis aimed to determine the impact of the DistilRoBERTa transformer on enhancing model performance. This comparative analysis provided in Table 8 presents details of Teacher (RoBERTa) and Teacher-Student (DistilRoBERTa) models of the Training and Validation dataset. The DL models incorporating DistilRoBERTa transformer architectures, which are trained using knowledge distillation from RoBERTa, demonstrate superior performance. The Teacher-Student DistilRoBERTa model achieved impressive accuracy of 92.22% and 88.45 in both the train and validation datasets. It outperformed the Teacher model (RoBERTa). The enhancement achieved represents a 5.55% improvement over the training of the proposed approach using the Teacher model, indicating a significant advancement for our model.

Table 8. Training parameters for the DistilRoBERTa (Student) model.

Model	Accuracy (%)		Loss	
	Train	Validation	Train	Validation
Teacher (RoBERTa)	86.67	82.33	0.2305	0.1724
Student (DistilRoBERTa)	92.22	88.45	0.0532	0.0350

An ablation study was conducted to illustrate the impact of each component on the performance of the proposed approach. Initially, our methodology was trained with the Teacher model, then used the Student model as a backbone, and finally integrated two models, Teacher and Student, into one model. According to Table 9, which provides an overview of the ablated models' performance, the proposed model outperformed the others and yielded an accuracy of 94.22 % in distinguishing between legitimate and phishing URLs. Furthermore, the precision of the proposed approach at 93.45% highlights its capacity to accurately detect positive situations (a phishing URL) while minimizing false positives compared to both teacher and Student models[28]. Notably, the recall metrics of the Teacher and Student models are the lowest at 86.67 and 91.77, signifying these models produce around 13.33% and

8.23% false positive predictions from the test dataset, in contrast to our Teacher-Student model, which generated just 4.89% false positive predictions.

Table 9. A comparative analysis evaluates the test results of the ablation study.

Model	Accuracy%	Precision%	Recall%	F1 Score%
Teacher only	88.89	90.70	86.67	88.64
Student only	91.14	92.09	91.77	91.93
Proposed	94.22	93.45	95.11	94.27

5. Comparison of the proposed BERTPHIURL approach with Prior Works

To emphasize the importance and contribution of the work in this paper, the suggested BERTPHIURL approach is compared with Some studies on phishing URL detection. Studies show that different scenarios work with various methods and provide different outcomes. According to these findings, our proposed BERTPHIURL approach has shown 94.22% accuracy, which can distinguish phishing from legitimate websites. The comparison results are reported in Table 10 below.

Table 10. Comparison of the BERTPHIURL approach with phishing URL detection-based previous studies.

Authors	Year	Architecture	Accuracy (%)
Korkmaz et al. [29]	2020	ANN	88.22
Subasi et al. [30]	2021	Adaboost	85.0
Hossen et al. [31]	2021	GRU	78.55
Jain et al. [32]	2021	CNN-LSTM	76.02
He et al. [33]	2021	XG Boost	88.0
Tan et al. [34]	2022	RoBERTa-LSTM	91.37
Omari [35]	2023	Logistic Regression	92.7
Dhanavanthini et al. [36]	2023	recurrent neural network	90.50
Raminedi et al. [37]	2023	Logistic Regression	92.28
Benavides-Astudillo et al. [38]	2023	RNN	80.0
Ige et al. [39]	2024	SVM	90.6
Our proposed approach	2025	RoBERTa → DistilRoBERTa	94.22

6. Conclusions

Phishing is a recognized crime that acquires consumers' information using a URL resembling a legitimate webpage. Phishing attacks are crucial for thwarting unauthorized access to users' information. The increase in victims, primarily due to inadequate security technology implementation, necessitates the development of an intelligent method to safeguard users against cyber-attacks. Phishing and text-based cyberattacks pose significant threats due to their linguistic complexity, complicating detection and mitigation efforts. Attackers use ambiguous phrasing, and multilingual attacks introduce language variations. Developing robust NLP frameworks is crucial for identifying and mitigating these threats. In this work, the NLP framework called BERTPHIURL, inspired by the DistilRoBERTa transformer, was proposed to detect and classify the URLs of the website. It achieved an accuracy of 94.22% on the testing set using the "Phishing Site URLs " dataset. In the future, we intend to incorporate hybrid and multi-model strategies to enhance the accuracy and training of our proposed model on a larger dataset.

Author Contributions: This project was executed in collaboration among all contributors. P.H.H. and S.M.M. contributed to the resolution of the problem statement. P.H.H. engaged in the training model, analyzed the findings, and authored the paper. The researcher, S.M.M., analyzed the results and edited and proofread the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset can be downloaded via URL https://www.kaggle.com/datasets/taruntiwarihp/phishing-site-urls/data

Conflicts of Interest: The authors declare no conflict of interest

References

[1] M. Somesha, A. R. Pais, R. S. Rao, and V. S. Rathour, "Efficient deep learning techniques for the detection of phishing websites," *Sādhanā*, vol. 45, no. 1, p. 165, Dec. 2020, doi: 10.1007/s12046-020-01392-4.

[2] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 2, pp. 590–611, Feb. 2023, doi: 10.1016/j.jksuci.2023.01.004.

[3] W. Sarasjati, S. Rustad, Purwanto, H. A. Santoso, and D. R. I. M. Setiadi, "Phishing Detection Using Random Forest-Based Weighted Bootstrap Sampling and LASSO+ Feature Selection," *Int. J. Saf. Secur. Eng.*, vol. 14, no. 6, pp. 1783–1794, Dec. 2024, doi: 10.18280/ijsse.140613.

[4] Y. Said, A. A. Alsheikhy, H. Lahza, and T. Shawly, "Detecting phishing websites through improving convolutional neural networks with Self-Attention mechanism," *Ain Shams Eng. J.*, vol. 15, no. 4, p. 102643, Apr. 2024, doi: 10.1016/j.asej.2024.102643.

[5] S. Nagarajan, "An Investigation of AI-Enabled Comprehensive Survey of Phishing Attacks detection techniques," *Int. J. Innov. Res. Eng.*, vol. 5, no. 1, pp. 59–63, Sep. 2024, doi: 10.59256/ijire.20220301001.

[6] D. R. I. M. Setiadi, S. Widiono, A. N. Safriandono, and S. Budi, "Phishing Website Detection Using Bidirectional Gated Recurrent Unit Model and Feature Selection," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 2, pp. 75–83, Jul. 2024, doi: 10.62411/faith.2024-15.

[7] F. Heiding, B. Schneier, A. Vishwanath, J. Bernstein, and P. S. Park, "Devising and Detecting Phishing Emails Using Large Language Models," *IEEE Access*, vol. 12, pp. 42131–42146, 2024, doi: 10.1109/ACCESS.2024.3375882.

[8] G. Desolda, L. S. Ferro, A. Marrella, T. Catarci, and M. F. Costabile, "Human Factors in Phishing Attacks: A Systematic Literature Review," *ACM Comput. Surv.*, vol. 54, no. 8, pp. 1–35, Nov. 2022, doi: 10.1145/3469886.

[9] T. Xu and P. Rajivan, "Determining psycholinguistic features of deception in phishing messages," *Inf. Comput. Secur.*, vol. 31, no. 2, pp. 199–220, May 2023, doi: 10.1108/ICS-11-2021-0185.

[10] C.-S. Lin, C.-N. Tsai, J.-S. Jwo, C.-H. Lee, and X. Wang, "Heterogeneous Student Knowledge Distillation From BERT Using a Lightweight Ensemble Framework," *IEEE Access*, vol. 12, pp. 33079–33088, 2024, doi: 10.1109/ACCESS.2024.3372568.

[11] L. Wang and K.-J. Yoon, "Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3048–3068, Jun. 2022, doi: 10.1109/TPAMI.2021.3055564.

[12] J. K. S and A. B, "Phishing URL detection by leveraging RoBERTa for feature extraction and LSTM for classification," in *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, Aug. 2023, pp. 972–977. doi: 10.1109/ICAISS58487.2023.10250684.

[13] M. Songailaitė, E. Kankevičiūtė, B. Zhyhun, and J. Mandravickaitė, "BERT-Based Models for Phishing Detection," in *CEUR Workshop Proceedings*, 2023, vol. 3575, pp. 34–44.

[14] K. Zhang, C. Zhang, S. Li, D. Zeng, and S. Ge, "Student Network Learning via Evolutionary Knowledge Distillation," *ArXiv*. Mar. 22, 2021. [Online]. Available: http://arxiv.org/abs/2103.01381

[15] A. Saleem Raja, R. Vinodini, and A. Kavitha, "Lexical features based malicious URL detection using machine learning techniques," *Mater. Today Proc.*, vol. 47, pp. 163–166, 2021, doi: 10.1016/j.matpr.2021.04.041.

[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] A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari, and S. R. K. Joga, "Phishing Detection System Through Hybrid Machine Learning Based on URL," *IEEE Access*, vol. 11, pp. 36805–36822, 2023, doi: 10.1109/ACCESS.2023.3252366.

[18] M. K. Prabakaran, P. Meenakshi Sundaram, and A. D. Chandrasekar, "An enhanced deep learning-based phishing detection mechanism to effectively identify malicious URLs using variational autoencoders," *IET Inf. Secur.*, vol. 17, no. 3, pp. 423–440, May 2023, doi: 10.1049/ise2.12106.

[19] M. Amanullah, V. Selvakumar, A. Jyot, N. Purohit, S. S, and M. Fahlevi, "CNN based Prediction Analysis for Web Phishing Prevention," in *2022 International Conference on Edge Computing and Applications (ICECAA)*, Oct. 2022, pp. 1–7. doi: 10.1109/ICECAA55415.2022.9936112.

[20] O. K. Sahingoz, E. BUBEr, and E. Kugu, "DEPHIDES: Deep Learning Based Phishing Detection System," *IEEE Access*, vol. 12, no. December, pp. 8052–8070, 2024, doi: 10.1109/ACCESS.2024.3352629.

[21] P. Vaitkevicius and V. Marcinkevicius, "Comparison of Classification Algorithms for Detection of Phishing Websites," *Informatica*, vol. 31, no. 1, pp. 143–160, Mar. 2020, doi: 10.15388/20-INFOR404.

[22] O. G. Nabila, H. R. Wicaksono, Girinoto, R. N. Yasa, and H. Setiawan, "Benchmarking Model URL Features and Image Based for Phishing URL Detection," in *2023 International Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS)*, Nov. 2023, pp. 177–182. doi: 10.1109/ICIMCIS60089.2023.10349059.

[23] 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*, vol. 12, no. 1, p. 232, Jan. 2023, doi: 10.3390/electronics12010232.

[24] T. Tiwari, "Phishing Site URLs," *Kaggle.com*, 2017. https://www.kaggle.com/datasets/taruntiwarihp/phishing-site-urls

[25] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *ArXiv*. Jul. 26, 2019.

[26] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 4171–4186.

[27] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *ArXiv*. pp. 2–6, Oct. 02, 2019. [Online]. Available: http://arxiv.org/abs/1910.01108

[28] A. Çetin and S. Öztürk, "Comprehensive Exploration of Ensemble Machine Learning Techniques for IoT Cybersecurity Across Multi-Class and Binary Classification Tasks," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 371–384, Feb. 2025, doi: 10.62411/faith.3048-3719-51.

[29] M. Korkmaz, O. K. Sahingoz, and B. Diri, "Detection of Phishing Websites by Using Machine Learning-Based URL Analysis," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Jul. 2020, pp. 1–7. doi: 10.1109/ICCCNT49239.2020.9225561.

[30] A. Subasi and E. Kremic, "Comparison of Adaboost with MultiBoosting for Phishing Website Detection," *Procedia Comput. Sci.*, vol. 168, pp. 272–278, 2020, doi: 10.1016/j.procs.2020.02.251.

[31] M. S. Hossen, A. H. Jony, T. Tabassum, M. T. Islam, M. M. Rahman, and T. Khatun, "Hotel review analysis for the prediction of business using deep learning approach," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, Mar. 2021, pp. 1489–1494. doi: 10.1109/ICAIS50930.2021.9395757.

[32] P. K. Jain, V. Saravanan, and R. Pamula, "A Hybrid CNN-LSTM: A Deep Learning Approach for Consumer Sentiment Analysis Using Qualitative User-Generated Contents," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 20, no. 5, pp. 1–15, Sep. 2021, doi: 10.1145/3457206.

[33] S. He, B. Li, H. Peng, J. Xin, and E. Zhang, "An Effective Cost-Sensitive XGBoost Method for Malicious URLs Detection in Imbalanced Dataset," *IEEE Access*, vol. 9, pp. 93089–93096, 2021, doi: 10.1109/ACCESS.2021.3093094.

[34] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis With Transformer and Recurrent Neural Network," *IEEE Access*, vol. 10, pp. 21517–21525, 2022, doi: 10.1109/ACCESS.2022.3152828.

[35] K. Omari, "Comparative Study of Machine Learning Algorithms for Phishing Website Detection," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 9, pp. 417–425, 2023, doi: 10.14569/IJACSA.2023.0140945.

[36] P. Dhanavanthini and S. S. Chakkravarthy, "Phish-armour: phishing detection using deep recurrent neural networks," *Soft Comput.*, Mar. 2023, doi: 10.1007/s00500-023-07962-y.

[37] S. Raminedi, T. N. Pandey, V. A. Woonna, S. C. Mascarenhas, and A. Bharani, "Classification of Phishing Websites using Machine Learning Models," in *2023 3rd International conference on Artificial Intelligence and Signal Processing (AISP)*, Mar. 2023, pp. 1–5. doi: 10.1109/AISP57993.2023.10134944.

[38] E. Benavides-Astudillo, W. Fuertes, S. Sanchez-Gordon, G. Rodriguez-Galan, V. Martínez-Cepeda, and D. Nuñez-Agurto, "Comparative Study of Deep Learning Algorithms in the Detection of Phishing Attacks Based on HTML and Text Obtained from Web Pages," in *Applied Technologies*, M. Botto-Tobar, M. Z. Vizuete, S. M. León, P. Torres-Carrión, and B. Durakovic, Eds. Springer Nature Switzerland, 2023, pp. 386–398. doi: 10.1007/978-3-031-24985-3_28.

[39] T. Ige, C. Kiekintveld, A. Piplai, A. Waggler, O. Kolade, and B. H. Matti, "An investigation into the performances of the Current state-of-the-art Naive Bayes, Non-Bayesian and Deep Learning Based Classifier for Phishing Detection: A Survey," *ArXiv*. Nov. 24, 2024. [Online]. Available: http://arxiv.org/abs/2411.16751