A Comparative Study on Detecting Phishing URLs Leveraging Pre-trained BERT Variants

1st Chanchal Patra
Department of Information Technology,
Maulana Abul Kalam Azad University of Technology,
West Bengal, Haringhata, Nadia 741249, India
Email: chanchalpatra89@gmail.com

2nd Debasis Giri
Department of Information Technology,
Maulana Abul Kalam Azad University of Technology,
West Bengal, Haringhata, Nadia 741249, India
Email: debasis_giri@hotmail.com

3rd Tanmoy Maitra
School of Computer Engineering,
KIIT Deemed to be University,
Odisha, Bhubaneswar 751024, India
Email: tanmoy.maitrafcs@kiit.ac.in

4th Bibekananda Kundu
CDAC Kolkata,
Salt Lake Electronics Complex, Sector V,
West Bengal 700091, India
Email: bibekananda.kundu@gmail.com

2024 6th International Conference on Computational Intelligence and Networks (CINE) | 979-8-3315-1679-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/CINE63708.2024.10881521

***Abstract*—In the cybersecurity sector, phishing attacks are one of the new security concerns that have received a lot of attention lately. Phishing efforts that try to steal confidential data or spread malware frequently use malicious universal resource locators (URLs). Accurately identifying phishing URLs is crucial. There are several methods available now for identifying phishing websites. Nevertheless, as attackers might adapt their strategies to evade recently implemented detection techniques, phishing website detection continues to be a research focus. Our study presents an enhanced model for detecting phishing URLs, which is based on optimizing the BERT model family to identify phishing URLs with more precision. BERT variants such as ALBERT, DistilBERT, and RoBERTa are used in the phishing website detection model presented in this research. These models perform significantly better than existing detection techniques and demonstrate impressive accuracy. In the experiment, we found that the transformer-based BERT model achieved the highest accuracy, precision, recall, and F1-score of 99.29%, 99.26%, 99.27%, and 99.27%, respectively. As per the findings, out of all the models that are now accessible, this one performs the best.**

***Index Terms*—Phishing attacks, Cybersecurity, Phishing URL detection, BERT**

I. Introduction

One social engineering tactic that hackers employ is called phishing and is one of the major threats today. As a form of social engineering, phishing has caused tremendous financial loss to recipients. It involves cyber criminals impersonating legitimate organizations' websites or URLs to deceive victims [4]. An average of 758,000 complaints per year have been received by the Internet Crime Complaint Center (IC3) during the past five years. The complaints in question pertain to a diverse range of online frauds that impact people worldwide [7]. Data on complaints and losses from 2019 to 2023, both annual and aggregate, are shown in Fig. 1. 3.79 million complaints about a reported loss of $37.5 billion were received by the IC3 during this time. The anti-phishing working group (APWG) [2] published about phishing activity trends report states that 1,077,501 phishing assaults were detected in 2023's fourth quarter. 2023 was the worst year for phishing ever recorded, with over five million assaults, according to APWG. Late in 2023, there was an increase in assaults on social media sites, which accounted for 42.8% of all phishing attacks.

Therefore, there is a necessity for phishing detection with high accuracy. Phishing is the practice of creating a harmful website with an appearance and feel similar to the authentic login page of a well-known online business, with the goal of obtaining credit card or other payment information or user passwords. Social networking platforms, web-based email portals, and online banking services are common targets for phishing attacks. Attackers employ a variety of techniques to lure victims to the phishing website so they can begin the assault.

Many anti-phishing systems are now in use to lessen the impact of phishing attempts. Indicators for phishing websites are already included in recent web browsers. As an illustration, browsers would verify the SSL certificate of the website and present the outcome to users in the form of a unique indicator symbol in the address bar. Regretfully, if the attacker utilized a URL that was strikingly close to the address of the actual website, this validation procedure might be readily circumvented. Research revealed that these indications may not be useful at all and may potentially place consumers at greater danger [6]. In order to avoid serious repercussions and possible losses, it would be crucial to find trustworthy and portable harmful URL detection tools that can quickly detect dangerous URLs.

This paper's contributions are as following:

- The research presents an improved phishing URL detection model by fine-tuning variants of the BERT family (ALBERT, DistilBERT, and RoBERTa), offering significantly better performance than existing techniques.
- The study leverages natural language processing models (BERT variants) for detecting phishing URLs, demon-

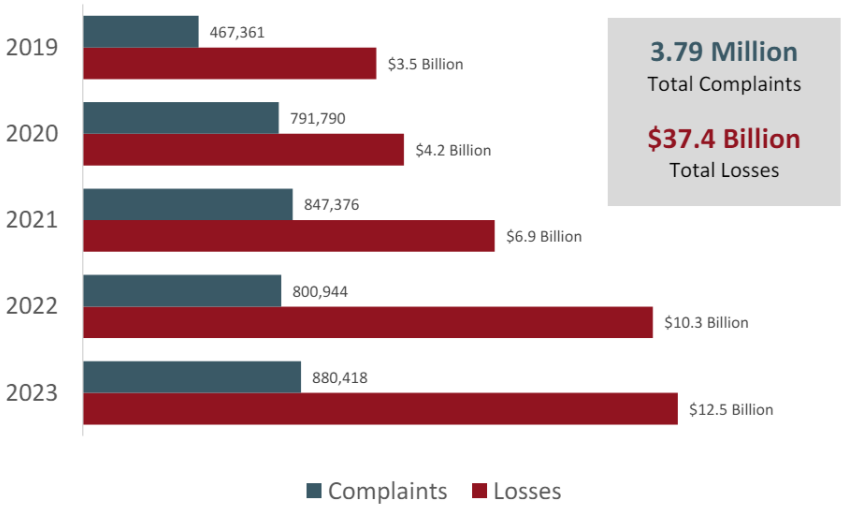

Fig. 1. Complaints and Losses over the Last Five Years [7]

strating their applicability and effectiveness in cybersecurity.

- The transformer-based model developed in this work achieves state-of-the-art accuracy, outperforming traditional detection models and ensuring more reliable identification of phishing websites.
- The model is tested using a publicly available dataset of legitimate and phishing URLs from the UCI Machine Learning Repository, and its performance is systematically compared to classical detection models, validating its superior accuracy.
- By addressing the adaptability of phishing tactics and the need for high-accuracy detection methods, the study contributes to ongoing research efforts aimed at improving cybersecurity defenses against evolving phishing strategies.

This study is divided into the following sections, in the following order: The literature review is covered in Section II. The pre-trained transformer models are described in Section III. The dataset's details are given in Section IV. Section V discusses the experimental methods. The evaluation result analysis and explanation of the results are included in Section VI. Comparison with related works discuss in Section VII. Section VIII offers further work as the paper's conclusion.

II. Literature Review

A comprehensive examination and conceptual understanding of machine learning-based malicious URL detection techniques are offered by Sahoo et al. [19]. They categorize the literature research that address different facets of the problem, explicitly characterize the challenge of malicious URL detection as one of machine learning, and evaluate the contributions of these research.

A Systematic Literature Review (SLR) that investigates and compares different phishing detection strategies, such as list-based, visual similarity, heuristic, machine learning, and deep learning techniques, is discussed by Safi et al. [18].

Using a combination of feature extraction and analysis, Mahajan et al. [13] examine the use of machine learning technology for phishing URL detection. Various machine learning algorithms were employed to detect phishing websites.

In order to provide a strategy for identifying phishing websites using URL analysis, Sanchez et al. [20] compare machine learning and deep learning methodologies.

By focusing on the behaviors and features of the provided URL, machine learning methods are used by Ahammad et al. [1] to detect malicious websites.

An efficient machine learning framework for identifying phishing URLs without visiting the website or utilizing third-party services was given by Jalil et al. [9]. In order to classify phishing URLs, the method uses the TF-IDF technique to analyze the entire URL, including the protocol scheme, hostname, path, entropy properties, suspicious phrases, and brand name matching.

Le et al. [11] present URLNet, a deep learning system that trains a nonlinear URL embedding for harmful URL detection directly from the URL using CNN. They conduct in-depth tests on a sizable dataset and demonstrate a notable improvement in performance over current approaches.

For malicious URL identification, Ren et al. [17] suggested an attention-based BiLSTM (AB-BiLSTM) model. The model employs an attention mechanism, Word2Vec for URL preprocessing, and BiLSTM for extracting sequence features. It achieved 98.06% accuracy in experiments on a collected dataset.

A Phishing URL detection framework called PhiUSIIL is presented by Prasad et al. [16]. It uses incremental learning and a similarity index. This framework constantly refreshes its knowledge base and detects visual similarity-based assaults with efficacy. When tested, PhiUSIIL's accuracy was 99.24%.

A thorough examination of transformer models is conducted

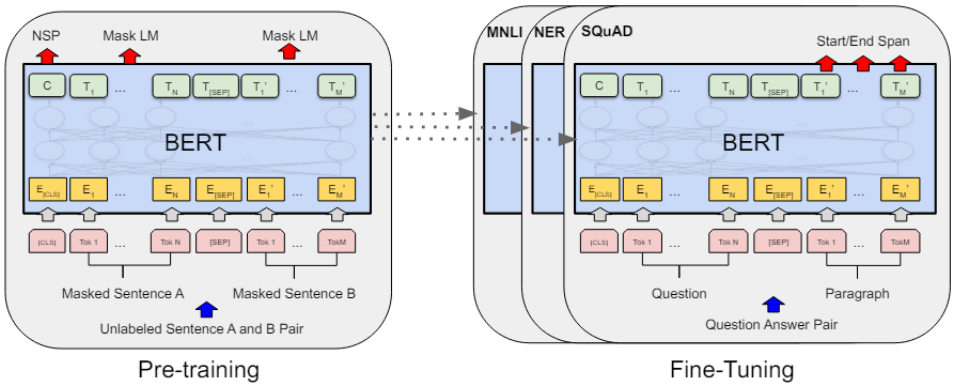

Fig. 2. Pre-training and fine-tuning procedures for BERT [5]

by Maneriker et al. [14] for the purpose of phishing URL detection. They contrast these models with RoBERTa and BERT optimized models, taking into account extra domain-specific pre-training activities and conventional masked language modeling. The eXpose neural network is proposed by Saxe et al. [22]. It employs a deep learning technique that learns to use CNN with character-level embeddings to simultaneously extract the features and classify the input's generic, unprocessed, brief character strings.

Huang et al. [8] presented a novel method for detecting phishing websites that uses a capsule-based neural network with parallel branches to analyze URLs. This method outperformed other detection approaches while keeping a manageable time overhead. It proved to be both efficient and effective.

Only URLs that are a component of a webpage, used to training the PhishTransformer model, according to Asiri et al. [3]. This approach uses transformer encoders and convolutional neural networks to extract data from webpage text and URLs. The classifier that is developed using these features is then able to distinguish between reliable websites and phishing schemes.

III. Pre-trained Transformer Models

In recent years, Transformers have gained popularity as a key area of research for various character-related tasks. These deep learning models, based on the encoder-decoder architecture, are widely used in natural language processing (NLP) applications such as question-answering, sentiment analysis, and language modeling. The introduction of the Transformer model by Vaswani et al. [24] has led to the displacement of RNN and LSTM-based sequence classification methods. Transformers use a mathematical technique called self-attention to discover complex relationships between different parts of a sequence. State-of-the-art pretrained transformer models, trained on large corpora, include BERT, ALBERT, DistilBERT, RoBERTa, DeBERTa, XLNet, MPNet, and others. In the following, we discuss some BERT variant models.

A. BERT

BERT, based on the Transformer architecture, is a Bidirectional Encoder Representation model that predicts context in various ways. It is pre-trained on large amounts of unannotated text and can be fine-tuned for specific tasks and datasets, leveraging transfer learning to achieve high accuracy with faster computation. Fig. 2 illustrates the typical pre-training and fine-tuning process for the BERT model, where the same architecture is used, except for the output layers. A [CLS] token is added to each input, and a [SEP] token separates question-answer pairs [5]. BERT is open-source and widely researched, producing state-of-the-art predictions. Since its release, many other versions have been developed, extending BERT's application across various industries.

B. ALBERT

ALBERT [10], short for 'A Lite BERT,' was developed to improve the efficiency of BERT by reducing the number of parameters. While BERT models have around 110 million parameters, ALBERT addresses the computational challenges with fewer parameters, making training faster. ALBERT uses two key methods: cross-layer parameter sharing, where parameters of the first encoder layer are shared across all layers, and factorized embedding parameterization, which reduces dimensions by separating input and hidden layer embeddings.

C. DistilBERT

DistilBERT [21] is a compressed, faster, and smaller version of BERT, designed to address the resource-intensive nature of BERT's large size and millions of parameters, which limit its use in practical applications, especially on mobile devices. Using a 'teacher-student' structure, or knowledge distillation, DistilBERT transfers knowledge from a larger 'teacher' network to a smaller 'student' network, improving performance while reducing computation time compared to models like BERT, RoBERTa, and XLNet.

D. RoBERTa

RoBERTa, or 'BERT Pretraining Approach with Robust Optimization' [12], enhances BERT's training by using larger

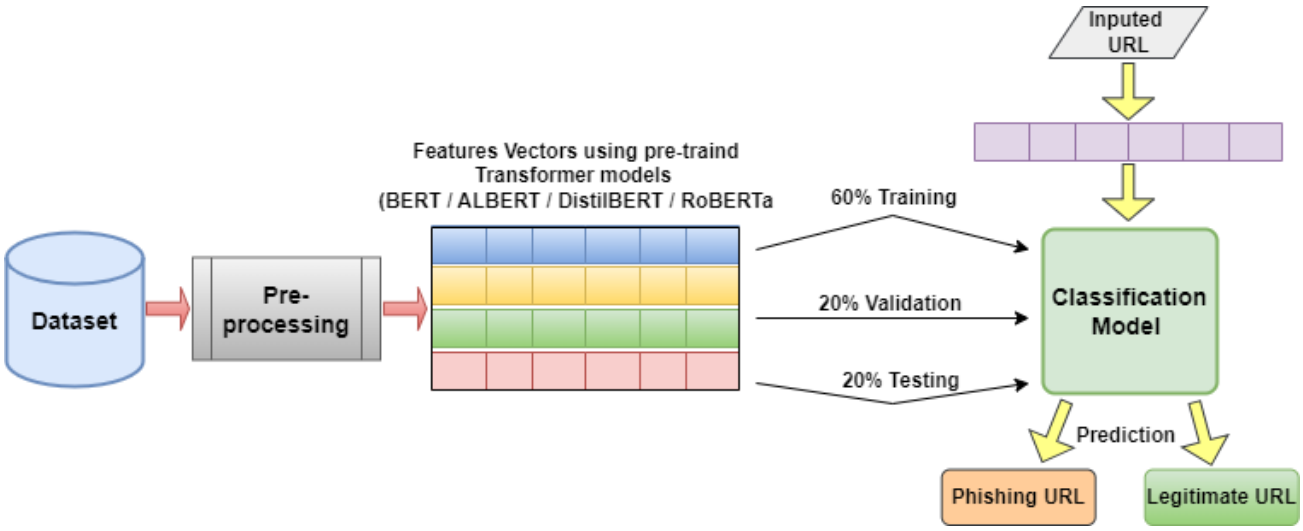

Fig. 3. Phishing URL detection flowgraph

datasets, longer sequences, and larger mini-batches. Unlike BERT, RoBERTa employs dynamic masking, where 15% of input sequences are randomly masked, allowing the model to learn multiple masking patterns within the same sequence. It also eliminates Next Sentence Prediction (NSP), instead feeding full sentences and marking document boundaries with a separator token. Additionally, RoBERTa performs better when trained in large mini-batches, significantly improving performance compared to BERT, which originally trains only 256 sequences at a time.

IV. Dataset Description

The UCI machine learning repository [23] and PhishTank [15] are the two sources that make up our dataset. A variety of formats, including.json and.csv, are available for the hourly updated phishing URL sets that PhishTank offers. By allowing the community to validate the phishing URLs on PhishTank, the original dataset is of higher quality. Research on phishing detection frequently uses PhishTank, a popular source of phishing URLs. Both phishing and genuine URLs may be found in the other UCI dataset. For training, validation, and testing purposes, we randomly picked 27,000 URLs from the datasets, including 14,100 benign URLs and 12,900 phishing URLs. Every chosen URL has a mark of 0 or 1, denoting benignity or phishing, respectively. We used 60% (16200 URLs) of the whole dataset for training, 20% (5400 URLs) for validation (to make sure the model is resilient and fine-tuned), and 20% (5400 URLs) for testing (to check how well the model performs on URLs that haven't been seen before). To increase the robustness of our model, we shuffled our labeled dataset prior to the training phase.

V. Experiment

The experimental process involves using pre-trained transformer models, specifically variants of BERT, including ALBERT, DistilBERT, and RoBERTa, to analyze the URLs. The flow graph of the phishing URL detection system is displayed in Fig. 3. The steps taken in the experiment include:

1) **Feature Extraction and Model Setup:** Pre-trained transformer models are leveraged to generate feature vectors from the URLs. These models are already fine-tuned for natural language processing (NLP) tasks. URL strings, treated as textual data, are passed through the transformer models to obtain embeddings.
2) **Model Training and Validation:** The dataset is processed in batches where 60% of the URLs are used to train the models, 20% is utilized for validation, and the remaining 20% is reserved for testing the trained models. The pre-trained models are fine-tuned using labeled URLs from the training dataset.
3) **Model Testing:** After training, the models are tested using the 20% test dataset to evaluate their performance in classifying phishing and legitimate URLs. Predictions from the models are compared to the actual labels.

The next section discusses the evaluation of classification models.

VI. Evaluation Results and Discussion

Our approach involved gathering and analyzing data, analyzing it, training the model, and then evaluating it to assess its success. We tested the prediction on the test dataset, which consists of 20% of records with both harmful and benign URLs combined, in order to assess our model. Following that, we created the confusion matrix using the original labels and the prediction results. In our experiment, the values of True negative (β) indicate the percentage of legitimate URLs that are incorrectly predicted as phishing; False positive (γ) indicates that the percentage of legitimate URLs that are incorrectly classified as phishing; False negative (δ) indicates the percentage of phishing URLs that are incorrectly classified as legitimate; and True positive (α) indicates the percentage of phishing URLs that are incorrectly classified as phishing. The

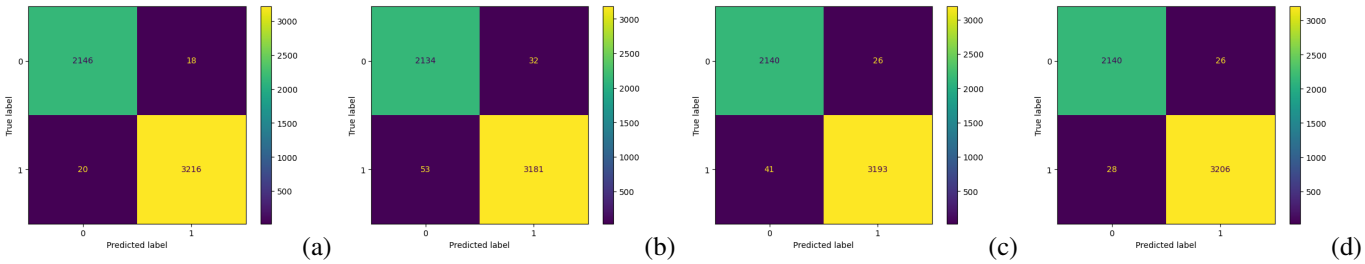

Fig. 4. Confusion matrices of classification models using BERT(a), ALBERT(b), DistilBERT(c) and RoBERTa(d).

TABLE I
PRECISION, RECALL, SPECIFICITY, F1-SCORE AND ACCURACY VALUES WITH RESPECT TO DIFFERENT CLASSIFICATION MODELS USING BERT VARIANTS ON TESTING DATASET

Model	Precession (P)	Recall (R)	Specificity (S)	F1-Score (F1)	Accuracy (A)
BERT	**99.2600**	**99.2750**	**99.4434**	**99.2675**	**99.2963**
ALBERT	**98.2903**	**98.4418**	**99.0040**	**98.3644**	**98.4259**
DistilBERT	**98.6562**	**98.7659**	**99.1923**	**98.7102**	**98.7592**
RoBERTa	**98.9520**	**98.9669**	**99.1955**	**98.9594**	**99.00**

confusion matrices for the classification models are displayed in Fig. 4. True positives (α) and true negatives (β) should predominate in the confusion matrix, while false positives (γ) and false negatives (δ) should be in smaller numbers. Based on the data displayed in the confusion matrix, our transformer models performs exceptionally well generally, despite the possibility of rare prediction errors.

Other metrics that we utilized to evaluate our Transformer model were accuracy, precision, recall, specificity, and F1-score. When calculating correct expected results over all forecasts, accuracy is the most logical statistic to use. The precise ratio of all positive forecasts to exact positive predictions is the measure of precision. Recall displays the accurate positive predictions in relation to the real positive results for a given class. Specificity is a metric that compares accurate negative forecasts to actual negatives. The weighted average of recall and accuracy is also computed via the F1-score. Based on the confusion matrix, the accuracy, precision, recall, and F1-score of the BERT pre-trained model are 99.29%, 99.26%, 99.27%, and 99.27%, respectively (see Table I). In comparison to the other three pre-trained models, these values are the best, demonstrating the superior performance of our model.

$$Accuracy(A) = \frac{(\alpha + \beta)}{(\alpha + \gamma + \beta + \delta)} \tag{1}$$

$$Precision(P) = \frac{\alpha}{(\alpha + \gamma)} \tag{2}$$

$$Recall(R) = \frac{\alpha}{(\alpha + \delta)} \tag{3}$$

$$Specificity(S) = \frac{\beta}{(\beta + \gamma)} \tag{4}$$

$$F1 - Score = \frac{2 * (P * R)}{(P + R)} \tag{5}$$

VII. MODEL COMPARISON

In this experiment, our method is compared with the industry standard used by the most advanced URL-based models. In order to do this, we used our test set to evaluate the models after they had been trained using our training and validation sets. As shown in Table II, our model achieved an accuracy of 99.29%, which is significantly better than the others.

VIII. CONCLUSION AND FUTURE WORK

In this study, we conducted a literature evaluation of several methods for predicting dangerous URLs. Our Transformer classifier approach for phishing URL prediction was introduced, drawing on the body of research already conducted in this area. We also presented our training dataset and associated training procedures. In addition, we assessed the models based on their quality and how well they performed on the validation dataset. Using our model comparison dataset to undertake predictions, After comparing the four models' performances, we discovered that the BERT-based model had the best overall performance. Robustness might be enhanced

TABLE II
COMPARISON WITH RELATED WORKS

Reference	Used Features	Model	Performance
Jalil et al. [9]	TF-IDF	Machine learning (Random forest)	96.85% Accuracy
Mahajan et al. [13]	Manual features extraction and selection	Machine learning algorithms	97.14% Accuracy
Le et al. [11]	URL embedding	Convolutional Neural Networks	98.85% Accuracy
Ren et al. [17]	Word2Vec	Attentional-based BiLSTM	98.06 % Accuracy
Prasad et al. [16]	Facial expressions and pulse rate	Machine learning (Random forest)	99.24% Accuracy
Our Model	Natural Language Processing (NLP) with Word Embedding	BERT Variants	99.29 % Accuracy

by further training with a larger dataset and the present finished model. More improvements should be made to the robustness and efficiency, particularly when utilizing a dataset that has a large percentage of short URLs.

REFERENCES

[1] S. H. Ahammad, S. D. Kale, G. D. Upadhye, S. D. Pande, E. V. Babu, A. V. Dhumane, and M. D. K. J. Bahadur, "Phishing url detection using machine learning methods," *Advances in Engineering Software*, vol. 173, p. 103288, 2022.

[2] APWG, "Phishing activity trends report (https://apwg.org/trendsreports/)," 4th Quarter, 2023.

[3] S. Asiri, Y. Xiao, and T. Li, "Phishtransformer: A novel approach to detect phishing attacks using url collection and transformer," *Electronics*, vol. 13, no. 1, p. 30, 2023.

[4] D. G. Bachrach and E. J. Rzeszut, *10 Don'ts on Your Digital Devices: The Non-techie's Survival Guide to Cyber Security and Privacy*. Springer, 2014.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[6] S. Egelman, *Trust me: Design patterns for constructing trustworthy trust indicators*. Carnegie Mellon University, 2009.

[7] FBI, "Ic3 (internet crime complaint center) annual report (https://www.ic3.gov/Home/AnnualReports)," 2023.

[8] Y. Huang, J. Qin, and W. Wen, "Phishing url detection via capsule-based neural network," in *2019 IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID)*. IEEE, 2019, pp. 22–26.

[9] S. Jalil, M. Usman, and A. Fong, "Highly accurate phishing url detection based on machine learning," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 7, pp. 9233–9251, 2023.

[10] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[11] H. Le, Q. Pham, D. Sahoo, and S. C. Hoi, "Urlnet: Learning a url representation with deep learning for malicious url detection," *arXiv preprint arXiv:1802.03162*, 2018.

[12] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[13] R. Mahajan and I. Siddavatam, "Phishing website detection using machine learning algorithms," *International Journal of Computer Applications*, vol. 181, no. 23, pp. 45–47, 2018.

[14] P. Maneriker, J. W. Stokes, E. G. Lazo, D. Carutasu, F. Tajaddodianfar, and A. Gururajan, "Urltran: Improving phishing url detection using transformers," in *MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM)*. IEEE, 2021, pp. 197–204.

[15] Phishtank, "[online]. (available: http://data.phishtank.com/data/ online-valid.csv)," August 2021.

[16] A. Prasad and S. Chandra, "Phiusiil: A diverse security profile empowered phishing url detection framework based on similarity index and incremental learning," *Computers & Security*, vol. 136, p. 103545, 2024.

[17] F. Ren, Z. Jiang, and J. Liu, "A bi-directional lstm model with attention for malicious url detection," in *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. IEEE, 2019, pp. 300–305.

[18] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 2, pp. 590–611, 2023.

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

[20] M. Sánchez-Paniagua, E. F. Fernández, E. Alegre, W. Al-Nabki, and V. Gonzalez-Castro, "Phishing url detection: A real-case scenario through login urls," *IEEE Access*, vol. 10, pp. 42 949–42 960, 2022.

[21] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[22] J. Saxe and K. Berlin, "expose: A character-level convolutional neural network with embeddings for detecting malicious urls, file paths and registry keys," *arXiv preprint arXiv:1702.08568*, 2017.

[23] UCI, "[online]. (available: https://archive.ics.uci.edu/dataset/967/ phiusiil+phishing+url+dataset)," March 2024.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.