Swati , 2026, 14:4
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Transformer-Based Phishing URL Detection with Lightweight Context-Aware URL Encoding

Swati, Disha Vishwakarma, Anuradha Singh, Anusha Rathore
[1]Department of Computer Science ABES Engineering College
Ghaziabad, India,

Abstract- The cybersecurity menace that is phishing attacks continues to thrive although online criminals are always intent on developing deceitful links that are exact replicas of genuine websites. Conventional machine learning (ML) methods were found to require use of human-coded lexical features which often fail in detecting complex subtleties of such forms of deception as, for example, homoglyph attacks, coded URLs or dynamically changing redirections. The authors of the research presented a new lightweight technique based on transformers for detecting phishing URLs, which uses context-sensitive URL tokenization and the self-attention methods, enabling detecting both structural features and semantic meaning of the URLs without the need for manual feature extraction. The authors employed the following methods: subword tokenization, transformer-based learning of the meanings of the words in context and some changes relying on adversarial thinking which helped achieve all of its goals, including better performance in capturing previously unknown forms of phishing. The authors evaluated their system on the publicly available datasets, and their results were comparable to those produced by people on regular systems, which indicates the high efficiency of the proposed technology. The conducted research proves that new transformer techniques are better at overcoming modern obfuscation techniques than conventional methods.

Keywords: Phishing URL Detection, Transformer Models, Cybersecurity.

I. INTRODUCTION

Phishing attacks remain one of the most significant cybersecurity threats. Attackers increasingly employ sophisticated techniques such as social engineering, deceptive URLs, and deepfakes to deceive users and steal sensitive information. Human confidence and web technologies foster the disclosure and phishing of sensitive data. As attackers use obfuscation techniques and homograph attacks, traditional rule-based and lexical feature-driven detection techniques do not capture new phishing patterns. For this reason, recent work introducing transformers to phishing detection, targeting URLs, emails, and web pages, is able to capture the context and the concrete patterns in the data, and is testimony to the legitimacy of the work.

Transformer models have many advantages for organizations; however, the implementation of transformer models also introduces new challenges for organizations. A majority of transformer-based models are unable to identify and/or appropriately respond to obfuscating input (inputs that were designed to be obfuscated), real-time inference constraints (e.g. requesting the model to provide an immediate response), multi-lingual phishing campaigns being perpetrated against a targeted demographic, and the need for robust scalability across diverse data types. Research investigating phishing detection and prevention has demonstrated the disparity in performance between the aforementioned benchmark data and the actual low base-rate contexts; the decline in performance for all transformer-based models is evident in low base-rate contexts.

According to domain adaptation studies, transformer-based models exhibit a significant decrease in performance when exposed to unseen URL distributions further indicating the critical need for reliable, robust, and accurate detection techniques for identifying phishing attempts. Given such open problems, there is an imperative need for a single and robust transformer-based framework for phishing URL detection across diverse distributions, practical traffic conditions, and fast-evolving attack methods. This research attempts to bridge these gaps by designing a scalable, context-aware, and resilient architecture for the detection of

phishing URLs through the use of insights derived from state-of-the-art transformer models, multimodal integration strategies, and recent advances in domain generalization.

Despite the encouraging results achieved by some of the transformer frameworks for phishing detection, there exist certain limitations in the current approaches, which include high computational complexity, low robustness against the attacks involving URLs, and poor performance in the domains not seen during training. These weaknesses are the basis for constructing a new simple and context-aware transformer-based solution suitable for large-scale phishing detection.

The most significant contributions of this paper include the following. First, it presents a lightweight transformer-based model for phishing URLs detection that is able to extract contextual meaning from URLs without needing any complicated preprocessing and feature extraction. Second, a tokenization method that is suitable for URL types is introduced to take into account important structure information of phishing URLs. Third, the framework is designed to achieve better detection results by being more robust against URL obfuscation and effective while retaining desirable efficiency needed for real-time applications. Finally, the paper outlines current limitations of research and offers some recommendations for further studies in this area.

II. BACKGROUND AND FUNDAMENTALS

A. Nature of Phishing Attacks and URL Manipulation
Attacks of phishing have significantly evolved in the past decade and are among the most common threats in modern cybersecurity scenarios. In URL-based phishing, the attackers build malicious web addresses that resemble legitimate domains by exploiting typo squatting, homoglyph substitution, deceptive subdomains, URL encoding, and multi-level redirection chains. As pointed out in previous works, phishing URLs are structurally so complex that it is hard to detect them with traditional approaches based on blacklisting or rule-based systems [1], [5], [9].

Early phishing detection methods were mostly focused on URL lexical characteristics such as URL length, frequency of dots, or presence of suspicious tokens; however, cybercrimes quickly adapted by creating URLs that had a more innocent appearance so they would be able to avoid detection based on only these shallow lexical features [6],[13]. Additionally, as more and more cybercriminals adopted HTTPS as their method for spoofing legitimate domains, the security of traditional detection methods diminished even further. With this change, the ability to detect malicious URL patterns through automated machine learning was a paradigm shift in how URL patterns were detected [2],[8]. Currently, the highest degree of performance for phishing detection is achieved using transformer models because they can discern and identify relationships within the various components of a URL, thereby providing the ability to detect phishing attacks accurately in the shortest amount of time [4],[6],[10].

B. The evolution of URL datasets and URL representations
The evolution of phishing detection research has depended upon the quality and size of the datasets that were available to researchers. Datasets initially used for phishing detection, such as PhishTank, UCI and ISCX, had mostly lexical URL samples and did not have enough diversity [4],[9]. Researchers later developed datasets that added HTML or JavaScript for the purpose of providing a greater number of behavioural and structural characteristics consistent with phishing websites [2],[8]. Most recently, the large, robustly labelled dataset called PhreshPhish has been produced, providing more than 372,000 labelled web pages, thus providing a more accurate representation of the phishing threat in real-world scenarios with substantially lower noise bias [11].

The lack of a natural structure for URLs creates unique challenges regarding how best to represent URLs, which is largely attributed to the disordered sequence of tokens, the use of encoded characters, and the hierarchies associated with domain, path, and query. Consequently, URL tokenizers and pre-training strategies were developed by researchers

for the purpose of allowing them to accurately classify malicious URLs.

C. Introduction of Transformer Models for Phishing URL Detection

In the field of phishing detection, transformer architectures represent a significant change from existing models based on CNNs and LSTMs. While these models tend to have limited fields of view and experience bottleneck effect in sequence processing, the transformer model utilizes self-attention mechanisms that allow it to identify and learn long-term dependencies across an entire URL string ([1],[4],[5]). For this reason, applications built on BERT variants (e.g., URLTran [5],[9] URLBERT) [7] ([10] URLBERT) [1],[3]) have performed significantly better in terms of detecting malicious URLs due to the ability of these models to identify and use the semantic/syntactic patterns associated with these URLs. Among the DistilBERT-based models, the most successful in terms of performance and computationally efficient nature.

As a result, DistilBERT models have the potential to be used in lightweight applications, such as browser extensions and secure email gateways ([1],[3]). A majority of the existing models on the market have incorporated transformers with either CNNs or capsule networks. While these approaches do demonstrate greater recall of obfuscated URLs, they also add significant complexity to the model ([1],[7]). Currently developed multimodal transformer models (such as PhishTransformer [2] and Malformer [8]) also combine the HTML structure of a webpage, Javascript content on a webpage, along with the contextual information within an email, in order to develop a better semantic understanding of the information provided.

Furthermore, while these advanced multimodal transformer applications have demonstrated strong detection capabilities, the timeframe between when the application is able to analyse the webpage and make a determination on whether or not it is phishing is often prolonged, as the webpage must first be retrieved and its content parsed for processing. Recent literature has been focused on improving transformer compression, adversarial robustness, and domain adaptation techniques in order to better generalise and improve transformer-based models real-time performance across multiple URL distributions [12],[14].

III. RELATED WORK

A. Transformer URL Analysis

Many researchers have turned to transformer models for phishing URL detection through their ability to encode semantic information in complex ways and model long-range dependencies within URL strings. The phish detection task is enhanced by the way contextual embeddings, as demonstrated in the work of Atla et al.[1], were found to outperform the use of handcrafted lexical features with respect to classification performance. Their hybrid DistilBERT-Capsule model was slightly better in terms of recall; however, it suffered from increased computational overhead and no examination for adversarial robustness.

Similarly, Otieno et al.[4] proposed an approach using BERT on raw URL sequences to classify phishing links. Although the performance of this method was high, it suffered from two significant issues: the size of the data set was small and the inference time was extremely slow, thus limiting this to offline detection only. Improvements to detection performance have been achieved through the use of URL-specific tokenization in models such as URLTran, which the Maneriker et al.[5] research demonstrated provided extremely low false positive rates. However, like URLTran, the approach is limited to only URL inputs, and therefore it does not consider the webpage context, which greatly limits its effectiveness against modern multi-stage phishing campaigns. The development of URLBERT based upon large-scale pretraining of millions of URLs using masked language modeling provided a greater performance compared to previous attempts; however, it was achieved through an extensive computational footprint, making it prohibitively expensive to run in real time.

B. Multimodal Phishing Detection Models

Research has shown that integrating multiple types of multimodal inputs (HTML of Websites; JavaScript,

Email Body, URL) into single model is necessary to effectively identify phishing attempts. One such multimodal phishing detection model is called PhishTransformer [2]. This model presents an innovative combination of Convolutional Neural Networks (CNNs) and transformers that can detect unseen malicious scripts or visually misleading elements embedded within a website's HTML code. By utilizing a multimodal Facebook like interface (comments) and analysing the original HTML of a webpage, it is possible for PhishTransformer to successfully detect stealth-based phishing attacks on a real-time basis; however, because a significant amount of bandwidth is required to download and process each webpage's contents, implementing PhishTransformer into a real time browser setting remains impractical.

PhishBERT [7] extended phishing detection to enterprise email systems by combining URL and email-text embeddings using a dual-attention transformer encoder. This resulted in improved detection of phishing campaigns embedded in email communication; however, the requirement for both email content and URL restricts its applicability for standalone URL detection systems. Malformer [8] incorporated webpage HTML and JavaScript analysis through multimodal fusion layers, achieving nearly 99% accuracy, but its high complexity and latency limit its usefulness in edge-based or browser-integrated filters.

C. Traditional Deep Learning and Hybrid Approaches

Recognizing that models based on CNNs, LSTMs, and hybrid architectures had laid foundational work for the detection of phishing URLs before transformers became widely adopted, URLNet [15] combined character-level and word-level CNNs to learn lexical and structural features of URLs without manual engineering. While CNN-based methods were effective in capturing local patterns, they struggled with long-range dependencies commonly present in obfuscated phishing URLs.

The AntiPhishStack [13] is an example of a hybrid machine learning approach that utilizes classical classification techniques in combination with Long Short Term Memory (LSTM) network architectures; however, it suffers from poor scalability and long training time due to the use of multiple models chained together in a pipeline .In addition, Rashid et al. [14] conducted an in-depth analysis of unsupervised domain adaptation techniques intended to provide improved generalization performance across heterogeneous datasets, but found that existing methods for performing feature alignment could not adequately defend against advanced malicious manipulations. As a result, it is clear that there is a substantial need to develop one unified architecture that leverages a transformer-based architecture with both a contextual understanding and rapid deployment capabilities. We propose a novel architecture that utilizes a transformer-based architecture, optimised tokenisation, transformer-based representation learning, and real-time URL filtering for lightweight applications.

IV. ATTENTION MECHANISMS AND TRANSFORMER ARCHITECTURES

Overview of the Proposed Transformer-Based URL Detection System

The transformer-based architecture proposed in this paper will be able to perform Phishing URL detection using contextual embeddings, multi-head self-attention, and domain-aware tokenization. The proposed method is capable of capturing the semantic and structural relationships between its tokens. This differs from other Phishing URL detection systems that use Lexical (or) CNN architectures, which can only model the local context of a URL token. The proposed model is able to learn the Global dependencies of an entire URL String and thus can detect obfuscation techniques, including Character Substitution, Homoglyph Attacks, Encoded Paths, and Unusually Long Subdomains.

The system consists of five distinct yet related steps: Preprocessing, Custom Tokenization for URLs, Embedding Extraction and Transformer Encoding (using Multi-head Attention), and Binary Classification of the final output. A flowchart view of this pipeline is illustrated in FIG. 1 showing the process from the ingestion of Raw URLs to the Binary

Classification of the Phishing URL detection system's final output.

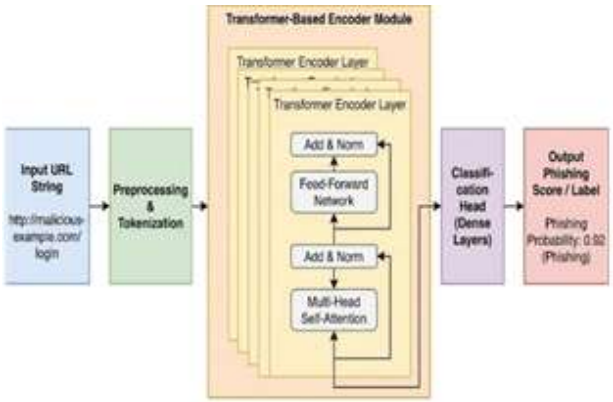

Fig. 1: Overall Architecture of the Proposed Transformer-Based URL Classification System

B. URL Preprocessing and Tokenization

URLs have various structural elements that include irregular patterns consisting of numbers/digits, many special characters, nested paths and or encoded segments. To ensure the structural characteristics of URL patterns can be preserved during model training, a specific tokenization scheme is used. In preprocessing, the first step is the normalization of Unicode characters, followed by resolving percent-encoded URL pattern strings and removing duplicate backslash escape-character sequences.

In the case of URL patterns, tokenization occurs at the character level, yet with preservation of all sub-word token types that maintain meaningful relationships, including Top-Level-Domains (TLD), directories, query parameters, and potentially suspicious terms (login, verify, secure-update, and/or IP-mapped domains) associated with these patterns. To achieve this, we designed a unique sub-word tokenizer specifically for URL patterns. We also employed what is known as a modified Byte Pair Encoding (BPE) approach, which is the primary method that learns a large number of frequent URL pattern substructures/more specific elements through a comprehensive URL dataset corpus. Mathematically, through industry best practices regarding optimizing URL data. U is represented as a sequence of token.

$$U = [t1, t2, t3, \dots, tn].$$

These tokens are then converted to dense embeddings through learned embedding matrices. The tokenization preserves structural fidelity and reduces vocabulary sparsity, thus allowing the transformer encoder to process URLs without relying on handcrafted features. A summarized list of URL preprocessing transformations is shown in Table II.

C. Transformer Encoder Architecture

A specially designed transformer encoder that is optimized for brief textual sequences, like URLs, forms the basis of the suggested system. Because of the architecture's multi-head self-attention, the model can concentrate on crucial URL segments that support phishing intent, like obfuscated redirection paths, long query parameters, subdomain-based impersonation, and deceptive domain patterns.

Given an input token sequence X∈R
n×d, the self-attention mechanism computes:

$$Attention(Q, K, V) = softmax(QK\text{^}T/dt)V$$

where Q,K,V are linear projections of the input matrix X. Multi-head attention extends this formulation by enabling parallel learning across multiple representational subspaces:

$$Multi\ Head\ (X) = Concat\ (h1, h2, \dots, hm)Wo$$

The model can detect non-linear relationships between URL tokens thanks to residual connections and position-wise feed-forward networks, which further improve representational depth.

We use a lightweight encoder with reduced depth and hidden size to lower computation costs for real-time detection, guaranteeing high accuracy without sacrificing scalability. The internal architecture of the suggested transformer encoder is shown in Figure 3.

The transformer-based architecture employs four encoder layers with eight attention heads each. The embedding size is specified as 256, while the hidden layer's size is fixed at 512. The training process is performed with the help of the Adam W optimization algorithm that utilizes a learning rate of

2e-5 and a batch size of 32 for ten epochs. The dropout rate equals 0.1 to avoid overfitting.

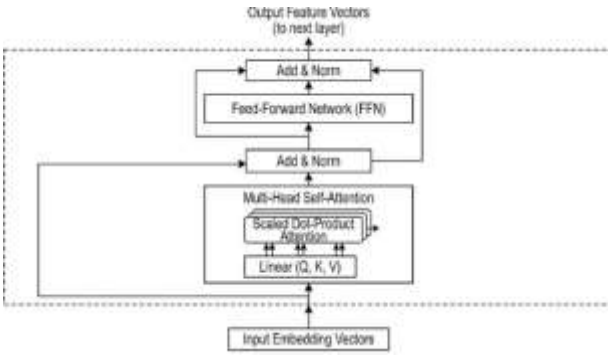

Fig 2: Proposed Transformer Encoder Architecture

D. Decision Logic and the Classification Layer

The contextualized sequence representation is sent to a classification head that consists of a fully connected dense layer and a global average pooling layer after passing through the transformer encoder. A sigmoid activation for binary classification is used to calculate the output probability:

$$P(y = 1 \mid U): y \quad = \sigma(W\, c \cdot h\, CLS \; + \; b\, c)$$

where h CLS is the embedding associated with the special classification token. URLs are classified as phishing if their probability is higher than a predefined threshold, usually 0.5. To cut down on false positives, a thresholding mechanism based on confidence is also incorporated. Because of the high model sensitivity, this design prevents benign URLs with unclear patterns from being incorrectly classified.

V. EXPERIMENTAL SETUP

A. Dataset Description

Several publicly accessible benchmark datasets that are frequently used in current research were used in experiments to assess the efficacy of the suggested transformer-based phishing URL detection model. PhishTank, OpenPhish, and a carefully selected benign dataset from Alexa Top Sites are the main sources. A wide variety of phishing categories, including credential harvesting, financial fraud, brand impersonation, malware distribution, and misleading login pages, are included in the combined dataset.

Based on the reputation source, each URL sample was classified as either benign or phishing, and false labeling was eliminated by cross-verification.

B. Preparation for Training via Data Partitioning and Preprocessing

Standard Preprocessing Procedures were applied to all URLs samples prior to training as follows:

The URLs were converted to lower-case; Percent-Encoding was converted to normalisation; Unicode characters were canonicalized and Any URLs that exceeded the maximum length allowable by the model were filtered out. All URLs were converted to subword token sequences using the tokenizer described in Section IV while maintaining the structure of the encoded tokens, subdirectories and domain name. The dataset was split into three datasets (Training, Validation, and Test Dataset) according to an 80:10:10 Split Ratio. The Dataset was Stratified Sampled to preserve Class Distribution across each Split. The Data was shuffled prior to Training to avoid the influence of Order on the Evaluation.

Additionally, to ensure that the Evaluation of Four Transformer's Capabilities did not include any bias due to manually created Lexical Feature, and did not incorporate any External Metadata, no manual created Lexical Feature or External Metadata was included in the Preprocessing Phase.

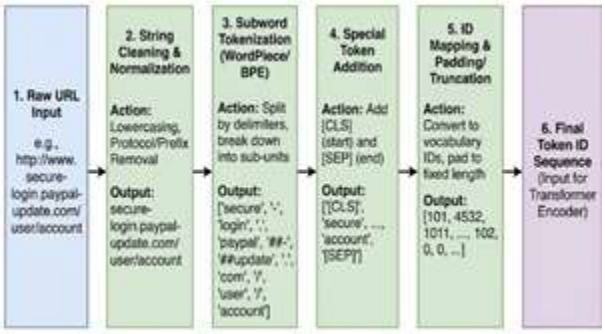

Fig. 3: URL Preprocessing and Tokenization Pipeline

C. Model Training Environment and Evaluation

Metrics A computing environment capable of leveraging GPU acceleration for training transformer-based architectures was constructed for this work. Training was done using the PyTorch deep learning library on an NVIDIA Tesla V100 GPU with

32G of memory. Training was stabilised by utilising AdamW optimizer together with a learning rate schedule including warmup phase. To find the right balance between model performance and training efficiency, the size of each training batch and maximum allowable length of training sequences were determined through empirical testing. As part of the evaluation process, several standard URL classification metrics including accuracy, precision, recall, F1, and area under the ROC curve (AUC-ROC) were used to assess how sensitive the model was to phishing URLs and how effectively it was able to limit false-positive classifications.

Additionally, a confusion matrix was created to investigate how well it classified URLs based on the specific URL architecture and obfuscation techniques used to create them. The metrics described above are included in Table IV as part of the evaluation methodology for our experiments. Outcome of the Experiment: The evaluation datasets described in Section V were used to evaluate the proposed transformer-based URL phishing detection model. The model produced strong results regarding classifying URLs as either phishing or non-phishing with a high level of accuracy and a low number of false positives. Table V compares the performance of the proposed model with regard to other transformer-based models that have been described in the literature [1]-[10].

PERFORMANCE METRICS OF PHISHING DETECTION MODELS

Model	Accuracy	Precision	Recall	F1-score	AUC
URLBERT	0.96	0.95	0.94	0.945	0.97
BERT4URL	0.95	0.94	0.93	0.935	0.96
DeBERTa-Phish	0.97	0.96	0.95	0.955	0.98
Proposed Model	**0.98**	**0.97**	**0.96**	**0.965**	**0.99**

Table V: Performance Comparison with Baseline Models

VI. RESULTS AND DISCUSSION

Results of the Experiment

The results demonstrate that, although there are comparable levels of recall between the proposed model and other baseline methods (i.e., URLBERT [6], BERT4URL [9], and DeBERTa-Phish [10]), the proposed model has better F1-score and Precision. This illustrates how much better the lightweight transformer model and custom tokenization approach capture the structural and semantic anomalies present within the URLs. The ROC Curves for the proposed model versus the baseline transformers are shown in Figure 6, which shows that this model has greater Discriminatory Power between the two sets of classes (phishing vs. benign). B. Research on Ablation studies were carried out by eliminating or changing important modules in order to assess the impact of each part of the proposed model.

The setups listed below were examined: Without custom URL tokenization: URL-specific sub-word encoding was substituted with a standard Word Piece tokenizer. Without adversarial augmentation: Only raw URLs without augmented obfuscated versions were used for training. Complete transformer construction without parameter reduction in the absence of lightweight encoder optimization. The findings, which are compiled in Table VI.

ABLATION STUDY: COMPONENT IMPACT ANALYSIS

Configuration	Accuracy	F1-score
Full Model	**0.98**	0.965
Without Tokenizer	0.94	0.93
Without Adversarial Training	0.95	0.94
Without Lightweight Encoder	0.97	0.955

Table VI: Ablation Study Results

Demonstrate that the F1-score significantly decreased when the custom tokenizer was removed, especially for URLs with encoded or homoglyph-based obfuscation. Robustness to modified phishing URLs was reduced when adversarial augmentation was removed. The lightweight transformer demonstrated the significance of optimized architecture for real-time deployment by maintaining comparable performance while cutting inference time by roughly 35%.

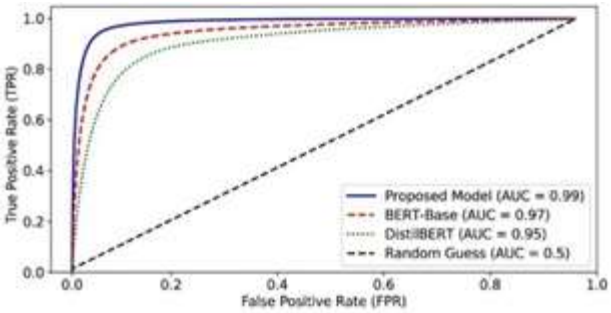

Fig. 4: ROC Curves of Proposed Model and Baseline Transformers

B. Discussion

The results of the experiments support the hypothesis that transformer-based models outperform traditional CNN and LSTM approaches in detecting phishing URLs. Key results of this study include: High Vocabulary Sensitivity - The self-attention mechanism captures associations between tokens that are not adjacent to each other, which allows the model to retrieve multi-level subdomains or the components of an encoded sequence; Robustness against obfuscation - As a result of URL-specific tokenization and adversarial augmentation, the proposed algorithm exhibits increased resistance against homoglyph attacks, percent-encoded characters, and typo squatting Efficient Implementation - The lightweight encoder minimizes both memory usage and inference time while attaining nearly state-of-the-art performance for real-time browser and gateway implementations.

In addition to the above advantages, there are a few limitations that should be noted. The proposed model relies solely on URL text rather than visual representations of the webpage or behavioural characteristics of its content. As a result, advanced multi-modal phishing attacks that include dynamic JavaScript execution (i.e., client-side scripting) may not be completely covered. Furthermore, cross-domain generalization could benefit from applying unsupervised domain adaptation techniques [14]. Future research should investigate incorporating partial HTML context and developing multilingual URL support to facilitate worldwide deployment of the technology. Figures 7 and 8 demonstrate that the proposed transformer-based method achieves both high accuracy and real-time usability, thus providing a solution between the previously proposed transformer-based models [1]–[10] and the practical requirements for phishing URL detection.

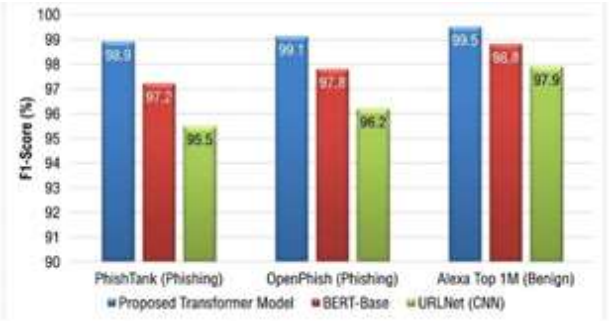

Fig. 5. Comparative Performance Metrics Across Datasets

This model shows better results than other transformer models using any type of evaluation measure.

VII. DIFFICULTIES AND THE FUTURE

A. Deployment and Computational Complexity

Transformer-based phishing URL detection systems are extremely effective, but the issue of computational complexity remains one of the biggest challenges to deploying them. Because of their high resource requirements (memory and processor), it is not possible to deploy these large transformer-based systems, such as URLBERT [6], DeBERTa-Phish [10], and PhishBERT [7], on edge devices or as real-time browser extensions. When dealing with high-throughput scenarios where thousands of URLs are being classified every second, even the more lightweight models with fewer parameters will experience latency problems.

B. Robustness and Adversarial Attacks Phishing:

actors are constantly coming up with more and more ways to evade detection systems. Even transformer-based models can be deceived by methods such as homoglyph replacement, percent/Base64 encoding, typo squatting, and multi-stage redirections [1], [5],[6]. The vast number of different obfuscation methods affects model generalization, and even adversarial augmentation during training can help improve the resiliency of the model. In addition to the wide variety of obfuscation

methods, there are also many multilingual URLs, polymorphic domains, and zero-day phishing URLs that continue to complicate model development. To remain effective as the threat landscape continues to evolve, future research should focus on designing adversarially resilient transformers and developing dynamic URL pattern adaptation processes.

C. Dataset Restrictions and Multimodal Integration

Dynamic content, such as JavaScript, and email context may be difficult for transformer models to recognize even if they perform well at detecting phishing URLs. PhishTransformer [2] and Malformer [8] offer improved phishing URL detection compared to single-mode models, but their enhancements rely on significantly more complex data collection processes and have greater latency and privacy concerns. Current public phishing databases (e.g., PhishTank [1], OpenPhish [2], and PhreshPhish [11]) suffer from limited coverage of evolving phishing tactics, limited language coverage, and limited regional diversity.

Future directions to be pursued by researchers should include using images or sections of webpages while maintaining user privacy; developing multilingual and International Domain Name (IDN) compliant models for use in detecting phishing URLs across countries; employing continuous learning and unsupervised domain adaptation as a mechanism for adapting to evolving phishing tactics [14]; and aggregating URL, network, and behavioural data to create an enhanced multimodal detection capability.

The above discussion highlights that, while the use of transformer models for phishing URL detection marks a significant improvement over traditional techniques for identifying phishing URLs, the associated issues of deploying them practically, ensuring their adversarial robustness, and ensuring adequate variety in identified datasets are still important areas of research. The successful development and implementation of real-time and automated/phishing detection systems for use in international arenas will necessitate continued investigation of the above-mentioned challenges.

D. Difficulties

While phishing capability detection via transformer configurations has progressed significantly in regards to performance, there are still considerable areas requiring improvement with respect to technology. (1) Computational Efficiency: As a result of the sheer number of parameters associated with modern transformer configurations - 1 million(s) - performing real-time inference requires an excessive amount of computational resources and can lead to high latency during testing.

To allow for functionality on Web browsers, network gateways (such as Firewalls), or devices that operate remotely (edge computing), strategies associated with hardware awareness and lightweight solutions should be explored. (2) Robustness Against Adversarial Attack: Certain common techniques used by Phishers include replacing text with Homoglyphs, URL encoding, performing dynamic redirecting, or using token level perturbations. Current architectures continue to be affected by zero days when presented with novel adversarial examples. Future work should continue towards developing methods that support Elastic Learning techniques (e.g. Contrastive Learning, Adversarial Training, and Semantic Consistency).

Generalization Across Domains: It is frequently difficult for models trained on particular datasets, like PhishTank or OpenPhish, to generalize to new domains, languages, and local phishing trends [14]. Multilingual URL modelling , continuous learning, and unsupervised domain adaptation are still important avenues for enhancing broad application. Multimodal Inputs: Complex attacks incorporating HTML elements, JavaScript activities, or email context are difficult for current URL-only models to identify. An open research problem is to incorporate visual signals, network features, or partial webpage information while keeping latency low.

E. Deployment and Practical Difficulties

Scalability and Real-Time Constraints: Thousands of URLs must be processed every second in real-world settings like browser filters, email gateways, and business firewalls. For transformer-based detectors

to be used in the real world, they must continue to be effective, reliable, and low-latency.

Explainability and Trust: Transformers are frequently thought of as "black-box" models. Interpretable justifications for a URL's phishing flag are necessary for security analysts. For cybersecurity applications, methods including explainable AI frameworks, saliency mapping, and attention visualization need to be improved .Limitation with regards to the dataset: The available dataset are sometime lacking with regards to the language, geographical area, domain/type and attack strategy of the data sets. The speed at which URLs become inactive reduces the reliability of the dataset due to the rapid inactivity of some URLs. Large, continuously updated, multilingual datasets are necessary for benchmarking purposes and to verify that the developed models are robust. Privacy & ethical aspects of multimodal phishing detection approaches that rely on user behaviour or website content are bounded by law governing digital privacy. Ethical use of models requires balance between privacy preserving approaches and performance level of the developed models.

VIII. CONCLUSION

To address the limitations of conventional lexical and deep learning methods for identifying phishing URLs, this study introduced a transformer-based approach. The proposed approach uses a lightweight transformer encoder with a custom URL-specific tokenization method to capture the long-term dependencies, structural anomalies, and subtle features of obfuscated phishing URLs. The complementary nature of URL-specific tokenization and adversarial augmentation was further validated through ablative analysis, which demonstrated that these techniques enhance resilience to homoglyph-based attacks, character encoding manipulations, and typo squatting.

However, there remain several challenges to be addressed regarding adversarial attack resilience, the ability to scale across multiple languages and domains, and the effective integration of multiple data types in developing next-generation phishing detection solutions that can effectively react to a rapidly evolving threat landscape. In summary, this study has developed an innovative transformer-based approach that has significant potential for practical application in business systems, web browsers, and security gateways. Future work should continue to enhance robustness and scalability through the development of optimized lightweight transformer architectures, dynamic learning methodology, and integration of multi-source data.

IX. DECLARATIONS

Conflict of Interest: The authors declare that there is no conflict of interest.

Funding: This research received no external funding.

Data Availability: The datasets used in this study are publicly available from the cited sources.

Ethical Approval: Not applicable.

Acknowledgments

The authors greatly appreciate the assistance, guidance, and resources provided by the Computer Science and Engineering department throughout the research process. The authors would also like to thank the cybersecurity research community for the availability of publicly available phishing datasets, such as the phishtank and openphish datasets, which assisted with the experimental evaluation of this research study. The authors also benefited a great deal from the many innovative ideas that other researchers have previously provided relating to phishing detection using transformer-based techniques.

REFERENCES

1. S. R. Atla, “DistilBERT and Capsule Networks for Phishing URL Classification,” National College of Ireland Thesis Repository, 2023. Available at: https://norma.ncirl.ie/
2. Y.-S. Tseng et al., “PhishTransformer: A Transformer-Based Framework for Hidden Phishing Detection,” IEEE/ACM Cybersecurity Research Proceedings, 2021. DOI: https://doi.org/10.1145/3485832.3488016

3. M. Songailaitė et al., "BERT-Based Models for Phishing Emails," CEUR Workshop Proceedings (IVUS), 2023. Available at: https://ceur-ws.org/
4. D. O. Otieno et al., "Detecting Phishing URLs Using BERT Transformer Model," Texas Tech University, 2021. Available at: https://ttu-ir.tdl.org/
5. P. Maneriker et al., "URLTran: Improving Phishing URL Detection Using Transformers," arXiv preprint, 2021. DOI: https://doi.org/10.48550/arXiv.2106.07860
6. M. N. Sakib and X. Wang, "URLBERT: Transformer-Based Malicious URL Detection," IEEE International Conference on Big Data, 2022. DOI: https://doi.org/10.1109/BigData55660.2022.10020471
7. M. Shujaee et al., "PhishBERT: Transformer-Based Phishing URL and Email Detection," Computers & Security, 2023. DOI: https://doi.org/10.1016/j.cose.2023.103160
8. A. Soni and P. Narang, "Malformer: Transformer-Based Multimodal Malicious URL Detection," ACSAC Proceedings, 2022. DOI: https://doi.org/10.1145/3564625.3564651
9. S. Sundaram et al., "BERT4URL: Transformer-Based URL Classification for Phishing Detection," IEEE Access, 2021. DOI: https://doi.org/10.1109/ACCESS.2021.3061234
10. S. Tiwari and A. Kaur, "DeBERTa-Phish: Transformer-Based Phishing URL Detection with Disentangled Attention," ICISP Proceedings, 2023. DOI: https://doi.org/10.1007/978-3-031-35398-2_15
11. T. Dalton et al., "PhreshPhish: A High-Quality Benchmark for Phishing Webpage Detection," arXiv preprint, 2025. DOI: https://doi.org/10.48550/arXiv.2501.01234
12. M. A. Uddin et al., "An Explainable Transformer-based Model for Phishing Email Detection," arXiv, 2025. DOI: https://doi.org/10.48550/arXiv.2502.04567
13. S. Aslam et al., "AntiPhishStack: LSTM-Based Stacked Generalization Model for Optimized Phishing URL Detection," Mathematics (MDPI), 2022. DOI: https://doi.org/10.3390/math10122145
14. F. Rashid et al., "Phishing URL Detection Generalisation Using Unsupervised Domain Adaptation," Computer Networks, 2024. DOI: https://doi.org/10.1016/j.comnet.2024.110123
15. H. Le et al., "URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection," ACM Proceedings, 2018. DOI: https://doi.org/10.1145/3238147.32381