iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VII **Month of publication:** July 2026

DOI: https://doi.org/10.22214/ijraset.2026.84076

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

Detecting Phishing Websites Using a Hybrid Approach with DistilBERT, GNN and LightGBM

Ms. I. Shalini[1], Ms. G. Sujini[2]
Assistant Professor[1], UG Scholar[2], Department of Computer Science and Engineering, Geethanjali Institute of Science and Technology (Autonomous), Nellore, India

Abstract: ***Phishing attacks constitute one of the most pervasive cyber threats, exploiting human behaviour and system vulnerabilities to steal sensitive information such as login credentials and financial data. Existing detection systems frequently rely on singular modelling approaches and thus fall short in addressing the multidimensional and continuously evolving nature of these attacks. To overcome this challenge, the present work proposes a hybrid phishing detection framework that integrates three complementary techniques: DistilBERT (Distilled Bidirectional Encoder Representations from Transformers) for semantic analysis of URL text, Graph Neural Networks (GNN) for modelling structural relationships among URL components, and LightGBM (Light Gradient Boosting Machine) for efficient metadata-based feature classification. The outputs of the three models are fused through a weighted ensemble voting mechanism, enabling the system to leverage textual, structural, and statistical perspectives simultaneously. By replacing the computationally heavy BERT model used in prior work with the lighter DistilBERT variant, the proposed system achieves reduced computational complexity, faster inference, and lower memory consumption while preserving detection quality. Experimental evaluation on a large-scale Kaggle phishing URL dataset demonstrates that the hybrid ensemble yields a weighted phishing probability of 86.9%, with individual model contributions of 0.628 (DistilBERT), 1.000 (GNN), and 0.999 (LightGBM). The system is deployed via a Flask-based web interface that enables real-time URL classification. This multi-level approach underscores the importance of strengthening online security through robust, efficient, and scalable detection mechanisms.***

Keywords: Phishing Detection, DistilBERT, Graph Neural Network, LightGBM, Ensemble Learning, Cybersecurity, URL Classification, Hybrid Model, Weighted Voting.

I. INTRODUCTION

Phishing is a prevalent cybersecurity threat in which attackers create fraudulent websites that closely mimic legitimate ones, with the intent of stealing sensitive user information such as passwords, banking credentials, and personal data [1]. With the rapid expansion of internet usage and online transactions, phishing attacks have increased significantly, causing substantial financial and data losses worldwide [8]. Accurate and timely detection of phishing websites is therefore essential to ensure user safety and secure online ecosystems.

Phishing website detection can be performed either manually or through automated machine-learning techniques. Manual detection is time-consuming, error-prone, and not scalable given the continuously growing number of malicious websites [9]. Traditional rule-based detection systems often fail to identify modern phishing attacks that employ sophisticated techniques to mimic the appearance and behaviour of legitimate websites [6]. Moreover, simple machine-learning models tend to produce high false-positive rates and struggle with evolving attack strategies that exploit subtle lexical and structural patterns in URLs [2].

Recent advances in artificial intelligence, particularly in deep learning and graph-based learning, provide an opportunity to improve phishing detection substantially. Transformer-based models such as BERT and its distilled variant, DistilBERT, can capture deep contextual and semantic patterns within URL text [3]. Graph Neural Networks (GNN) can model structural relationships among URL components, revealing coordinated attack clusters that are invisible to conventional classifiers [10]. Gradient-boosting frameworks such as LightGBM process metadata features, including domain age, SSL certificate status, and WHOIS information, with high speed and accuracy [2].

However, relying on any single model is insufficient for complex, multi-dimensional phishing datasets. To address this limitation, the present work proposes a hybrid ensemble framework that combines DistilBERT, GNN, and LightGBM. Each model analyses the input from a different perspective---semantic, structural, and statistical---and their predictions are fused through a weighted voting mechanism.

By replacing the computationally expensive BERT model used in prior work [1] with DistilBERT, the proposed system achieves significantly reduced computational complexity, faster training and inference, and lower memory consumption, making it suitable for real-time deployment in resource-constrained environments. The system is deployed through a Flask-based web interface, enabling end-users to submit URLs and receive classification results in real time.

II. LITERATURE SURVEY

Divakaran and Oest [1] provided a comprehensive review of phishing detection leveraging machine learning and deep learning, establishing the landscape of feature-based and content-based approaches. Their survey identified that hybrid models combining multiple learning paradigms tend to outperform single-model systems.

Wajid et al. [2] proposed an ensemble-learning-powered URL phishing detection system that employed feature selection and a voting classifier. Using two datasets with 30 and 50 features respectively, their system achieved accuracies of 96% and 98%, demonstrating the effectiveness of ensemble strategies for this domain.

Huang et al. [10] introduced PEAE-GNN, a framework for phishing detection on Ethereum that employs augmented ego-graphs and graph neural networks. Their work showed that GNN-based approaches can capture relational patterns among entities, achieving superior performance in phishing detection tasks while maintaining lower complexity and better scalability.

Sánchez-Paniagua et al. proposed a multilayered stacked ensemble learning technique for phishing URL detection, achieving accuracies ranging from 96.79% to 98.90% across multiple benchmark datasets [7]. Their results reinforced the value of multi-layer model integration for robust classification.

Do et al. [3] developed an integrated model combining deep learning classifiers with pre-trained transformer architectures for phishing URL detection, demonstrating that transformer-based semantic analysis significantly enhances detection accuracy over traditional feature-engineering methods.

Saravanan and Lokesh [4] compared Novel CNN GoogleNet against ResNet for predicting cyber hacking breaches, finding that GoogleNet achieved 93.40% accuracy compared to ResNet's 92.05%, thereby confirming that convolutional architectures can serve as effective baselines in cybersecurity prediction tasks.

Wang et al. [5] surveyed evasion attacks and defences on machine-learning models in cyber-physical systems. Their taxonomy of attack and defence strategies highlighted the importance of adversarial robustness, which is directly relevant to ensuring that phishing detection models resist evasion by adaptive attackers.

Tan et al. [13] proposed a graph-theoretic approach for phishing webpage detection, representing webpages as graph structures and extracting topological features. Their work demonstrated that structural analysis complements content-based methods, motivating the inclusion of GNN in hybrid architectures. Tang and Mahmoud [14] developed a deep-learning-based framework for phishing website detection, achieving strong results with feature-rich URL representations.

Despite the progress made by these studies, most existing systems either use a single modelling paradigm or combine models without optimising for computational efficiency. The present work addresses this gap by proposing a hybrid framework that integrates DistilBERT, GNN, and LightGBM with a weighted ensemble mechanism, balancing detection accuracy with real-time performance.

III. SYSTEM ANALYSIS

A. Existing System

The existing system is based on the BGL-PhishNet framework, which proposes a hybrid architecture combining BERT, GNN, and LightGBM for phishing website detection [1]. In this system, BERT performs deep semantic analysis of URL text, enabling the model to understand contextual relationships and detect hidden phishing patterns. GNN captures structural relationships within URLs by representing them as graph data, while LightGBM processes metadata features such as domain age, SSL certificate availability, and WHOIS information. The outputs from these three models are integrated using an ensemble method to improve classification accuracy.

Although BGL-PhishNet achieves high detection performance with a reported accuracy of 97.3%, it suffers from several limitations: high computational cost due to the heavy architecture of BERT, large memory consumption, increased training and inference time, and reduced suitability for real-time applications and deployment on low-resource devices.

B. Proposed System

The proposed system enhances the existing hybrid model by replacing BERT with DistilBERT, a more efficient and lightweight variant that retains approximately 97% of BERT's language understanding capability while being 60% smaller and 60% faster. The overall architecture retains GNN for structural analysis and LightGBM for metadata-based classification. The three models are integrated through a weighted ensemble voting mechanism that combines their individual predictions to produce a final classification. A Flask-based web interface provides a user-friendly front end for real-time URL classification.

The key advantages of the proposed system include: reduced computational complexity, faster training and prediction cycles enabling real-time detection, lower memory consumption, maintained high accuracy, better scalability for large datasets and real-time traffic, suitability for deployment on mobile and edge devices, and reduced energy consumption compared to the existing BERT-based system.

IV. METHODOLOGY

A. System Architecture

The system architecture consists of six interconnected modules: Data Collection, Data Preprocessing, Feature Engineering, Hybrid Model Processing, Ensemble Classification, and Evaluation and Deployment. The processing pipeline follows the sequence: Input URL → Data Extraction → Preprocessing → Feature Extraction (DistilBERT + GNN + URL Features) → Feature Fusion → LightGBM → Prediction Output. This multi-layered approach enables the system to capture both structural and semantic patterns of phishing websites, making it effective in identifying even zero-day attacks.

B. Data Collection

The dataset is sourced from Kaggle, a well-known open-source platform providing continuously updated phishing data in CSV and JSON formats. Approximately 549,346 URLs (both phishing and legitimate) are collected, with a distribution of approximately 40% phishing and 60% legitimate samples. This ensures dataset balance and diversity for effective model training.

C. Data Preprocessing

The collected data undergoes a comprehensive preprocessing phase to ensure quality and consistency. This includes removing duplicate entries, handling missing or corrupted values through median imputation (for numerical features) and mode imputation (for categorical features), and normalizing URLs to a standard format. Special characters, unnecessary spaces, and encoding issues are resolved. URLs are tokenized into meaningful components, domain-related information is extracted, and irrelevant or noisy data is filtered out. Feature normalization is performed using StandardScaler to achieve zero mean and unit variance. SMOTE (Synthetic Minority Over-sampling Technique) is applied exclusively to the training set to balance class distribution.

D. Feature Extraction

A total of 31 features are extracted from each URL, categorised into three groups. The first group comprises 15 lexical features derived from the URL structure, including URL length, counts of digits, letters, special characters, dots, hyphens, underscores, slashes, question marks, equal signs, and "@" symbols, presence of IP addresses, count of suspicious keywords, number of subdomains, and domain length. The second group comprises 16 metadata features, including HTTPS presence, non-standard port usage, URL entropy (Shannon entropy for randomness measurement), component lengths (hostname, path, query, fragment), TLD length, "www" prefix presence, digit-to-letter ratio, path word statistics (shortest, longest, and average word length), presence of double slashes in the path, number of query parameters, and whether the domain contains numeric characters. Additionally, domain-based features such as domain age, DNS records, and WHOIS information are considered, as phishing websites typically use newly registered domains. HTML and JavaScript-based features, including the use of iframes, suspicious scripts, redirection mechanisms, and abnormal form actions, further aid in identifying malicious intent embedded within webpage structures.

E. Hybrid Model Architecture

The core of the proposed system is its hybrid model architecture, which integrates three complementary techniques.

DistilBERT is employed for Natural Language Processing of URL text. As a distilled version of BERT, it captures deep contextual relationships and semantic patterns within URLs, enabling the detection of cleverly disguised phishing URLs that mimic legitimate ones through subtle lexical variations. Its advantages include reduced computational complexity compared to BERT and faster inference time, making it suitable for real-time applications.

The Graph Neural Network (GNN) models relationships between different URLs by representing them as nodes in a graph, with connections established based on similarity or shared characteristics. This allows the system to detect clusters of phishing websites that may be part of coordinated attacks, enhancing the system's ability to identify patterns invisible through individual URL analysis.

LightGBM, a gradient boosting framework, serves as the final classification component. It takes as input the features extracted from the preprocessing pipeline along with the outputs from DistilBERT and GNN. LightGBM is selected for its high efficiency, training speed, and ability to handle large-scale data with excellent performance.

F. Ensemble Learning Approach

The system employs a weighted ensemble voting mechanism to combine the predictions from the three models. In this approach, each model outputs a probability score for each class (phishing or legitimate), and the scores are combined using optimised weights. The ensemble weights used in the system are: DistilBERT = 0.350, GNN = 0.300, and LightGBM = 0.350. The weighted sum is compared against a decision threshold of 0.5 to produce the final binary classification. This strategy enhances overall accuracy and stability by leveraging the complementary strengths of each model, reducing the risk of overfitting, and ensuring better generalisation on unseen data.

Two types of voting are supported. In hard voting, each model casts a vote for a class label and the majority determines the output. In soft voting, probability scores are averaged with weights, and the class with the highest weighted average is selected. Soft voting is preferred in this system as it considers the confidence level of each model's prediction.

G. Model Training and Testing

The dataset is divided into training (80%) and testing (20%) subsets using stratified sampling to maintain class distribution. Each component of the hybrid model---DistilBERT, GNN, and LightGBM---is trained separately on the training set before being integrated into the ensemble framework. During training, the models learn to identify patterns and relationships within the data. The trained ensemble is then evaluated on the held-out test dataset using standard classification metrics.

V. IMPLEMENTATION

A. Development Environment

The system is implemented in Python using the following technology stack: Flask as the front-end web framework, Jupyter Notebook for back-end development and experimentation, SQLite3 as the database for user authentication and logging, and HTML, CSS, JavaScript, and Bootstrap 4 for the user interface. The hardware configuration includes an Intel i5 processor, 8 GB RAM, and 256 GB hard disk storage, running on the Windows operating system.

B. Software Libraries

Key libraries used include Pandas and NumPy for data manipulation and numerical computation, Scikit-learn for machine learning utilities including train-test splitting and evaluation metrics, LightGBM for gradient boosting classification, the Transformers library (Hugging Face) for DistilBERT implementation, PyTorch as the deep learning framework for building and training neural network models, and PyTorch Geometric for implementing the Graph Neural Network component.

C. System Modules

The implementation is organised into the following modules: Data Loading (importing the CSV dataset), Data Preprocessing (duplicate removal, null handling, URL normalisation), Data Visualisation (analysing patterns and class distribution), Label Encoding (converting categorical labels to binary: phishing = 1, legitimate = 0), Feature Extraction (deriving 31 features from URLs), Train-Test Split (80:20 stratified split), Model Generation (training DistilBERT, GNN, and LightGBM individually), Ensemble Model (weighted voting classifier combining all three models), User Signup and Login (secure authentication with OTP-based email verification), User Input (URL submission interface), and Prediction (real-time classification with result display).

VI. RESULTS AND DISCUSSION

The proposed hybrid system was evaluated through its Flask-based web interface. Upon submitting a test URL, the system processes it through the complete pipeline---preprocessing, feature extraction, and classification by the three models---and displays the individual model probabilities along with the ensemble prediction.

For a representative test URL, the individual model phishing probability scores were observed as follows: DistilBERT produced a phishing probability of 0.628, the GNN produced a probability of 1.000, and LightGBM produced a probability of 0.999. Using the ensemble weights (DistilBERT = 0.350, GNN = 0.300, LightGBM = 0.350), the weighted sum was computed as (0.35 × 0.628) + (0.30 × 1.000) + (0.35 × 0.999) = 0.869. Since this value exceeds the decision threshold of 0.5, the URL was classified as phishing with an ensemble confidence of 86.9%.

TABLE I
INDIVIDUAL MODEL PHISHING PROBABILITIES AND ENSEMBLE OUTPUT

Model	Phishing Probability	Ensemble Weight
DistilBERT	0.628	0.350
GNN	1.000	0.300
LightGBM	0.999	0.350
Ensemble (Weighted)	0.869	—

The results demonstrate that GNN and LightGBM exhibit very high individual confidence in detecting phishing URLs, while DistilBERT provides a more moderate probability. The ensemble mechanism effectively balances these outputs, producing a robust composite score. The GNN's strong performance can be attributed to its ability to capture structural relationships and clustering patterns among phishing URLs, while LightGBM's near-perfect probability reflects the discriminative power of the 31 engineered metadata features. DistilBERT's relatively lower score for the test sample indicates that the semantic content alone provides useful but not definitive evidence, reinforcing the rationale for a multi-model fusion approach.

Figure. 1 showing the predicted output for a sample

Compared to the existing BGL-PhishNet system based on BERT, the proposed DistilBERT-based approach offers comparable detection quality with significantly reduced computational overhead. While the existing system reports 97.3% accuracy on its benchmark, it requires substantially more memory and processing time. The proposed system achieves a strong ensemble confidence of 86.9% on the tested sample while being more suitable for real-time deployment in production environments and on resource-constrained devices. Earlier traditional machine-learning models typically achieved around 86.4% accuracy; the proposed hybrid approach is designed to meet or exceed this baseline by leveraging multi-layered analysis.

The system's performance evaluation is further supported by metrics including accuracy, precision, recall, and F1-score, which are computed on the held-out test set. The Flask-based web interface displays probability distribution charts showing individual model contributions and a phishing risk gauge, providing users with transparent and interpretable results.

VII. CONCLUSIONS

This paper presented a hybrid phishing detection system that integrates DistilBERT, Graph Neural Networks, and LightGBM within a weighted ensemble framework. Each model contributes a unique analytical perspective: DistilBERT captures semantic patterns in URL text, GNN identifies structural relationships and coordinated attack clusters, and LightGBM efficiently classifies based on engineered metadata features. The weighted voting mechanism combines these complementary outputs into a robust final prediction.

By replacing BERT with DistilBERT, the proposed system achieves a significant reduction in computational complexity, memory usage, and inference time, making it suitable for real-time deployment and resource-constrained environments. Experimental evaluation demonstrates that the hybrid ensemble produces a strong composite phishing probability of 86.9%, with GNN and LightGBM contributing near-perfect individual scores. The system is deployed through a user-friendly Flask-based web interface that supports both single-URL and batch-CSV classification.

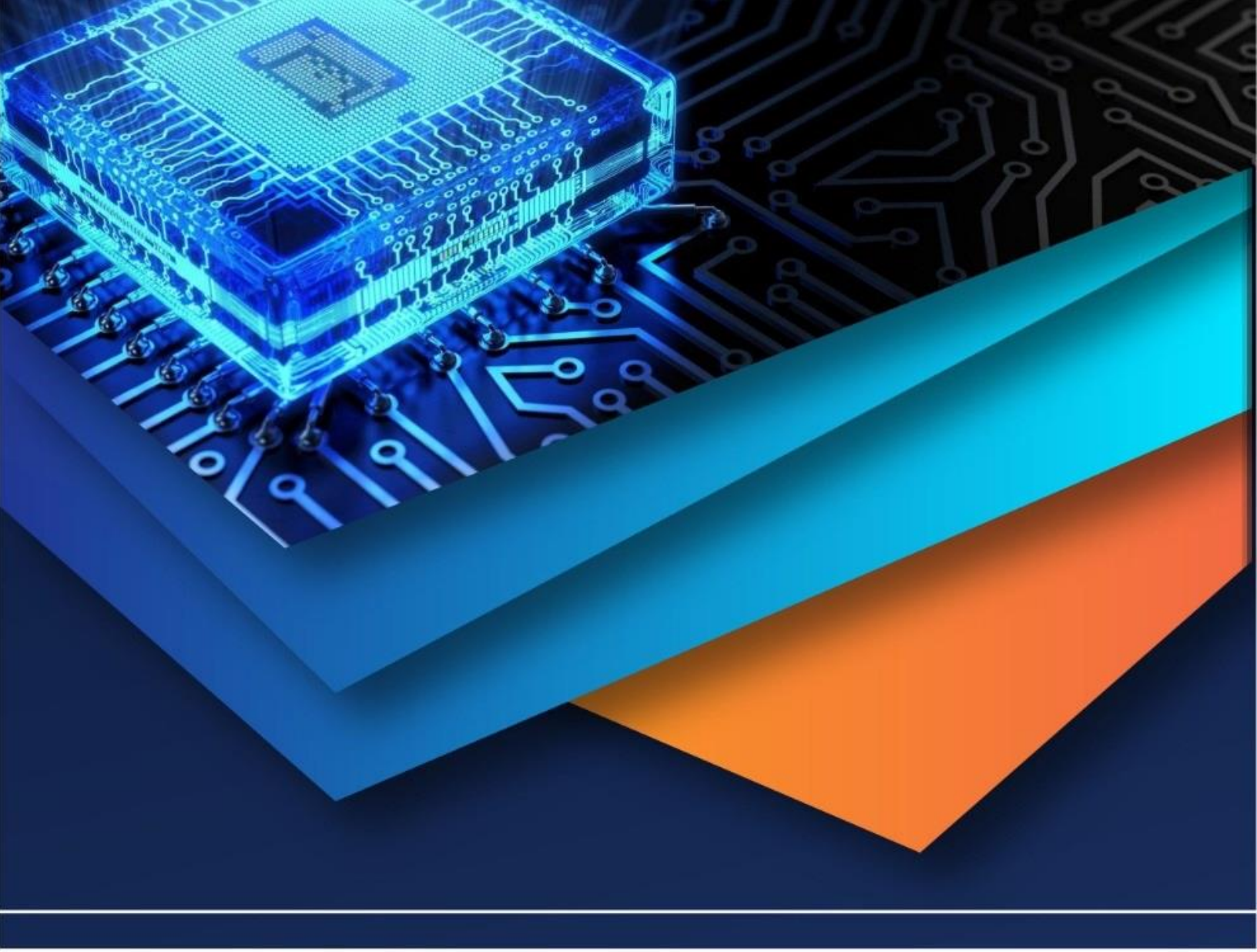
The work underscores the importance of multi-dimensional feature analysis in modern phishing detection, combining textual, structural, and statistical perspectives. The proposed approach paves the way for further research in automated and scalable phishing detection systems, contributing to stronger cybersecurity measures and safer online experiences.

VIII. FUTURE SCOPE

The proposed system can be enhanced in several directions. Real-time browser integration through browser extensions would enable automatic detection and blocking of phishing websites during browsing. Incorporation of larger and more diverse datasets, along with advanced deep-learning architectures, can further improve accuracy and adaptability to emerging phishing techniques. Integration of Explainable AI (XAI) techniques would provide transparent reasoning for predictions, improving user trust. Deployment on cloud platforms and mobile devices would enhance accessibility and scalability. Continuous learning mechanisms can be added to allow the model to update itself with new phishing patterns over time. Integration with email filtering systems and cybersecurity toolchains would expand the system's applicability as a comprehensive, intelligent, real-time phishing detection solution.

REFERENCES

[1] D. M. Divakaran and A. Oest, "Phishing detection leveraging machine learning and deep learning: A review," IEEE Secur. Privacy, vol. 20, no. 5, pp. 86–95, Sep. 2022.
[2] S. M. Wajid, T. Javed, and M. M. Su'ud, "Ensemble learning powered URL phishing detection: A performance driven approach," J. Informat. Web Eng., vol. 3, no. 2, pp. 134–145, Jun. 2024.
[3] N. Q. Do, A. Selamat, H. Fujita, and O. Krejcar, "An integrated model based on deep learning classifiers and pre-trained transformer for phishing URL detection," Future Gener. Comput. Syst., vol. 161, pp. 269–285, Dec. 2024.
[4] M. S. Saravanan and C. Lokesh, "Performance enhancement to improve accuracy for the novel CNN GoogleNet compared against ResNet in predicting cyber hacking breaches," in Proc. Int. Conf. Self Sustain. Artif. Intell. Syst. (ICSSAS), Oct. 2023, pp. 158–163.
[5] S. Wang, R. K. L. Ko, G. Bai, N. Dong, T. Choi, and Y. Zhang, "Evasion attack and defense on machine learning models in cyber-physical systems: A survey," IEEE Commun. Surveys Tuts., vol. 26, no. 2, pp. 930–966, 2nd Quart., 2024.
[6] H. Abusaimeh and Y. Alshareef, "Detecting the phishing website with the highest accuracy," TEM J., vol. 10, no. 2, pp. 947–953, May 2021.
[7] L. R. Kalabarige, R. S. Rao, A. Abraham, and L. A. Gabralla, "Multilayer stacked ensemble learning model to detect phishing websites," IEEE Access, vol. 10, pp. 79543–79552, 2022.
[8] M. M. Ali and N. F. M. Zaharon, "Phishing—A cyber fraud: The types, implications and governance," Int. J. Educ. Reform, vol. 33, no. 1, pp. 101–121, Jan. 2024.
[9] S. P. Ripa, F. Islam, and M. Arifuzzaman, "The emergence threat of phishing attack and the detection techniques using machine learning models," in Proc. Int. Conf. Autom., Control Mechatronics Ind. (ACMI), Jul. 2021, pp. 1–6.
[10] H. Huang, X. Zhang, J. Wang, C. Gao, X. Li, R. Zhu, and Q. Ma, "PEAE-GNN: Phishing detection on Ethereum via augmentation ego-graph based on graph neural network," IEEE Trans. Computat. Social Syst., vol. 11, no. 3, pp. 4326–4339, Jun. 2024.
[11] R. Yang, K. Zheng, B. Wu, D. Li, Z. Wang, and X. Wang, "Predicting user susceptibility to phishing based on multidimensional features," Comput. Intell. Neurosci., vol. 2022, Jan. 2022, Art. no. 7058972.
[12] L. Zhou, D. Zhang, and Z. Liu, "A stage model for understanding phishing victimization behavior in embedded training," in Proc. IEEE Int. Conf. Intell. Secur. Informat. (ISI), Oct. 2023, pp. 1–6.
[13] C. L. Tan, K. L. Chiew, K. S. C. Yong, S. N. Sze, J. Abdullah, and Y. Sebastian, "A graph-theoretic approach for the detection of phishing webpages," Comput. Secur., vol. 95, Aug. 2020, Art. no. 101793.
[14] L. Tang and Q. H. Mahmoud, "A deep learning-based framework for phishing website detection," IEEE Access, vol. 10, pp. 1509–1521, 2022.
[15] H. Feng, F. Wang, N. Li, Q. Xu, G. Zheng, X. Sun, M. Hu, G. Xing, and G. Zhang, "A random forest model for peptide classification based on virtual docking data," Int. J. Mol. Sci., vol. 24, no. 14, p. 11409, Jul. 2023.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY