IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

# PHISHING DETECTION USING MACHINE LEARNING

**[1]Het Raghuvanshi, [2]Isha Prajapati**

[1]Lecturer,|[2]Lecturer

Department of Information Technology

Drs. Kiran & Pallavi Patel Global University, Vadodara, Gujarat, India

**Abstract:** Phishing attacks persist as one of the most financially destructive forms of cybercrime, deceiving users through fraudulent emails, malicious URLs, and spoofed web pages to harvest sensitive credentials and payment data. Conventional blacklist and rule-based defenses are fundamentally inadequate against newly registered phishing domains and zero-day threats that evade static catalogues. Machine learning has emerged as a powerful adaptive alternative, enabling detection systems to learn discriminative patterns from labeled data and generalize to previously unseen attacks. This paper presents a structured comparative review of machine learning and deep learning methods for phishing detection, synthesizing findings from fourteen recent studies spanning classical classifiers, ensemble methods, deep neural architectures, and reinforcement learning. Algorithms examined include Random Forest, SVM, XGBoost, LightGBM, Artificial Neural Networks, CNN, RNN-GRU, and hybrid ensemble models. Comparative analysis reveals that multi-modal ensemble and sequence-based deep learning approaches achieve the highest detection accuracy, with reported values reaching 99.98% for email phishing and 98.74% for large-scale URL classification. Persistent challenges including dataset imbalance, zero-day evasion, and real-time latency constraints are critically discussed. Future directions encompassing explainable AI, federated learning, transformer-based detection, and lightweight on-device models are proposed.

**Keywords:** Phishing Detection, Machine Learning, Deep Learning, URL Classification, Cybersecurity

## I. INTRODUCTION

The exponential growth of internet-based services across banking, commerce, healthcare, and government has simultaneously created fertile ground for cybercriminals. Phishing, a social engineering attack, is one of the most common dangers. To attempt to get login credentials, financial information, and personally identifying information, adversaries pose as reputable organizations using fake websites, fraudulent emails, or deceptive URLs. The Anti-Phishing Working Group (APWG) reports that damages from phishing-related fraud exceeded USD 54 million in 2020 alone, and that phishing events more than doubled between 2020 and 2021 [2], [6].

Conventional anti-phishing countermeasures, primarily blacklists and heuristic rule sets suffer from a structural weakness: they can only flag threats already indexed in maintained databases. These safeguards are completely circumvented by newly registered phishing domains, which are often online for less than 24 hours before decommissioning. A page becomes invisible to blacklist matching when a single character is changed in the URL [1], [13]. Heuristic rules are quicker to react, but they don't keep up with the evolution of phishing techniques and require manual threshold definition.

A fundamentally different paradigm is provided by machine learning (ML). ML systems learn decision limits using labeled training data instead of referencing static threat libraries, giving them the capacity to generalize to unknown phishing patterns. Rich feature vectors that consistently distinguish phishing from trustworthy sources are formed using features extracted from URL lexical structure, domain registration

metadata, HTML content, and network-level activity [3], [11]. By enabling automatic feature extraction from raw character sequences, deep learning architectures like Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Gated Recurrent Units (GRU) have further advanced the domain by doing away with the need for manually developed feature engineering [2], [13].

In order to (i) catalog algorithms, datasets, and feature strategies used; (ii) compare detection performance across studies; (iii) identify common limitations and open research gaps; and (iv) suggest fruitful future research directions, this paper synthesizes findings from fourteen recent research works. The scope includes the use of classical machine learning, deep learning, ensemble, and reinforcement learning techniques for phishing detection based on URLs, web pages, and emails.

## II. LITERATURE STUDY

A significant amount of work has been produced using deep learning, hybrid detection paradigms, and conventional machine learning. Key contributions are summarized in this part, which is followed by Table I for a systematic comparison. The findings are arranged by detection approach rather than article by paper summary.

### A. URL-Based Classical ML Approaches

A real-time detection framework that processes URL strings using NLP, word vector, and hybrid feature extraction techniques over 73,575 URLs was proposed by Sahingoz et al. [1]. With no reliance on outside resources, Random Forest with NLP features achieved 97.98% accuracy a crucial operational advantage. After benchmarking nine classifiers on almost 11,000 URL instances, Shahrivari et al. [11] discovered that XGBoost produced the fastest training time and the greatest classical accuracy at 98.32%. Rashid and associates [12] showed that PCA-based dimensionality reduction, which reduced the feature space to 22.5% of its original size, maintained 95.66% SVM accuracy while significantly lowering inference cost—a crucial discovery for deployment with limited resources. Using 3,000 URLs, Ahammad et al. [5] assessed five classifiers, including LightGBM, Random Forest, and SVM. They discovered that LightGBM was the top generalizer at 86% test accuracy thanks to its leaf-wise gradient boosting technique. In order to achieve 98.12% and outperform each component classifier separately, Karim et al. [14] created a hybrid LSD model that included Logistic Regression, SVC, and Decision Tree through canopy-based feature selection and soft voting.

### B. Web Page and Multi-Feature Approaches

Using the UCI Phishing Dataset (11,055 cases, 30 characteristics), Patil et al. [3] tested SVM, ANN, and Random Forest and found that SVM was superior at 89.84% accuracy with the lowest false positive rate. The study also created a deployable Chrome browser plugin. Using 11,430 balanced URLs with 87 features including URL structure, content hyperlinks, and external service signals, Hannousse and Yahiouche [15] created a repeatable benchmark. In addition to showing that external service features resulted in extraction delays surpassing 58,000 ms per sample, making them unsuitable for real-time use, Random Forest dominated all five experimental setups, peaking at 96.83% using chi square ranked features. Odeh et al. [10] examined deep and traditional machine learning techniques using the UCI and PhishTank datasets, verifying Random Forest's near 99.55% accuracy in several investigations [10].

### C. Deep Learning and Browser-Integrated Systems

Tang and Mahmoud [2] delivered a comprehensive browser-integrated framework pairing a cloud-hosted RNN - GRU model with a Chrome extension. Trained on 120,000 URLs across multiple sources, the GRU architecture achieved 99.18% accuracy, with a closed-loop feedback mechanism incorporating user-reported misclassifications to continuously update the model. Sahingoz et al. [13] developed DEPHIDES, training five deep architectures ANN, CNN, RNN, Bidirectional RNN, and Attention Networks on 5.1 million character-level URL embeddings. CNN with seven convolutional layers achieved 98.74% accuracy at throughput exceeding 130,000 URLs per second, demonstrating genuine real-time capability at production scale. Alsubaei et al. [8] proposed a ResNeXt-GRU hybrid framework integrating SMOTE oversampling, Jaya-algorithm hyperparameter optimization, and an ensemble autoencoder feature extraction module (EARN), achieving 98% accuracy and surpassing BERT, CNN, ResNet, and SVM baselines by 11–19 percentage points on Kaggle phishing data.

### D. Email Phishing and Reinforcement Learning

A two-phase framework was used by Mohammad et al. [4] to solve workplace email phishing: The integrated system achieved 94% accuracy by combining Feed-Forward Neural Networks for URL classification with a Mamdani Fuzzy Inference System that analyzes social factors like shared contacts, email frequency, and common behaviors to differentiate known from unknown phishing senders. The DARTH ensemble system was presented by Mittal et al. [6]. They broke down emails into body text, embedded URLs, and header metadata, trained BERT-based NLP, URL neural network, and metadata classifiers independently, and then combined them using an Ensemble Neural Network. DARTH obtained 99.98% accuracy, 99.97% precision, and a 99.98% F1-score on more than 150,000 emails. the highest documented in every study that was reviewed. After analyzing actual email data from university infrastructure, Salahdine et al. [16] discovered that a two-hidden-layer ANN with ReLU activation achieved 94.5% accuracy with just ten properly chosen characteristics and a 1.5% false alarm rate. In order to investigate Deep Reinforcement Learning, Chatterjee and Namin [9] trained a Deep Q-Network agent on 14 lexical characteristics, where each classification choice represents an action that is rewarded for accuracy. The DRL technique offers intrinsic adaptability to distribution change, a crucial conceptual feature for changing attack contexts, and achieves 90.1% accuracy, which is lower than supervised counterparts. In a systematic review of 43 deep learning studies, Catal et al. [7] confirmed the dominance of supervised learning, identified DNN/CNN/RNN as the top architectures, and identified model interpretability and benchmark standardization as the main unresolved issues.

**Table I: Comparative Summary of Reviewed Studies**

| Author(s) | Year | Algorithm | Dataset | Accuracy | Key Strength | Limitation |
|---|---|---|---|---|---|---|
| Sahingoz et al. [1] | 2019 | Random Forest (NLP) | 73,575 URLs | 97.98 % | No 3rd-party dependency | URL-only; short domains missed |
| Tang & Mahmoud [2] | 2022 | RNN-GRU | 120,000 URLs | 99.18 % | Browser extension; feedback loop | Short URLs; >200 char truncation |
| Patil et al. [3] | 2020 | SVM, ANN, RF | UCI (11,055 URLs) | 89.84 % | Chrome extension deployed | Small dataset; no deep learning |
| Mohammad et al. [4] | 2021 | FFNN + Fuzzy | 2,000 URLs / enterprise | 94% | Social-feature email analysis | MATLAB-based; limited scalability |
| Ahammad et al. [5] | 2022 | LightGBM, RF, DT | 3,000 URLs | 86% | Feature importance analysis | Small dataset; no HTML features |
| Mittal et al. [6] | 2022 | DARTH Ensemble (BERT) | 150,000 emails | 99.98 % | Multi-modal email fusion | Computationally heavy; English only |
| Catal et al. [7] | 2022 | DNN, CNN, RNN/LSTM | 43 studies (SLR) | — | Comprehensive gap analysis | No new model; reproducibility gap |
| Alsubaei et al. [8] | 2024 | ResNeXt-GRU | 10,000 URLs | 98% | SMOTE; Jaya optimisation | Modest dataset; not live-tested |
| Chatterjee & Namin [9] | 2019 | Deep Q-Network (DRL) | Ebbu2017 (73,575) | 90.1% | Adaptive RL policy | Lower accuracy; lexical-only |
| Odeh et al. [10] | 2021 | RF, SVM, CNN-LSTM | UCI, PhishTank (survey) | 99.55 % | Broad classical + DL survey | Review only; overfitting noted |

| Author(s) | Year | Algorithm | Dataset | Accuracy | Key Strength | Limitation |
|---|---|---|---|---|---|---|
| Shahrivari et al. [11] | 2020 | XGBoost, RF, ANN | ~11,000 URLs | 98.32 % | XGBoost fastest & most accurate | Small dataset; ANN underperforms |
| Rashid et al. [12] | 2020 | SVM + PCA | UCI Dataset | 95.66 % | 22.5% feature compression | SVM only; no deep learning |
| Sahingoz et al. [13] | 2024 | CNN (DEPHIDES) | 5.1 million URLs | 98.74 % | 130K URLs/sec throughput | High GPU demand; no URL hijacking |
| Karim et al. [14] | 2023 | Hybrid LSD | 11,054 URLs | 98.12 % | Canopy feature selection + voting | URL-only; no deep learning |

## III. MACHINE LEARNING TECHNIQUES FOR PHISHING DETECTION

### A. Decision Tree and Random Forest

Decision Trees partition the feature space through recursive binary splits guided by Gini impurity or information gain, producing interpretable classification rules that clearly articulate which URL attributes signal phishing. While individually prone to overfitting, they serve as the foundation of Random Forest an ensemble that constructs multiple independently trained trees on bootstrapped data subsets and aggregates predictions through majority voting. Across reviewed studies, Random Forest consistently ranks among the top classical classifiers: 97.98% with NLP features [1], dominant across all feature combinations in benchmark studies [15], and cited near 99.55% in survey literature [10]. Its robustness to high-dimensional inputs, implicit feature importance estimation, and freedom from feature scaling make it a reliable, interpretable production baseline [11].

### B. Support Vector Machine and Gradient Boosting

SVM identifies a maximum-margin hyperplane in a kernel-transformed feature space to separate phishing from legitimate instances. Patil et al. [3] found SVM the best single classifier on the UCI dataset at 89.84% with the lowest false positive rate, while Rashid et al. [12] demonstrated that PCA compressed features preserved 95.66% SVM accuracy with minimal computational overhead. XGBoost extends gradient boosting with regularization and parallel tree construction, achieving 98.32% accuracy with faster training than any neural network tested in the same study [11]. LightGBM applies leaf-wise tree growth for aggressive loss reduction, performing best among five classifiers in URL feature analysis [5].

### C. Artificial Neural Networks

Multi-layer ANNs learn non-linear feature interactions through backpropagation. Salahdine et al. [16] demonstrated a two-hidden-layer ANN achieving 94.5% accuracy on real-world email data with only 10 features and a 1.5% false alarm rate. Mohammad et al. [4] combined Feed Forward Backpropagation Networks with Fuzzy Inference for enterprise phishing, integrating social behavioral signals unavailable to URL only classifiers. Hybrid fuzzy-neural architectures are particularly valuable in organizational settings where sender reputation context improves detection of targeted spear-phishing attempts.

### D. Deep Learning: CNN, RNN-GRU, and Hybrid Architectures

CNN architectures apply convolutional filters over character-level URL embeddings to extract local n-gram patterns indicative of phishing. DEPHIDES [13] demonstrated seven layer CNN achieving 98.74% accuracy on 5.1 million URLs at 130,000 URLs/second throughput the only system reviewed demonstrating genuine production-scale real-time capability. RNN variants with GRU gates model sequential dependencies within URL character strings. Tang and Mahmoud [2] showed GRU outperforms LSTM and basic RNN, achieving 99.18% accuracy in a browser integrated deployment. Hybrid architectures amplify complementary strengths: Alsubaei et al. [8] fused ResNeXt convolutional extraction with GRU sequential modeling and ensemble autoencoders, gaining 11 to 19 percentage points over single

model baselines. For email phishing, Mittal et al. [6] demonstrated that BERT-based contextual embeddings, capturing bidirectional word relationships, enable the DARTH framework to achieve 99.98% accuracy by quantifying brand impersonation and urgency language patterns invisible to URL only systems.

### E. Ensemble Methods and Deep Reinforcement Learning

Ensemble strategies combine heterogeneous classifiers to reduce variance and improve robustness. The LSD model [14] achieved 98.12% by fusing Logistic Regression, SVC, and Decision Tree through canopy feature selection and soft voting, outperforming each constituent individually. DARTH [6] demonstrated that multi-modal ensembles integrating URL, body text, and metadata classifiers provide accuracy gains of 2 to 8 percentage points over any single modality. Deep Reinforcement Learning [9], while currently trailing supervised methods at 90.1%, offers an architecturally distinct advantage: the DQN agent continuously updates its classification policy as it encounters new distributions, making it conceptually promising for adversarially dynamic environments where attack patterns shift continuously.

## IV. COMPARATIVE ANALYSIS

Table II presents the best-reported performance metrics per reviewed study. Direct cross-study comparison is complicated by varying dataset sizes (2,000 to 5.1 million instances), feature sets, class balance strategies, and evaluation protocols. Nonetheless, consistent patterns emerge.

**Table II: Algorithm Performance Comparison (Best Results per Study)**

| Algorithm | Study | Accuracy | F1-Score | Key Characteristic |
|---|---|---|---|---|
| DARTH Ensemble (BERT+URL+Meta) | Mittal et al. [6], 2022 | 99.98% | 99.98% | Multi-modal email features; highest reported accuracy |
| RNN-GRU | Tang & Mahmoud [2], 2022 | 99.18% | 99.15% | Character-level URL sequences; browser-integrated |
| CNN (DEPHIDES) | Sahingoz et al. [13], 2024 | 98.74% | ~0.99 | 130,000 URLs/sec; language-independent embeddings |
| XGBoost | Shahrivari et al. [11], 2020 | 98.32% | 0.976 | Fastest classical training; regularised boosting |
| Hybrid LSD (LR+SVC+DT) | Karim et al. [14], 2023 | 98.12% | 0.958 | Canopy feature selection; soft-voting ensemble |
| ResNeXt-GRU | Alsubaei et al. [8], 2024 | 98% | ~98.7% | SMOTE balancing; Jaya hyperparameter tuning |
| Random Forest (NLP) | Sahingoz et al. [1], 2019 | 97.98% | 98% | No third-party dependency; language-independent |
| SVM + PCA | Rashid et al. [12], 2020 | 95.66% | ~95.5% | Feature space compressed to 22.5%; efficient |
| ANN (2 hidden layers) | Salahdine et al. [16], 2021 | 94.5% | ~92% | Real-world email data; low false-alarm (1.5%) |
| Deep Q-Network (DRL) | Chatterjee & Namin [9], 2019 | 90.1% | 87.3% | Adaptive RL policy; inherent distribution shift handling |

Multi-modal ensemble frameworks and sequence-based deep architectures dominate the performance hierarchy. DARTH [6] records the highest accuracy (99.98%) by fusing heterogeneous email features, while RNN-GRU [2] leads URL detection at 99.18%. DEPHIDES CNN [13] achieves 98.74% with unmatched real-time throughput the only reviewed system validated at production scale. Among classical methods, XGBoost [11] leads at 98.32%, confirming that gradient boosting remains highly competitive on

structured tabular URL features without the computational overhead of deep architectures. The Hybrid LSD model [14] at 98.12% demonstrates that carefully engineered ensemble architectures can rival deep learning on moderate datasets. DRL [9] at 90.1% currently trails supervised methods but offers adaptability properties that supervised systems fundamentally lack. The convergence of results across independent studies confirms ensemble learning and deep sequential architectures as the current state of the art, while also underscoring that dataset quality and feature diversity are equally important determinants of performance.

## V. CHALLENGES AND LIMITATIONS

Despite significant advances, several fundamental barriers impede robust and widely deployable phishing detection.

### A. Dataset Imbalance and Benchmark Scarcity

Classifiers are biased toward the majority class because real-world online traffic comprises considerably more authentic URLs than phishing ones. Class imbalance reduces F1-score by up to 42%, as Hannousse and Yahiouche [15] showed. Alsubaei et al.'s use of SMOTE oversampling [8] increased accuracy from 83% to 98%, demonstrating the effect of balancing techniques. Beyond balance, the majority of datasets experience temporal obsolescence — archived phishing URLs quickly reduce the representativeness of contemporary strategies and meaningful cross-study comparison is hard in the absence of a well recognized benchmark [7].

### B. Zero-Day Phishing and Adversarial Evasion

Newly registered domains absent from training data evade all ML models trained on historical distributions. Simultaneously, sophisticated phishers employ URL shorteners, punycode encoding, Unicode homoglyphs, multi-hop redirections, and HTTPS certificate acquisition. APWG reported over 83% of phishing sites used HTTPS in 2021, rendering conventional SSL indicators unreliable [10]. Adversarial perturbations specifically crafted to deceive trained classifiers represent an emerging and largely unresolved threat [9], [13].

### C. Real-Time Detection Latency

External feature queries — WHOIS records, DNS lookups, Google PageRank, Alexa traffic rank — introduce network-dependent delays of multiple seconds per URL, fundamentally incompatible with browser-integrated real-time protection. Hannousse and Yahiouche [15] measured average extraction delays exceeding 58,000 ms for two hyperlink features despite their high discriminative power. While DEPHIDES [13] achieves 130,000 URLs/sec on GPU infrastructure, edge-device deployment remains technically constrained, and the ResNeXt-GRU system [8] reports 37-second batch processing unsuitable for live use.

### D. Model Interpretability

Deep learning architectures function as black boxes, offering no insight into why specific URLs or emails are flagged. In security-critical applications, the inability to audit model decisions limits operational trust and enterprise adoption. Catal et al. [7] identify interpretability as the foremost open challenge in the field, for which no satisfactory general solution has been proposed across the 43 reviewed deep learning studies.
.

## VI. FUTURE SCOPE

The following research directions hold particular promise for advancing phishing detection beyond current limitations.

### A. Explainable AI (XAI)

Security analysts might assess model choices, pinpoint failure reasons, and establish the operational trust necessary for enterprise deployment by including SHAP or LIME explanations into detection pipelines. An additional interpretability pathway is provided by gradient-based attention visualization in RNN/CNN models [7].

### B. Federated Learning

Federated architectures handle both data scarcity and privacy concerns at the same time by enabling cooperative model training across companies, including internet service providers, businesses, and browser vendors, without centralizing sensitive URL or email data [6].

### C. Transformer and LLM Integration

Building on the established performance of DARTH [6] for English email phishing, extending BERT-style pre-training to URL character sequences and multilingual phishing content could significantly increase generalization to semantically complicated evasion methods.

### D. Lightweight On-Device Models

By compressing high-accuracy architectures like RNN-GRU [2] into browser-native extensions that function without cloud round-trips, knowledge distillation, model pruning, and quantization approaches could combine detection quality with true real-time latency budgets.

## VII. CONCLUSION

The results of fourteen research contributions including deep learning, hybrid ensemble, classical machine learning, and reinforcement learning approaches to phishing detection were compiled in this study. With DARTH reaching 99.98% on email data [6], RNN-GRU achieving 99.18% on URL sequences [2], and DEPHIDES CNN reaching 98.74% at production-scale throughput [13], the analysis confirms a distinct performance hierarchy: multi-modal ensemble frameworks and sequence-based deep architectures achieve the highest accuracy. Among standard methods, a Random Forest with NLP features at 97.98% [1] and XGBoost is at 98.32% [11] provide competitive results with useful computational efficiency and interpretability advantages. The key research frontier is defined by persistent issues such as dataset imbalance, adversarial URL manipulation, zero-day evasion, real-time delay under edge limitations, and black-box opacity. There isn't a single method that works best in every deployment scenario; instead, the best option is determined by factors including data size, feature availability, latency constraints, and interpretability criteria. Coordinated developments in explainable AI to foster operational trust, adversarially based robust training to harden models against intentional evasion, federated learning to overcome data scarcity, and lightweight inference architectures to enable true browser-native real-time protection are necessary for the future. The development of adaptable, comprehensible, and operationally feasible detection systems continues to be a top concern for the cybersecurity research community as phishing tactics continue to increase in complexity and scope.

## REFERENCES

[1] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine Learning Based Phishing Detection from URLs," Expert Systems with Applications, vol. 117, pp. 345–357, Mar. 2019.

[2] L. Tang and Q. H. Mahmoud, "A Deep Learning-Based Framework for Phishing Website Detection," IEEE Access, vol. 10, pp. 1509–1521, Jan. 2022.

[3] S. Patil, Y. Shetye, and N. Shendage, "Detecting Phishing Websites Using Machine Learning," International Research Journal of Engineering and Technology, vol. 7, no. 2, Feb. 2020.

[4] G. B. Mohammad, S. Shitharth, and P. R. Kumar, "Integrated Machine Learning Model for an URL Phishing Detection," International Journal of Grid and Distributed Computing, vol. 14, no. 1, pp. 513–529, 2021.

[5] S. K. H. Ahammad et al., "Phishing URL Detection Using Machine Learning Methods," Advances in Engineering Software, vol. 173, p. 103288, Nov. 2022.

[6] A. Mittal, D. W. Engels, H. Kommanapalli, R. Sivaraman, and T. Chowdhury, "Phishing Detection Using Natural Language Processing and Machine Learning," SMU Data Science Review, vol. 6, no. 2, Art. 14, 2022.

[7] C. Catal, G. Giray, B. Tekinerdogan, S. Kumar, and S. Shukla, "Applications of Deep Learning for Phishing Detection: A Systematic Literature Review," Knowledge and Information Systems, vol. 64, pp. 1457–1500, 2022.

[8] F. S. Alsubaei, A. A. Almazroi, and N. Ayub, "Enhancing Phishing Detection: A Novel Hybrid Deep Learning Framework for Cybercrime Forensics," IEEE Access, vol. 12, pp. 8373–8389, Jan. 2024.

[9] M. Chatterjee and A. S. Namin, "Detecting Phishing Websites through Deep Reinforcement Learning," in Proc. 43rd IEEE Annual COMPSAC, Milwaukee, WI, 2019, pp. 227–232.

[10] A. Odeh, I. Keshta, and E. Abdelfattah, "Machine Learning Techniques for Detection of Website Phishing: A Review for Promises and Challenges," in Proc. IEEE Conf. Software and Application, 2021.

[11] V. Shahrivari, M. M. Darabi, and M. Izadi, "Phishing Detection Using Machine Learning Techniques," arXiv:2009.11116, Sep. 2020.

[12] J. Rashid, T. Nazir, T. Mahmood, and M. W. Nisar, "Phishing Detection Using Machine Learning Technique," in Proc. 1st Int. Conf. Smart Systems and Emerging Technologies (SMARTTECH), IEEE, 2020, pp. 43–46.

[13] O. K. Sahingoz, E. Buber, and E. Kugu, "DEPHIDES: Deep Learning Based Phishing Detection System," IEEE Access, vol. 12, pp. 8052–8070, 2024.

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

[15] A. Hannousse and S. Yahiouche, "Towards Benchmark Datasets for Machine Learning Based Website Phishing Detection: An Experimental Study," arXiv:2010.12847, Oct. 2020.

[16] F. Salahdine, Z. El Mrabet, and N. Kaabouch, "Phishing Attacks Detection: A Machine Learning-Based Approach," in Proc. IEEE Int. Conf. Communications (ICC), 2021.