SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I5P110

Volume 13 Issue 5, 97-108, May 2026


*Original Article*

# Optimized and Explainable Multimodal Deep Learning Framework for Accurate Phishing Website Detection

Alaa Kamel Ali[1], Mohsen Rezvani[2]

[1,2]*Faculty of Computer Engineering, Shahrood University of Technology, Shahrood 3619995161, Iran.*

[2]*Corresponding Author : mrezvani@shahroodut.ac.ir*

Received: 09 February 2026 Revised: 10 March 2026 Accepted: 11 April 2026 Published: 27 May 2026

***Abstract*** *- Phishing websites remain a significant threat to online users by exploiting deceptive visual layouts, manipulated HTML content, and misleading structural designs to obtain sensitive information. Existing detection approaches mainly rely on single-modal features and often function as black-box systems, which limits their effectiveness against advanced phishing strategies and reduces user trust. This paper proposes an optimized and explainable multimodal deep learning framework for accurate phishing website detection. The proposed model integrates Bidirectional Long Short-Term Memory (BiLSTM) networks for sequential HTML analysis, Graph Convolutional Networks (GCN) for capturing DOM structural dependencies, and Convolutional Neural Networks (CNN) for extracting visual features from webpage screenshots. To enhance training stability and overall performance, Ant Colony Optimization (ACO) is employed to automatically tune critical hyperparameters. In addition, Shapley Additive exPlanations (SHAP) are incorporated to interpret model decisions by quantifying the contribution of each feature modality. Extensive experiments conducted on a large-scale phishing dataset demonstrate that the proposed framework outperforms conventional single-modal and hybrid models, achieving improved accuracy and F1-score. The results confirm that the integration of multimodal learning, metaheuristic optimization, and explainable AI provides a reliable and transparent solution for phishing website detection.*

***Keywords*** *- Explainable Artificial Intelligence, Multimodal Learning, Phishing Detection, Deep Learning, Hyperparameter Optimization.*

## 1. Introduction

Phishing websites have become one of the most severe and persistent cyber threats, exploiting visual imitation, deceptive content, and social engineering techniques to mislead users into disclosing sensitive information such as login credentials and financial details [1-3]. The continuous advancement of phishing techniques, including web page cloning, dynamic HTML generation, and complex user interface manipulation, has significantly degraded the performance of traditional blacklist-based and rule-based detection approaches [4]. Recent cybersecurity reports indicate that phishing attacks remain among the most prevalent entry points for cybercriminals, causing substantial financial losses and undermining user trust in online services [5, 6].

To address these threats, a wide range of Machine Learning (ML) and Deep Learning (DL) approaches have been proposed for phishing detection by analyzing URLs, HTML content, and webpage structures [7, 8]. Although these methods demonstrate promising detection capabilities, most existing approaches rely on single-modal features (e.g., URL-based or content-based features), which are insufficient to capture the complexity of modern phishing websites that intentionally evade lexical and syntactic detection patterns [9]. In addition, many DL-based frameworks operate as black-box systems, providing only binary outputs without explaining the reasoning behind their predictions. This lack of transparency limits their adoption in real-world security environments, where interpretable and trustworthy decisions are essential for analysts [10, 11]. Another important limitation of current phishing detection systems is the absence of effective hyperparameter optimization strategies. Model performance is highly influenced by parameters such as learning rate, network depth, and regularization factors, which are often selected manually or empirically. This may result in unstable training, non-convergence, or poor generalization across diverse and evolving datasets. Furthermore, existing approaches provide limited explainability, making it difficult to identify which webpage attributes contribute most to phishing detection, thereby reducing user trust and situational awareness.

Nonetheless, there is still a strong research gap when it comes to a holistic design that combines multimodal feature learning, automated optimization, and interpretable decision

making all under the same roof. Many aspects of the problem (for example, feature extraction vs classification) are addressed separately without providing a unified and interpretable solution. Heterogeneity is a significant concern, which exposes the limitation of phishing detection in this way, as future work needs to further develop a more robust detection framework with emphasis on how transparency can play an important role.

To fill the gaps of these works, this research proposes a new optimized multimodal phishing detection framework for phishing detection with explainability. We propose a new model consisting of three components: Bidirectional Long Short Term Memory (BiLSTM) networks to sequentially represent HTML contents, Graph Convolutional Networks (GCN) to show structural dependencies of DOM graphs, and Convolutional Neural Network (CNN) for visual feature extraction from web page screenshots. To achieve that, Ant Colony Optimization (ACO) is employed to perform an automatic search for hyperparameters, which are important aspects for stable training and improved detection performance. SHapley Additive exPlanations (SHAP), which provides visibility into how much each feature modality contributes to the decisions of a model, is yet another layer to enhance interpretability/trust for users.

Figure 1 illustrates the complete research processes of the proposed framework with various stages, including multimodal data acquisition, feature extraction modules, ACO-based optimization mechanism, feature fusion mechanism, as well as classification and explainable components.

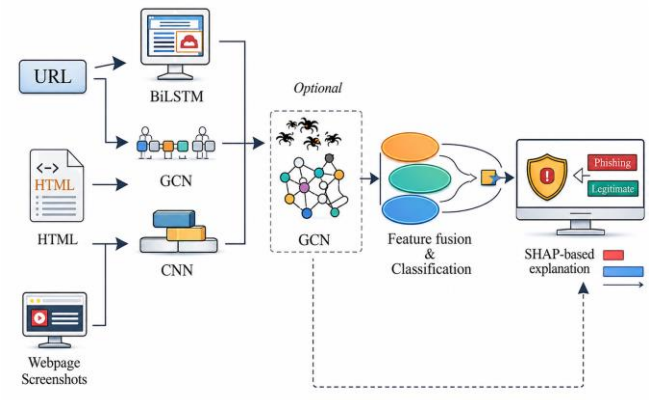


**Fig. 1 Overall research workflow**

The main contributions of this study could be summarized by the following points:

- The first multimodal phishing detection framework, which jointly considered textual, structural, and visual webpage features using BiLSTM, GCN, and CNN architectures to create a more holistic representation compared to existing single-modal solutions.
- ACO-based hyper-parameter optimization for detection that automates the tuning process, while ensuring resilience and generalization of the proposed solution.
- SHAP-based explainable decision module to get interpretability from the phishing detection process and extra transparency at an add-on system level.

The rest of the paper is organized as follows. Section 2 reviews related work on phishing detection. In Section 3, we present the proposed methodology and methodological background. In Section 4, we present the experimental setup, datasets used, as well as evaluation metrics and comparative results. Lastly, Section 5 concludes the paper and proposes paths for future work.

## 2. Related Work

### *2.1. Traditional Phishing Detection*

The earliest phishing detection mechanisms had extensively depended on handcrafted rules and traditional classifiers based on lexical and syntactic features from URLs and HTML source code. Malicious webpages were detected using techniques such as blacklist matching, pattern-based heuristics, and domain reputation scoring [12, 13]. While these approaches were reported to work well in detecting known phishing templates, they remain intolerably vulnerable to zero-day attacks and adaptive polymorphic phishing techniques that intentionally alter the lexical tokens of a message to evade detection [14]. Additionally, outlined methods are not scalable and cannot adapt to changing environments since they rely on using handcrafted features.

### *2.2. Deep Learning–Based Detection*

Deep learning techniques have gained considerable attention in recent studies because they learn the representative features from raw data automatically, which avoids the drawbacks of traditional methods. As an example, CNNs have been used to extract the visual similarity between legitimate and phishing pages using screenshots of web pages [15], while RNN-based models (including LSTM networks) have shown good performance in modeling sequential dependencies in URLs and HTML content [16, 17].

Phishing detection has been further enhanced using transformer-based architectures and attention mechanisms that capture contextual semantics in textual data and long-range dependencies [18]. Although these advancements are crucial, most deep learning approaches still focus on a single modality, which is not robust to more sophisticated phishing attacks that can simultaneously exploit textual, structural, and visual features.

### *2.3. Hybrid DL–Graph Models*

We propose a new hybrid framework that makes use of structurally complicated interdependencies between elements on web pages, inspired by recent breakthroughs in deep learning and graph-based learning. Graph Convolutional Networks (GCN) have been adopted to represent both the

DOM trees and hyperlink topologies [13, 19] as a novel approach to detect phishing behaviors that are missed by isolated text features. These hybrid methods achieved better mining of the phishing features by combining CNN-based visual feature extraction and GCN-based structural learning [20, 21]. However, most DL–Graph models are computationally expensive without any systematic optimization methods, which restricts their applications to real-time conditions. More importantly, those models are often not interpretable, making them unsuitable for production cybersecurity environments.

### *2.4. Metaheuristic Optimization in Security*

Various metaheuristic optimization methods, such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), and Ant Colony Optimization (ACO), have gained significant interest in the systematization of cyber defense frameworks via hyperparameter autotuning/feature selection [22-24]. Results in [25] show that these techniques can, to a large extent, improve the identification accuracy and convergence stability of phishing attacks compared with traditional hand-crafted deep neural networks. However, the current studies use only uni-modal architectures with metaheuristic optimization and do not investigate their potential for optimizing multimodal models in phishing detection. This implies that optimization methods rarely apply to complex, multi-source data representations.

### *2.5. Explainable AI in Phishing Detection*

This, combined with the black-box nature of deep learning models, has spurred a vast amount of research into Explainable Artificial Intelligence (XAI). Domain of malware detection and classification tasks, intrusion detection, etc. For such reasons, the approaches enable explanations by quantifying the contribution of individual components to model decisions, and enhance transparency and user trust while providing for decision support across a variety of Cyber Security use cases. Nonetheless, the explainability in phishing detection is still limited, especially for multimodal settings. This demonstrates a clear gap in the research, as most current works focus solely on single-modal explanations and do not provide for unified interpretability across textual, structural, and visual features.

Table 1 provides a summary of the main characteristics, datasets, and methodological limitations of evaluated phishing detection studies, while identifying the research gaps that are filled by this work.

**Table 1. Summary of reviewed phishing detection studies**

| Ref. | Year | Approach Type | Data Modalities | Core Technique | Optimization | Explainability | Main Limitation |
|---|---|---|---|---|---|---|---|
| [12] | 2024 | DL (URL-based) | URL | CNN-based URL classification | None | No | Limited to lexical features |
| [13] | 2025 | Multimodal DL | URL + HTML | GNN + Transformer (D-PhishNet) | None | No | No explainability, limited optimization |
| [14] | 2021 | DL (Visual) | Screenshot images | CNN-based logo matching | None | No | No structural/textual modeling |
| [15] | 2022 | DL (Visual) | Screenshot + behavior | Visual phishing detection | None | No | Weak feature fusion |
| [16] | 2022 | DL–Graph | DOM graph, hyperlinks | GCN-based detection | None | No | No visual modality |
| [17] | 2025 | ML + XAI | Tabular phishing features | XGBoost + SHAP | None | Yes | No deep learning, limited modalities |
| [18] | 2023 | DL (Transformer) | Textual content | Attention/Transformer models | None | No | Single-modal limitation |
| [19] | 2025 | XAI-based ML | Tabular features | SHAP-based feature selection | None | Yes | No multimodal integration |
| [20] | 2025 | XAI Framework | Model outputs | XAI evaluation framework | None | Yes | Not specific to phishing DL models |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| [21] | 2024 | DL + Optimization | Network traffic | LSTM/GRU + GWO | GWO | No | Not applied to phishing, no XAI |
| [22] | 2025 | Hybrid DL-ML | URL, HTML | CNN + SVM | GWO | No | No visual modality, no explainability |
| [23] | 2024 | Metaheuristic + DL | Generic | ABC + Deep Learning optimization | ABC | No | General framework, not phishing-specific |
| [24] | 2024 | Visual DL | Screenshots | Visual similarity analysis | None | No | Single-modal approach |
| [25] | 2022 | Visual DL | Logos + images | CNN-based logo detection | None | No | Limited feature diversity |
| [26] | 2020 | DL Hybrid | URL + Screenshot | CNN-based phishing detection | None | No | No graph modeling, no explainability |
| Proposed Model | 2026 | Multimodal DL + Graph + XAI | HTML + DOM + Screenshot | BiLSTM + GCN + CNN fusion | ACO | SHAP | Addresses multimodal, optimization, and interpretability gaps |

## 3. Proposed Methodology

### *3.1. Framework Overview*

The proposed architecture is described below in this section. We develop three complementary deep learning modules to extract different features of phishing websites: (i) sequential patterns in HTML contents using a BiLSTM network, (ii) structural dependencies captured from the Document Object Model (DOM) tree utilizing Graph Convolutional Network (GCN), and (iii) discriminative visual content features derived from website screenshots using Convolutional Neural Network. The extracted features from these heterogeneous modalities are then fused together and input into a classification layer, thus enabling efficient learning and achieving high robustness for deployment in different environments to predict webpage legitimacy.

Figure 2 illustrates the overall architecture of the proposed multimodal framework, including data acquisition, feature extraction modules, ACO-based hyperparameter optimization, multimodal features fusion, and SHAP-based explainability. contributing to strong performance as well as interpretability, in addition to scalability and reliability, enabling it to be ready for real-time deployment. In this paper, we propose an efficient framework that captures complementary information from multiple feature spaces to boost detection capability against both traditional and advanced phishing attacks. Moreover, Stable convergence of the model is achieved by ACO optimization along with parameter tuning. It gives more transparency while performing the predictions and helps to make decisions in context with real-world cyber-impact.

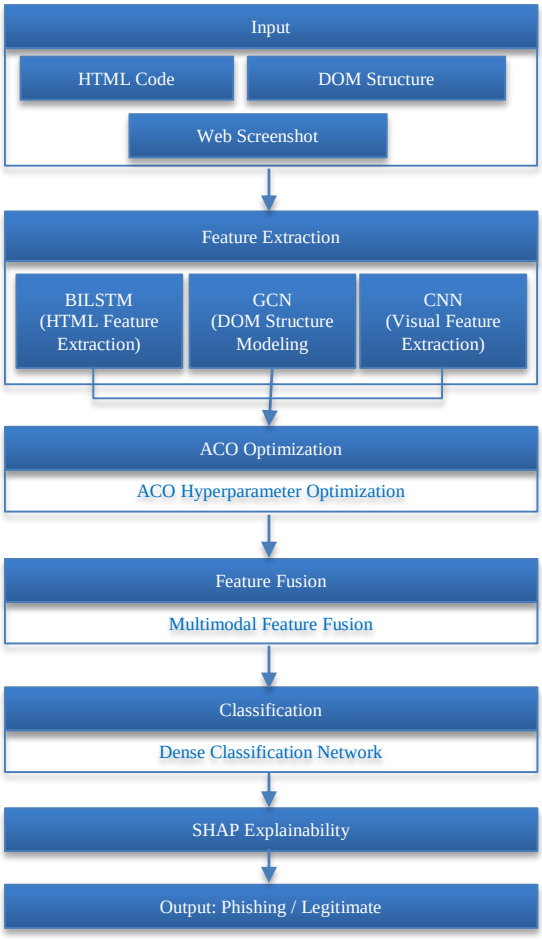


**Fig. 2 Architecture of the proposed multimodal framework**

### *3.2. Data Collection and Preprocessing*

The experimental dataset is developed in this study, which combines multiple phishing as well as legitimate web repositories. More concretely, the phishing samples are retrieved from PhishTank and OpenPhish, whereas benign URLs are acquired from the Tranco Top-1M list. In particular, different types of data modalities are collected for each webpage instance, including the URL, the raw HTML page's text content (text within all tags), the DOM structure, and a screenshot image from the loaded web page to better capture accompanying characteristics about pages. In the preprocessing step, multiple operations are conducted to transform the data into a structure amenable to optimal model training. Initially, it processes and tokenizes the HTML content, taking special care to examine script-based elements and embedded patterns before integrating them into a linear sequential representation. This enables the BiLSTM model to learn contextual dependencies in HTML structure. Second, the DOM tree is converted into a graph-based representation in which nodes represent HTML tags and edges store parent–child relationships, so that structural learning can be performed through the GCN model. Third, we resize and normalize the webpage screenshots, which gives a fixed size for matched dimensions to input for the CNN component, ensuring stable training. This removes duplicates and corrupted HTML files to improve the quality of data before training. Moreover, to ensure the coherence of contributions among different modalities in the fusion stage and avoid one specific modality prevailing over others, feature normalization is also exercised on all modalities. Lastly, the dataset is divided into training, validation, and testing (70:15:15) subsets, maintaining class distribution across the data splits to improve proper evaluation. This preprocessing strategy helps to strengthen the robustness and generalization of the proposed multimodal framework. In Table 2, we summarize the data sources and the distribution of phishing samples vs. legitimate ones throughout the dataset. Figure 3 shows the multilayer data preprocessing pipeline, which transforms raw webpage data into a structured, as well as a graph and visual representation.

**Table 2. Data sources and sample distribution**

| Source | Type | Modality Collected | No. of Samples |
|---|---|---|---|
| PhishTank | Phishing | URL, HTML, DOM, Screenshot | 9,200 |
| OpenPhish | Phishing | URL, HTML, DOM, Screenshot | 7,500 |
| PhishStats | Phishing | URL, HTML, DOM, Screenshot | 5,300 |
| Tranco Top-1M | Legitimate | URL, HTML, DOM, Screenshot | 22,000 |
| Total | — | Multimodal | 44,000 |

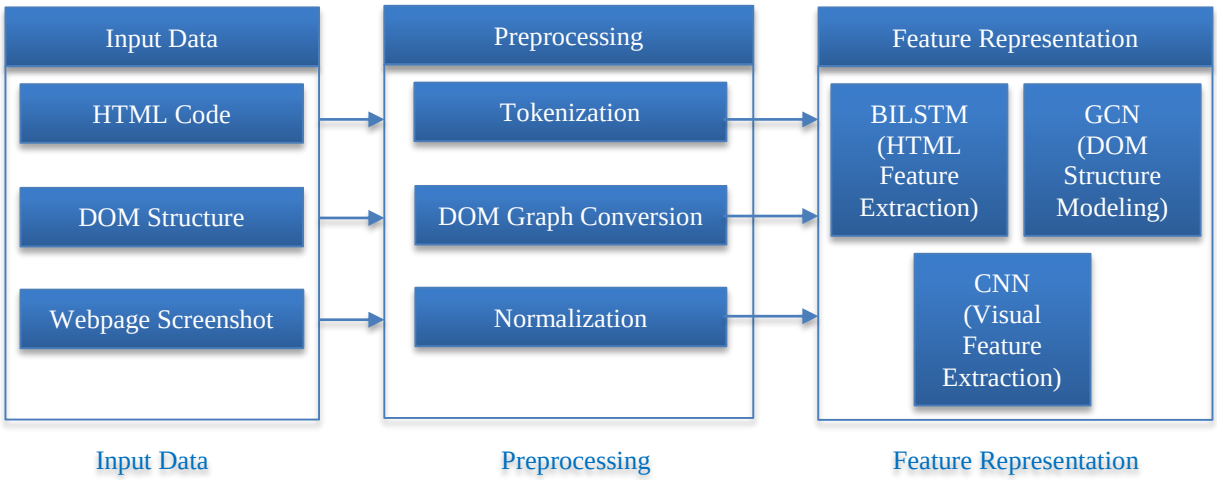


**Fig. 3 Multilayer data preprocessing pipeline**

### *3.3. Feature Extraction Modules*

*3.3.1. HTML Feature Learning Using BiLSTM*

The dependencies of HTML token streams are modeled by a Bidirectional Long Short-Term Memory (BiLSTM) network.

BiLSTM passes through the input sequence in two ways (forward and backward); it helps models to explore the contextual relationship of words processed before a particular position or after. The hidden representation at each time step is defined as:

$$\overrightarrow{h_t} = \text{LSTM}(x_t, \overrightarrow{h_{t-1}}) \quad (1)$$

$$\overleftarrow{h_t} = \text{LSTM}(x_t, \overleftarrow{h_{t+1}}) \quad (2)$$

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \quad (3)$$

Where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ denote the forward and backward hidden states, respectively.

*3.3.2. DOM Structural Features using GCN*

A Graph Convolutional Network models the structural relationships between HTML elements. Each web page's DOM tree is expressed as a graph, and the GCN propagates information between neighboring nodes through an iterative process to identify structural anomalies that are highly indicative of phishing behavior. The propagation rule for the l-th layer is given as follows:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \quad (4)$$

Where Ã is the adjacency matrix with self-loops, D̃ is the degree matrix, $W^l$ is the learnable weight matrix, and $\sigma(\cdot)$ denotes the activation function.

*3.3.3. Visual Feature Extraction using CNN*

A CNN is used to examine the visual similarity of a given link with known phishing pages by extracting spatial contents from webpage screenshots. This module captures visual similarity, cues of brand imitation, and layout inconsistencies commonly employed by phishing attackers.

***3.4. Multimodal Feature Fusion***

A single multimodal feature vector encapsulating sequential, structural, and visual properties of the webpage under analysis is built by concatenating the feature representations obtained by the BiLSTM, GCN, and CNN. Finally, this fused representation is passed through a fully connected classification layer that predicts whether the webpage is genuine or phishing.

Figure 4 provides an overview of the fusion and classification process. 4, meaning the heterogeneous feature streams are merged into one decision stream.

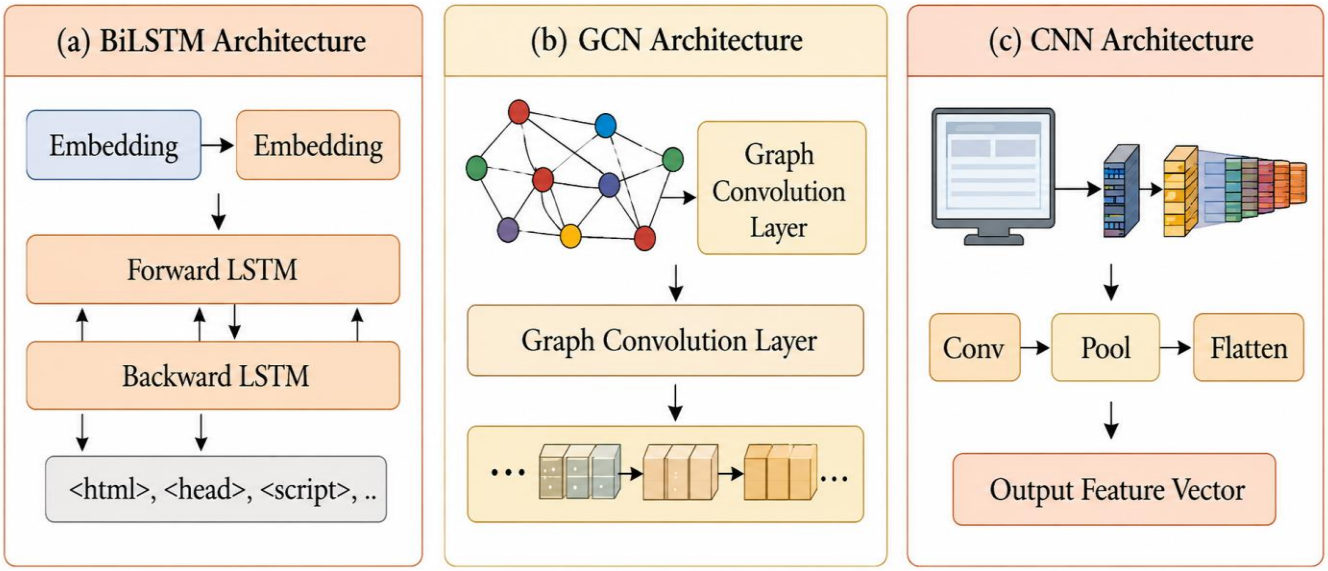


**Fig. 4 Feature fusion and classification layer**

***3.5. Model Optimization using Ant Colony Optimization (ACO)***

Hyperparametric optimization of the proposed multimodal framework using Ant Colony Optimization (ACO). In this case, we take each ant to be a hyperparameter setting (e.g., learning rate, number of hidden units, batch size; dropout ratio, etc.). We evaluate the performance of each configuration based on the validation accuracy achieved by training the BiLSTM–GCN–CNN model with respective parameters. The full optimization process is shown in Algorithm 1, while the iterative pheromone updating and solution construction flow is demonstrated in Figure 5, illustrating the execution of the ACO optimization procedure.

**Algorithm 1. ACO-Based Hyperparameter Optimization**

Input:

- Training set Dtrain, validation set Dval
- Search space Ω for hyperparameters (e.g., learning rate, batch size, dropout, hidden units)
- Number of ants m, maximum iterations T
- Evaporation rate ρ ∈ (0,1), pheromone influence α, heuristic influence β
- Pheromone bounds [τmin, τmax]

Output:

- Best hyperparameter configuration θ*
- Best validation score Acc*

1: Initialize pheromone trails τ for each hyperparameter choice in Ω (set τ = τ0 within [τmin, τmax])
2: Set Acc* ← 0, θ* ← null
3: for iter = 1 to T do
4: for k = 1 to m do
5: // Construct a candidate solution (hyperparameter set) using probabilistic selection
6: Initialize θk as empty

7: for each hyperparameter hp in Ω do
8: Compute selection probabilities for each option v ∈ hp:
9: P(v) = (τ(v)^α · η(v)^β) / Σu∈hp (τ(u)^α · η(u)^β)
10: Sample one option v according to P(v) and add it to θk
11: end for
12:
13: // Train and evaluate candidate configuration
14: Train the multimodal model (BiLSTM–GCN–CNN + fusion) on Dtrain using θk
15: Compute validation accuracy Ack on Dval
16:
17: if Ack > Acc* then
18: Acc* ← Ack
19: θ* ← θk
20: end if
21: end for
22:
23: // Evaporate pheromone
24: for each pheromone entry τ(v) do
25: τ(v) ← (1 − ρ) · τ(v)
26: end for
27:
28: // Reinforce pheromone based on the best solutions (elitist update)
29: Select θbest_iter with the highest validation accuracy in the current iteration
30: for each selected option v in θbest_iter do
31: τ(v) ← τ(v) + Δτ(v)
32: where Δτ(v) = Acc(θbest_iter) // reward proportional to performance
33: Clamp τ(v) to [τmin, τmax]
34: end for
35:
36: end for
37: Return θ*, Acc*

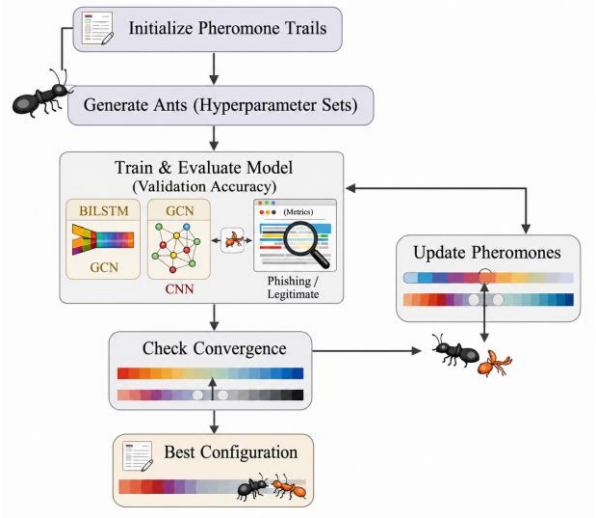


**Fig. 5 ACO optimization process**

### *3.6. Explainability using SHAP*

In order to explain the predictions made by the BiLSTM–GCN–CNN model, Gradient SHAP is utilized to interpret the proposed multimodal phishing detection framework. This approach predicts the importance of each feature by calculating the average gradients between baseline samples and the input. Algorithm 2 summarizes the explainability process. HAP values reveal which HTML tokens, DOM structural elements, and visual cues are more relevant to identify if a given page is legitimate or phishing. Overall explanation process and a sample feature attribution heatmap are illustrated in Figure 6, showing the contribution of each modality to the final prediction.

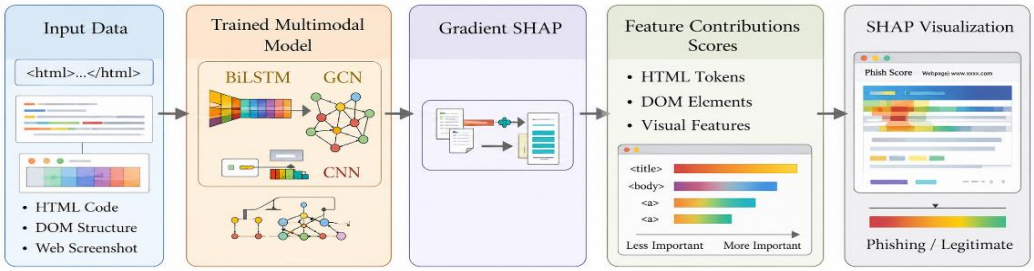


**Fig. 6 SHAP-Based Explainability Framework for Multimodal Phishing Detection**

**Algorithm 2. Gradient SHAP Explanation**

Input:
- Trained multimodal model f(·) with BiLSTM–GCN–CNN + fusion classifier
- Target sample x = {x_html, x_dom, x_img}
- Baseline set B = {b1, b2, …, bM} (reference samples)
- Number of interpolation steps S
- Target output class c (e.g., phishing)

Output:
- SHAP attribution vector ϕ for features in {HTML, DOM, Image}

1: Initialize attribution vector ϕ ← 0
2: for each baseline sample b in B do
3: for s = 1 to S do
4: // Interpolate between baseline and input
5: α ← s / S
6: x̃(s) ← b + α · (x − b)

7:
8: // Forward pass and compute gradient w.r.t. input features
9: yc ← f_c(x̃(s)) // model score for class c
10: g(s) ← ∂yc / ∂x̃(s) // gradient of score w.r.t. inputs
11:
12: // Accumulate attributions along the interpolation path
13: ϕ ← ϕ + g(s) ⊙ (x − b) // ⊙ is element-wise multiplication
14: end for
15: end for
16: // Average across steps and baselines (expected gradients approximation)
17: ϕ ← ϕ / (M · S)
18: // Split attributions by modality (optional reporting)
19: ϕ_html ← attributions corresponding to x_html features
20: ϕ_dom ← attributions corresponding to x_dom (graph) features
21: ϕ_img ← attributions corresponding to x_img (visual) features
22: Return ϕ = {ϕ_html, ϕ_dom, ϕ_img}

## 4. Experimental Evaluation

### *4.1. Experimental Setup*

All experiments were conducted on a workstation with an Intel Core i7 CPU, 32GB RAM, and an NVIDIA RTX-3080 GPU. We implemented the model in Python with TensorFlow and PyTorch. HTML parsing and construction of the DOM graph were performed by BeautifulSoup and NetworkX libraries, respectively.

The visual snapshots of webpages were resized to 224×224 prior to inputting them into the CNN module. The ACO algorithm was utilized for optimization to adjust the learning rate, batch size, number of hidden units, and dropout ratio, as shown in Figure 5. The dataset was split randomly into 70% training,15% for validation, and 15% for testing.

### *4.2. Baseline Models*

To evaluate the effectiveness of the proposed multimodal phishing detection framework, several baseline models were implemented for comparative analysis, representing common single- and bimodal strategies in phishing detection. Specifically, URL-CNN utilizes lexical URL features, HTML-BiLSTM processes sequential HTML tokens, Screenshot-CNN captures visual patterns from webpage screenshots, and DOM-GCN models structural relationships within DOM trees. In addition, two bimodal models, BiLSTM-GCN and CNN-GCN, combine sequential–structural and visual–structural features, respectively.

**Table 3. Architecture comparison of baseline models and the proposed framework**

| Model | URL | HTML | DOM | IMG | Fusion |
|---|---|---|---|---|---|
| URL-CNN | ✓ | ✗ | ✗ | ✗ | ✗ |
| HTML-BiLSTM | ✗ | ✓ | ✗ | ✗ | ✗ |
| Screenshot-CNN | ✗ | ✗ | ✗ | ✓ | ✗ |
| DOM-GCN | ✗ | ✗ | ✓ | ✗ | ✗ |
| BiLSTM-GCN | ✗ | ✓ | ✓ | ✗ | ✗ |
| CNN-GCN | ✗ | ✗ | ✓ | ✓ | ✗ |
| Proposed | ✗ | ✓ | ✓ | ✓ | ✓ |

The architectural differences between these baseline models and the proposed BiLSTM–GCN–CNN framework are summarized in Table 3.

### *4.3. Evaluation Metrics*

The classification performance of the proposed framework and all baseline models was evaluated using four standard metrics, namely Accuracy, Precision, Recall, and F1-score. These metrics are defined as follows:

$$\text{Accuracy} = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c+FP_c+FN_c)} \quad (5)$$

$$\text{Precision}_c = \frac{TP_c}{TP_c+FP_c} \quad (6)$$

$$\text{Recall}_c = \frac{TP_c}{TP_c+FN_c} \quad (7)$$

$$\text{F1}_c = \frac{2\times\text{Precision}_c\times\text{Recall}_c}{\text{Precision}_c+\text{Recall}_c}$$

Where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively [27].

### *4.4. Comparative Performance Analysis*

The detection results for the proposed multimodal framework and baseline methods are presented in Table 4. As seen from the table, single-modal methods such as URL-CNN, HTML-BiLSTM, and Screenshot-CNN obtain less classification accuracy because they cannot fully utilize the complicated features in the modern phishing webpages. Hybrid models exploiting the joint use of two modalities achieve low, yet acceptable performance, but still miss the presentation of phishing multi-modal anomalies.

It can be observed that compared to other approaches, the proposed BiLSTM–GCN–CNN multimodal model has superior classification performance in terms of accuracy and F1-score, which verifies that integrating text information, network structural information, and visual perception messages into a single detection model is effective.

**Table 4. Performance comparison of baseline models and the proposed framework**

| Model Name | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| URL-CNN | 93.12 | 92.48 | 92.65 | 92.56 |
| HTML-BiLSTM | 94.36 | 93.89 | 94.02 | 93.95 |
| Screenshot-CNN | 95.02 | 94.61 | 94.88 | 94.74 |
| DOM-GCN | 94.21 | 93.74 | 93.91 | 93.82 |
| BiLSTM-GCN | 96.14 | 95.87 | 96.03 | 95.95 |
| CNN-GCN | 96.47 | 96.02 | 96.29 | 96.15 |
| Proposed Model | 98.93 | 98.71 | 98.85 | 98.78 |

The ROC curves for all the tested models are shown in Figure 7, on which the proposed model achieves the maximum area under the curve value, demonstrating better discriminative power than baseline methods. Furthermore, the proposed method shows stable convergence trends during training epochs, which demonstrates learning efficiency and less overfitting.

All these performance improvements happen because the URL patterns, dependencies in HTML structure, and visual layout cues serve to encode phishing representation more effectively. These results indicate how multi-modal-based learning has proved to be beneficial in generalizing and enhancing the correctness of detection on both web interfaces.

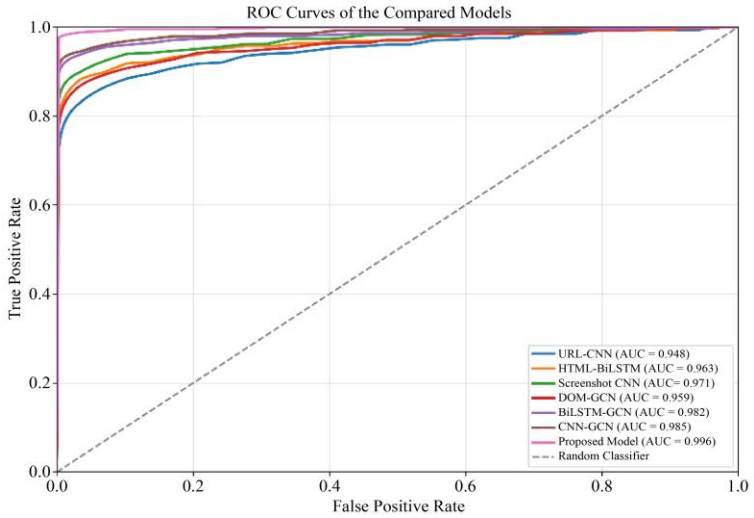


**Fig. 7 ROC curves of the compared models**

### *4.5. Impact of ACO Optimization*

The convergence behavior of the ACO process is shown in Figure 8, indicating stable optimization across iterations. Table 5 compares performance before and after ACO-based tuning. The optimized model achieves faster convergence and improved generalization.

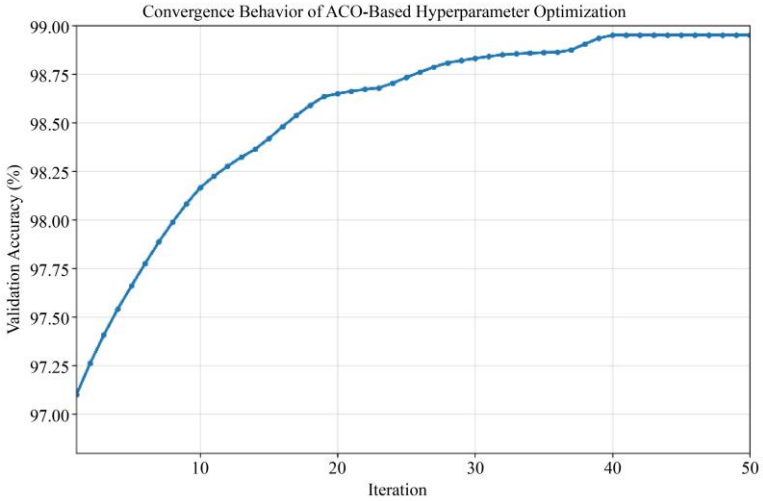


**Fig. 8 Convergence behavior of the ACO-based hyperparameter optimization**

**Table 5. Impact of ACO-based hyperparameter optimization**

| Model Configuration | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| Proposed Model (Before ACO) | 97.21 | 96.84 | 97.03 | 96.93 |
| Proposed Model (After ACO) | 98.93 | 98.71 | 98.85 | 98.78 |

### *4.6. Explainability Evaluation*

Meanwhile, the SHAP-based explanation module interprets the contribution spirit of how much each input feature drives the output prediction as displayed in Table 6. As illustrated in Figure 9, suspicious HTML entities, irregular DOM structures, and deceptive visual shapes in phishing webpages are highly important tokens according to SHAP visualization. This demonstrates that the model can find and leverage discriminative features consistently over diverse modalities. Additionally, the explanations offered are human-interpretable: they explain how the model arrived at its decisions. This transparency is a critical step to further proving the trustworthiness of the system and assisting in its deployment under real-world cybersecurity conditions.

Additionally, we see reproducibility of feature importance on samples, which further confirms the strength of representations learned. The multimodal interpretation also permits security analysts to trace back the cause-and-effect of phishing predictions confidently. SHAP offers the interpretability and trust, especially important for security applications, so we use it as an additional component to improve our proposed framework.

**Table 6. Contribution of each feature modality based on SHAP analysis**

| Modality | Top Contributing Features | Mean \|SHAP\| |
|---|---|---|
| HTML | form_action_mismatch, suspicious_keywords, login_form_presence | 0.076 |
| DOM | hidden_iframes, deep_dom_tree, external_resource_ratio | 0.077 |
| Visual | brand_logo_similarity, visual_layout_similarity, padlock_icon_misuse | 0.085 |

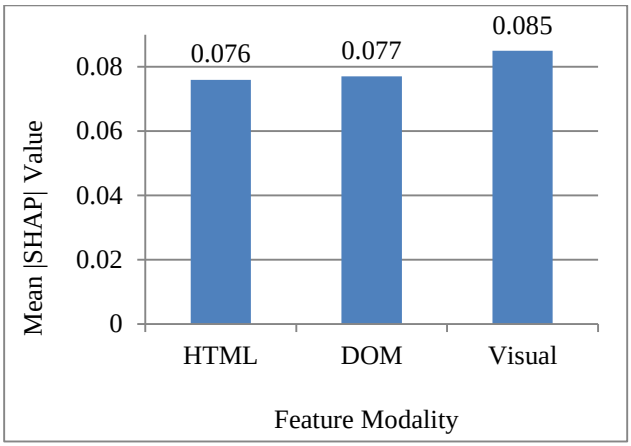


**Fig. 9 SHAP-based feature importance across modalities**

### *4.7. Comparison with State-of-the-Art Methods*

Our proposed framework was compared against recent phishing detection methods [13, 14, 16, 28], such as Phishpedia [14], HTMLPhish [13], and PhishGNN [16]; a multimodal method based on DOM. Although Phishpedia captures visual similarity, HTMLPhish focuses on the actual HTML content and user-visible features, while PhishGNN applies a GNN structure over the DOM. While these methods show effectiveness, they are either constrained to unimodal or bimodal feature learning and lack interpretability or intelligent hyperparameter tuning. Rather than that, we presented an end-to-end BiLSTM–GCN–CNN model where HTML, DOM, and visual features are combined in one structure that's capable of optimizing detection results using ACO-based optimization. Furthermore, we explain interpretable phishing prediction using SHAP. As shown in Table 7, the proposed model outperforms all comparative state-of-the-art models overall in terms of accuracy and F1-score, showcasing its healing power for performance increase.

**Table 7. Comparison with state-of-the-art phishing detection methods**

| Ref. | Method | Modalities Used | Optimization | Explainability | Accuracy (%) |
|---|---|---|---|---|---|
| **[14]** | Phishpedia | Screenshot (Visual) | None | No | 95.3 |
| **[13]** | D-PhishNet | URL + HTML | None | No | 97.0 |
| **[16]** | PhishGNN | DOM Graph | None | No | 96.2 |
| **[28]** | DOM-Multimodal | HTML + DOM | None | No | 97.05 |
| **Proposed Model** | BiLSTM–GCN–CNN + ACO + SHAP | HTML + DOM + Screenshot | ACO | SHAP | 98.93 |

### *4.8. Discussion*

Experimental results indicate that the proposed BiLSTM–GCN–CNN multimodal network outperforms all baseline models that use either single modality or a combination of bimodality. This performance improvement mostly comes from the excellent fusion of heterogeneous representation features. Specifically, using the BiLSTM component, we are able to preserve sequential dependencies

present in HTML contents, which allows us to look for potentially suspicious lexical patterns; and with the aid of the GCN module, we construct a structural relationships DOM (Document Object Model) tree, thus over-finding irregular hierarchical behaviors. The CNN extracts visual layout features from a web page screenshot that aid in identifying misleading interface designs. These complementary modalities allow the model to learn complex features that single approaches may not capture.

As exhibited in Table 5 and further depicted graphically in figure, the results in Figure 8 confirm that the ACO is able to enhance both convergence stability and detection accuracy compared to previous works. This is because it is indeed exploring the hyperparameter search space effectively without evaluating bad configurations. This allowed the final model to reach convergence quickly and demonstrate strong generalization performance, suggesting that such collaborative hyper-parameter-based tuning strengthens high-level multimodal-learning systems.

Additionally, the explainability analysis (SHAP) was performed in Table 6 and Figure 9, which gives us a more interpretable decision-making process of the model. Detection performance is largely driven by evil HTML tokens, weird DOM structure, and visual deception. The model will not learn noise, which fosters transparency (in practice) and confidence regarding cybersecurity threats received because of training on meaningful, interpretable features in the data.

The comparison with the state-of-the-art methods (Table 7) confirms that multi-modal feature learning, optimization strategies, and interpretable artificial intelligence are beneficial to the framework designed. While traditional methods either use one kind of data or restrict the feature personnel, which makes them lack complete evidence to detect phishing websites in different environments, this approach uses different aspects of complementary information to complement each other, thereby increasing detection ability. Anyway, the suggested framework is also very powerful against different kinds of phishing attacks, which could mean it has good generalization ability. The system utilizes a combination of sequential modeling, graph-based reasoning, and visual input extraction to not only detect syntactical manipulation techniques previously known but also more complex semantic or even visual-based manipulations. The results confirm the effectiveness and real-world feasibility of our proposed architecture in web security contexts.

## 5. Conclusion

We presented a multi-modality deep learning model for jointly analyzing HTML content along with the DOM structural relationships between multiple elements/features on the page and visual webpage characteristics based on BiLSTM, GCN, and CNN architecture for phishing detection. By taking advantage of the complementary nature of these feature modalities, there is still an effective detection pattern in this context, with several weaknesses in the usual single-modal methods.

To improve detection robustness and training stability, Ant Colony Optimization (ACO) was used to automatically tune hyperparameters for faster convergence and a better generalization capability. Furthermore, we applied SHAP-based explainability to explain the model prediction in a human-readable format and help security analysts understand how these features, including HTML, DOM, and visual impact, contribute to each prediction for phishing status.

We present a comprehensive series of experimental results, where we further show how our approach compares with recent state-of-the-art methods and demonstrate the effectiveness of the proposed approach on accuracy as well as F1-score measures while retaining a high level of interpretability.

These results confirm the usefulness of this framework in real-world anti-phishing systems, and it provides a utility-centric path towards interpretable, adaptive system designs for resisting adversarial attacks. This framework will further be used in a practical way in a browser-edge security toolkit designed to inform the detection of such malicious behavior at run time by a more general phishing training dataset or cross-language phishing examples in the future.

## References

[1] Muhammad Usman Javeed et al., "Phishing Website URL Detection Using A Hybrid Machine Learning Approach," *Journal of Computing & Biomedical Informatics*, vol. 9, no. 1, pp. 1-9, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[2] Bryan Lim et al., "EXPLICATE: Enhancing Phishing Detection through Explainable AI and LLM-Powered Interpretability," *arXiv Preprint*, pp. 1-9, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[3] Eleni Kytidou et al., "Machine Learning Techniques for Phishing Detection: A Review of Methods, Challenges, and Future Directions," *Intelligent Decision Technologies*, vol. 19, no. 6, pp. 4356-4379, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Saif Wali Ali Alsudani et al., "Enhancing Spam Detection: A Crow-Optimized FFNN with LSTM for Email Security," *Wasit Journal of Computer and Mathematics Science*, vol. 3, no. 1, pp. 28-39, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[5] Alexey Vulfin et al., "A Multimodal Phishing Website Detection System using Explainable Artificial Intelligence Technologies," *Machine Learning and Knowledge Extraction*, vol. 8, no. 1, pp. 1-43, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[6] Alexey Vulfin et al., "Do We Really Need Reference-Based Phishing Detection? Unleashing the Power of GNN," *2025 9th Network Traffic Measurement and Analysis Conference (TMA)*, Copenhagen, Denmark, pp. 1-4, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[7] Husnain Mansoor Butt et al., "Detecting Phishing URLs Using Hybrid Deep Learning Models," *International Journal for Electronic Crime Investigation*, vol. 9, no. 1, pp. 1-14, 2025. [CrossRef] [Publisher Link]

[8] Brahim Khalil Sedraoui et al., "Cybersecurity in E-Learning: A Literature Review on Phishing Detection Using ML and DL Techniques," *2025 International Conference on Networking and Advanced Systems (ICNAS)*, El Tarf, Algeria, pp. 1-10, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Zeinab Shahbazi, Rezvan Jalali, and Maryam Molaeevand, "AI-Based Phishing Detection and Student Cybersecurity Awareness in the Digital Age," *Big Data and Cognitive Computing*, vol. 9, no. 8, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[10] Alsadig Hadi Alsadig, and Md Oqail Ahmad, "Phishing URL Detection Using Deep Learning with CNN Models," *2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, Coimbatore, India, pp. 768-775, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] J. Govindaraaj, "The Role of Explainable AI in Understanding Phishing Susceptibility," *Journal of Recent Trends in Computer Science and Engineering*, vol. 12, no. 1, pp. 1-6, 2024. [Google Scholar] [Publisher Link]

[12] Qazi Emad ul Haq, Muhammad Hamza Faheem, and Iftikhar Ahmad, "Detecting Phishing URLs Based on a Deep Learning Approach to Prevent Cyber-Attacks," *Applied Sciences*, vol. 14, no. 22, pp. 1-17, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Haifeng Jiang et al., "D-PhishNet: A Dual-Branch Network for URL and HTML Feature Fusion in Phishing Webpage Detection," *Computer Networks*, vol. 271, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[14] Yun Lin et al., "Phishpedia: A Hybrid Deep Learning Based Approach to Visually Identify Phishing Webpages," *30th USENIX Security Symposium (USENIX Security 21)*, pp. 3793-3810, 2021. [Google Scholar] [Publisher Link]

[15] Ruofan Liu et al., "Inferring Phishing Intention via Webpage Appearance and Dynamics: A Deep Vision Based Approach," *31st USENIX Security Symposium (USENIX Security 22)*, pp. 1-18, 2022. [Google Scholar] [Publisher Link]

[16] Tristan Bilot, Grégoire Geis, and Badis Hammi, "PhishGNN: A Phishing Website Detection Framework using Graph Neural Networks," *Proceedings of the 19th International Conference on Security and Cryptography Secrypt*, Lisbon, Portugal, vol. 1, pp. 428-435, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Khandaker Mohammad Mohi Uddin et al., "Explainable Machine Learning for Phishing Site Detection: A High-Efficiency Approach using Boosting Models and Shap," *The Journal of Engineering*, vol. 2025, no. 1, pp. 1-18, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] Scott M. Lundberg, and Su-In Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Google Scholar] [Publisher Link]

[19] Sakib Shahriar Shafin, "An Explainable Feature Selection Framework for Web Phishing Detection with Machine Learning," *Data Science and Management*, vol. 8, no. 2, pp. 127-136, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[20] Maria Carla Calzarossa, Paolo Giudici, and Rasha Zieni, "An Assessment Framework for Explainable AI with Applications to Cybersecurity," *Artificial Intelligence Review*, vol. 58, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[21] Shaimaa Ahmed Elsaid et al., "Hybrid Intrusion Detection Models based on GWO Optimized Deep Learning," *Discover Applied Sciences*, vol. 6, pp. 1-34, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Santosh Kumar Birthriya, Priyanka Ahlawat, and Ankit Kumar Jain, "Intelligent Phishing Website Detection: A CNN–SVM Approach with Nature-Inspired Hyperparameter Tuning," *Cyber Security and Applications*, vol. 3, pp. 1-12, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[23] Faiz Akram et al., *Integrating Artificial Bee Colony Algorithms for Deep Learning Model Optimization: A Comprehensive Review*, Solving with Bees, pp. 73-102, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Fujiao Ji et al., "Evaluating the Effectiveness and Robustness of Visual Similarity-Based Phishing Detection Models," *34th USENIX Security Symposium (USENIX Security 25)*, 2025. [Google Scholar] [Publisher Link]

[25] Padmalochan Panda, Alekha Kumar Mishra, and Deepak Puthal, "A Novel Logo Identification Technique for Logo-Based Phishing Detection," *Future Internet*, vol. 14, no. 8, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[26] Saad Al-Ahmadi, "A Deep Learning Technique for Web Phishing Detection Combined URL Features and Visual Similarity," *International Journal of Computer Networks & Communications*, pp. 1-14, 2020. [Google Scholar] [Publisher Link]

[27] Saif Wali Ali Alsudani, and Ghassan Khudair Saud, "Recurrent Neural Network Optimized by Grasshopper for Accurate Audio Data-Based Diagnosis of Parkinson's Disease," *Wasit Journal for Pure Sciences*, vol. 4, no. 2, pp. 56-75, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[28] Jun-Ho Yoon, Seok-Jun Buu, and Hae-Jung Kim, "Phishing Webpage Detection via Multi-Modal Integration of HTML DOM Graph Modeling and URL Feature Analysis," *Electronics*, vol. 13, no. 16, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]