International Journal of Engineering Trends and Technology
ISSN: 2231-5381/ https://doi.org/10.14445/22315381/IJETT-V74I11P120

Volume 74 Issue 11, 275-303, November 2026
© 2026 Seventh Sense Research Group®

Original Article

PhishTrustXAI: A Trust-Aware Evaluation Framework for Explainable Multimodal Phishing Detection in Evolving IoT Environments

Gaddam Lakshmi[1], P Swetha[2]

[1,2]*CSE Department, JNTUH, Kukatpally, Hyderabad, India.*

[1]*Corresponding Author: laxmi81.gaddam@gmail.com*

Received: 06 August 2026 Revised: 12 September 2026 Accepted: 18 September 2026 Published: 25 November 2026

Abstract *- Phishing attacks evolve to use fake URLs, realistic web pages, spoofed domain metadata, and AI-generated content to outsmart traditional security systems in IoT and edge environments. Recently, multimodal deep-learning methods have been proposed to fuse evidence from different modalities to improve detection accuracy further. However, most existing methods are tested on clean, in-domain datasets, and evaluation of adversarial robustness, explanation reliability, cross-dataset generalisation, and deployment trustworthiness remains limited. To address these limitations, this study introduces a trust-aware evaluation framework (PhishTrustXAI) to evaluate candidate explainable multimodal phishing detection systems based on four complementary aspects: predictive performance, adversarial robustness, explanation reliability, and generalisation across databases. It performs controlled URL obfuscation, visual spoofing, manipulation of metadata, and multimodal attacks in a coordinated manner; quantifies explanation fidelity, stability, and modality consistency; examines cross-dataset generalisation; evaluates the normalised performance, robustness, explanation, and generalisation dimensions; and integrates them into a composite Trust Score that is used for model ranking and deployment-oriented evaluation. Experiments on clean data on PhishTank, URLNet and LogoDet-3K/WebLogo-2M show that the adaptive multimodal detector attains 98.12% accuracy, 97.96% F1-score, and 99.14% ROC-AUC. It achieves 90.84% robust accuracy in medium-severity coordinated attacks, scores 0.902 in the explanation task and 0.959 in the generalisation task, and a final score of 0.945 in the TrustScore stable ranking in statistical and sensitivity analyses. PhishTrustXAI offers an approach to deployment-oriented comparisons and selection of trusted phishing detectors; the reported model size, latency, memory, and throughput indicate suitability for resource-constrained edge and IoT-oriented environments. However, the deployment results are shown in a controlled evaluation, not on physical IoT devices.*

Keywords *- Multimodal phishing detection, Explainable Artificial Intelligence, Adversarial robustness, Trust-Aware evaluation, Internet of Things.*

1. Introduction

Phishing remains one of the most serious cybersecurity threats on the web, IoT, edge and cyber-physical environments. Attackers now use a combination of deceptive URLs, visually convincing webpages, domain manipulation, AI-generated content, and more to bypass traditional security controls. However, traditional blocklist, rule-based, and handcrafted-feature-based methods cannot detect new attacks, and many deep-learning systems are limited to a single input type. They are mainly assessed using clean-data accuracy [1, 2]. While multimodal models are effective at detecting phishing by jointly considering textual, visual, and contextual evidence, their trustworthiness when operationally manipulated, distributed in different ways, and attacked with evolving behaviour has not yet been adequately studied [3, 4]. Recent studies on phishing and other cybersecurity applications include deep URL representations, webpage-image classification, multimodal attention, adversarial robustness, and Explainable Artificial Intelligence (XAI) [5-8]. Previous research has shown that multimodal fusion enhances detection accuracy by leveraging complementary information, and that attribution and saliency can help explain a model's decisions [9, 10]. Most evaluations, however, only assess accuracy, precision, recall, or F1-score; they do not determine whether a highly accurate model is robust, explainable, transferable, and reliable when deployed under real conditions. Strong in-domain results may significantly underestimate model reliability due to adversarially manipulated URLs, visibly modified brand elements, hidden metadata, and/or differences across the datasets used for distribution [11, 12]. Thus, multimodal phishing detection in a dynamic IoT/edge environment requires a common trust-

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

based evaluation scheme. This research introduces a trust-aware evaluation framework for explainable multimodal phishing detection systems, namely PhishTrustXAI. The research evaluates candidate detectors from four complementary perspectives: predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation. The main innovations are modality-specific adversarial stress testing and quantitative explanation validation via fidelity, stability, and modality consistency. The results show that while predictive performance, adversarial robustness, and explanation reliability are important, cross-dataset generalisation is also important, which we did not measure in our previous work. The results demonstrate that, in addition to predictive performance, adversarial robustness and explanation reliability, cross-dataset generalisation is likewise important, something we hadn't evaluated in our earlier paper. The contributions are fourfold; the main ones are as follows.

First, a single evaluation framework is designed to evaluate multimodal phishing detectors beyond traditional accuracy-based comparisons.

Second, we provide a controlled adversarial testing environment for URL obfuscation, visual spoofing, metadata manipulation, and coordinated multimodal attacks, spanning multiple levels of attack severity.

Third, we measure the reliability of explanations quantitatively and qualitatively with examples, so we can assess whether the highlighted URL tokens, webpage regions, and metadata attributes actually affect the model prediction and remain stable under perturbation.

Fourth, a TrustScore-based benchmarking procedure is proposed to rank the candidate models based on performance, robustness, explainability, and generalisation, using confidence intervals, statistical significance testing, weight-sensitivity analysis, and deployment suitability assessment. PhishTrustXAI does not propose a new architecture for phishing detection; rather, its novelty lies in the trust-oriented composite evaluation. Proposed TrustScore combines four complementary dimensions-predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation-and provides a common 0-1 scale for comparative assessments of candidate phishing detectors. Unlike other multi-criteria evaluation methods that combine independent performance measures for general decision-making, the TrustScore is designed to capture the reliability of phishing detection systems under adversarial perturbations, explanation requirements, and dataset distribution shifts. Therefore, the TrustScore gives a normalized view of the reliability of the candidate detectors based on trustworthiness and not just clean-data predictive capability.

The rest of this paper is structured as follows. Section 2 summarises the studies related to multimodal phishing detection, adversarial robustness, explainable cybersecurity, cross-domain generalisation, and trust evaluation. Section 3 presents the proposed methodology, datasets, evaluation protocol, robustness assessment, TrustScore formulation, and benchmarking algorithm. Section 4 presents clean-data performance, efficiency, adversarial, explainability, cross-dataset, and statistical results. Section 5 discusses study limitations and findings. The paper ends in Section 6 with a list of future research directions.

2. Related Work

Over the years, phishing detection has evolved from blocklist and rule-based detection to machine-learning and deep-learning techniques that can detect lexical, host, webpage, network and behavioural features. However, traditional phishing URL classifiers still struggle with adversarial perturbations, URL obfuscation, distribution changes, noisy inputs, and changes in phishing campaign patterns [1]. The same reliability issues have been encountered with deep anomaly-detection systems, which have inspired efforts to integrate adversarial robustness and explainable artificial intelligence into cybersecurity models [2]. Comparative studies of interpretable white-box models and high-performing black-box models also show that high detection accuracy alone is not enough; systems need to make transparent and verifiable decisions for operations [3]. Explainable deep ensembles have thus been investigated to integrate complementary spatial, temporal, and reconstruction-based representations and enhance the explainability of cybersecurity alerts [4].

Feature representation remains the main focus of phishing classification. Explainable feature-selection methods perform importance-guided refinement to filter out redundant URL, host, and content variables while retaining discriminative information [5]. Lexical, DNS, certificate, HTML, network and visual indicators offer complementary evidence, as reflected in broader reviews. However, manually engineered features may become outdated as phishing attacks evolve [6]. Hence, XAI assessment frameworks have suggested measuring explanations by means of faithfulness, stability, consistency, sparsity, robustness, usability, and domain relevance, rather than visual interpretation alone [7]. Large language models have also been explored for semantic phishing analysis, but they may suffer from self-consistency, faithfulness and robustness issues when dealing with obfuscated or adversarial inputs [8]. Deep sequence learning and XAI are increasingly used together in research on IoT phishing and language-based explanation. CNN-LSTM models have been trained to extract spatial and temporal attributes from IoT traffic, URLs, and payloads, and SHAP and LLM-generated rationales have been provided to help analysts interpret the results [9]. In this direction, Federated LLM-based mechanisms enable privacy-preserving collaborative phishing detection without exchanging raw client information [10]. Multimodal webpage models connect

DOM graphs and URL sequences using graph convolution and transformer-based cross-attention [11], and URL-HTML feature integration proved that structural webpage signals can complement lexical signals [12]. Keeping detection performance in step with evolving attacks is a major challenge. Relying on replay, knowledge distillation, and parameter regularisation, continual learning methods can limit catastrophic forgetting in sequential phishing campaigns [13]. A drift-detection framework integrates statistical tests, model-error monitoring, uncertainty estimation, and distribution-divergence measures to detect changing data conditions [14].

However, evaluating drift-handling approaches raises open questions about false alarms, adaptation delay, benchmark standardisation, and resource usage [15]. Federated continual phishing models offer distributed adaptation for non-independent client data [16], and related intrusion-detection research validates the importance of continual learning in network security challenges [17]. Recent phishing reviews also identify adversarial resilience, temporal adaptability, explainability, and privacy as key research directions [18]. Drift-aware federated malware detection is another example that shows the need to consider drift in client-specific changes that should be addressed in the distributed security learning process [19].

Explainability is also a key aspect in heterogeneous IoT and cyber-physical environments. For IoT systems connected to cloud services, XAI can support phishing classifiers [20]. In contrast, for cross-layer IoMT detection, physical, network, and application indicators can provide context-sensitive explanations [21]. Other architectures and systems, such as large-scale cyber-threat frameworks [22] and streaming IoT intrusion detectors [23], also illustrate the need for real-time, understandable security decisions. Extensive attribution datasets can be used to analyse a variety of IoT attacks and forensic evidence [24], and hybrid feature selection and explainable ensembles can enhance resistance to noisy and heterogeneous traffic [25].

Multimodal phishing studies compare URL, textual, visual, HTML, and network representations [26]. Temporal Graph Fusion models the relationships between different domains, web pages, resources, and campaign evolution [27], while multimodal agents use visual and textual reasoning to minimise inference costs [28]. Similarly, CNN-based feature fusion has also enhanced malicious website identification by fusing multiple evidence sources [29]. Hybrid content and behavioural analysis [30] is needed for specialised phishing forms such as misleading online ads.

Developing intelligent security monitoring is made difficult by resource constraints, protocol diversity, edge deployment, and dynamic attacks, as evidenced by research in connected vehicles [31], microgrids [32], and industrial IoT systems [33]. Social-engineering research also focuses on semantic manipulation and deception toward the user [34]. Reinforcement-learning methods provide adaptive phishing-response policies [35], and neuro-symbolic models integrate both learned representations and domain rules to provide more actionable explanations [36]. Interpretable phishing email platforms [37] and deep URL classifiers [38], such as CNN, LSTM, and hybrid CNN-LSTM [39], have shown good predictive performance. However, such performance is normally limited to accuracy, robustness, interpretability, and temporal adaptation. Evaluations of IoT detection botnets also indicate a lack of trust, unreliable explanations, multimodal robustness, and evaluation of emerging threats [40]. Thus, a common framework for trust assessment is needed to evaluate predictive performance, adversarial resilience, cross-dataset generalisation, explanation fidelity, and adaptation to phishing evolution in heterogeneous IoT environments.

Table 1. Comparative analysis of existing phishing detection and evaluation approaches

Study/Approach	Multimodal	Adversarial Robustness	Explainability	Cross-Dataset Generalization	Composite Trust Evaluation	Main Limitation
Conventional ML phishing detectors	Limited	Limited	Usually not evaluated	Limited	No	Mainly focuses on predictive performance
Deep-learning phishing detectors	Partial	Limited	Limited	Limited	No	High benchmark performance may not reflect robustness
Multimodal phishing detection	Yes	Partial	Partial	Varies	No	Primarily focuses on detection accuracy and multimodal fusion
Explainable phishing detection	Partial/Yes	Limited	Yes	Limited	No	Explanation quality is generally evaluated separately from robustness

Adversarial robustness studies	Usually URL-focused	Yes	Limited	Partial	No	Focuses mainly on attack resistance
Cross-dataset evaluation studies	Varies	Partial	Limited	Yes	No	Generalization is usually treated as a separate evaluation criterion
PhishTrustXAI (this work)	Candidate multimodal detectors	Yes	Yes	Yes	Yes - TrustScore	Provides an integrated trust-oriented evaluation framework

As indicated by the comparative analysis, most of the studies on phishing detection focus on a single and/or combination of evaluation dimensions, including, but not limited to, predictive performance, multimodal detection, adversarial robustness, explainability, and cross-dataset generalisation. In the past, these dimensions have often been considered in isolation rather than integrated into a single trust assessment. Hence, the specific gap filled by PhishTrustXAI is the lack of a single evaluation framework that jointly quantifies all the above factors and rolls them out into a single metric, dubbed the TrustScore, for unified comparison and ranking of the candidate phishing detectors. PhishTrustXAI is not a new phishing detection architecture or framework, but it is an evaluation and benchmarking framework that fills this void.

3. Materials and Methods

This section proposes the PhishTrustXAI methodology to evaluate multimodal phishing detection systems in the context of trust in an evolving IoT environment. It outlines the general evaluation system; how to prepare and assess multimodal datasets; how to perform adversarial robustness analysis; how to compute the trust score; and how to perform comparative benchmarking. Finally, the full trust-aware evaluation algorithm is provided to systematically evaluate the model reliability, robustness, explainability and deployment readiness. The evaluation framework is not the same as the detector being evaluated. One of the candidate models (phishing detector) adopted in the experiments is the adaptive multimodal detector. At the same time, PhishTrustXAI is used to specify the evaluation protocol and TrustScore for evaluating the candidate phishing detectors. The detector categorises different types of phishing and illustrates the PhishTrustXAI assessment of the detector's trustworthiness across four dimensions.

3.1. Trust-Aware Evaluation Framework

The PhishTrustXAI framework proposed in this paper systematically analyses the trustworthiness of multimodal phishing detection systems in changing IoT and edge scenarios. The proposed framework evaluates classification accuracy, adversarial robustness, explanation accuracy, and cross-dataset generalisation altogether. These evaluation dimensions are combined into a single workflow to assess the reliability of a phishing detector for real-world use in a dynamic, resource-limited cybersecurity setting. The framework is fed multimodal phishing samples in the form of URL text, webpage screenshots, and domain-level metadata. These modalities align with the input structure used in LightPhishAI, which comprises textual, visual, and contextual features processed by lightweight, modality-specific encoders and fused by adaptive attention. Instead of re-engineering or retraining the multimodal detector, PhishTrustXAI uses it as the evaluation target. This will eliminate redoing feature extraction and fusion, and will allow the methodology to focus on trust assessment.

The *i-th* multimodal phishing sample is modelled as,

$$X_i = (T_i, V_i, M_i), \tag{1}$$

The webpage screenshot V_i replaces the textual input T_i and the domain-level and contextual metadata M_i. The input defined in Equation (1) is fed to the pretrained multimodal detector, which outputs the phishing probability as

$$\hat{y}_i = f_\theta(X_i), \tag{2}$$

Here, the pretrained phishing detection model is represented by f_θ denotes the pretrained model with the learned parameters, and $\hat{y}_i \in [0, 1]$ denotes the predicted phishing probability. The corresponding binary decision is acquired by means of

$$y_i^p = \begin{cases} 1, & \hat{y}_i \geq \tau, \\ 0, & \hat{y}_i < \tau, \end{cases} \tag{3}$$

Where τ is the classification threshold, 1 is the phishing class, and 0 is the legitimate class. The detector output obtained as a result of Equations (2) and (3) is then sent to the PhishTrustXAI evaluation engine. The engine comprises four coordinated assessment modules. In the first module, the performance of the predictive method is assessed by means of accuracy, precision, recall, F1 score and area under the receiver operating characteristic curve. The second module conducts adversarial robustness testing of a URL under

obfuscation, homograph substitution, brand spoofing in the visual component, and metadata manipulation. The third module examines the reliability of explanations through modality-appropriate interpretation. Feature-attribution analysis is used to evaluate the URLs and metadata contributions, while visual saliency maps are used to analyse image-based contributions.

The fourth module assesses cross-dataset generalisation, that is, the detector's ability to perform well on training and testing data from different modality distributions or phishing repositories. The framework also incorporates the robustness and explainability mechanisms introduced in NeuroSymbolicPhishDefend, which involve analysing textual, visual, and symbolic/metadata evidence to produce explainable phishing decisions.

The neuro-symbolic reasoning process is not repeated in PhishTrustXAI. Instead, it checks whether the model predictions and explanations are faithful, stable, and consistent under clean, adversarial, and distribution-shifted settings. The evaluation architecture proposed is also suitable for detectors that are continually learned. The drift-aware workflow developed earlier monitors temporal phishing streams, detects warning and drift states, updates the detector using the replay feature, and redeploys the adapted detector. In PhishTrustXAI, the performance of the original model state and the adapted model state can be assessed individually to determine whether continual learning enhances detection performance while maintaining the stability of adversarial robustness, explanations, and previously learned knowledge.

Normalization is done for the outputs of the four evaluation modules before aggregation of trust. Let A represent the normalised predictive performance, R the normalised adversarial robustness, E the normalised explanation reliability, and G the normalised cross-dataset generalisation. The composed trust score is given by Equation (4).

$$T_s = w_A A + w_R R + w_E E + w_G G,\ w_A + w_R + w_E + w_G = 1, \quad (4)$$

w_A, w_R, w_E, and w_G represent the weights of the four trust dimensions. Weights can be equal for general benchmarking or vary by application, giving more importance to robustness or explanation reliability in high-risk IoT deployments. These formulations are not repeated in this subsection; the detailed calculation of the individual scores and their normalisation is presented in Section 3.4.

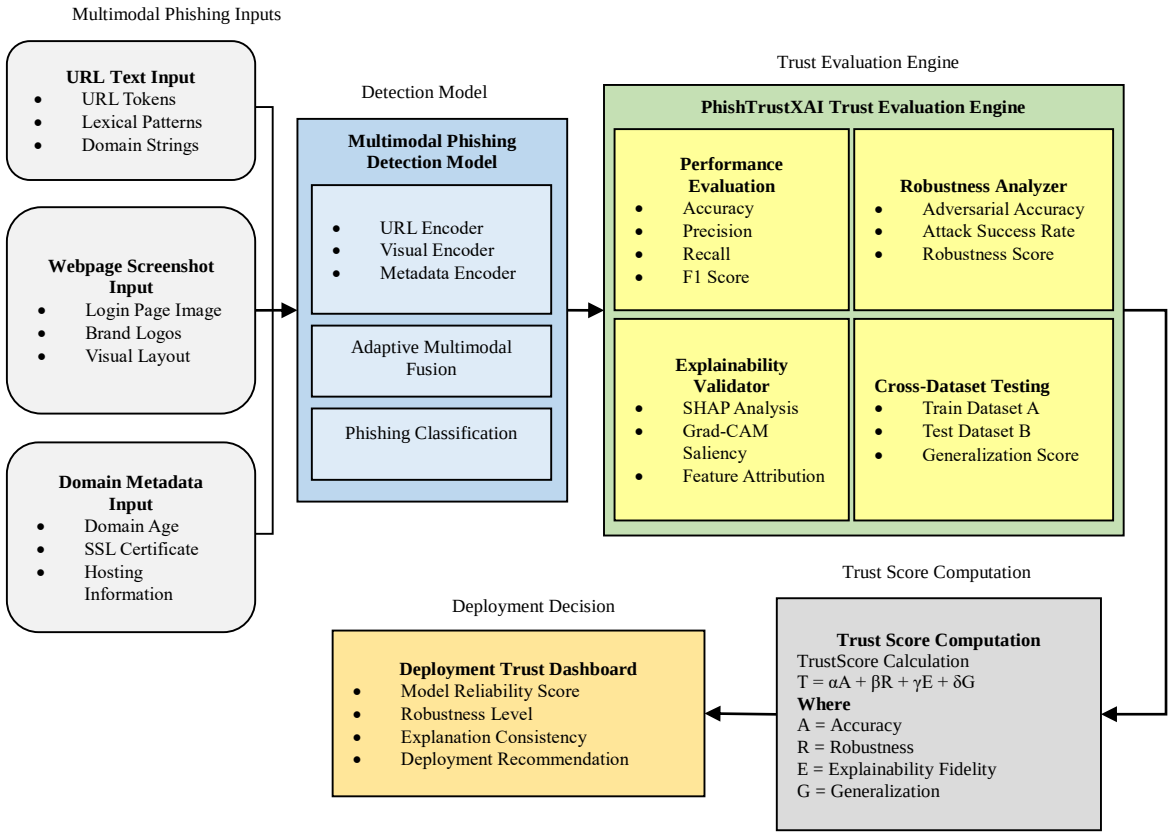

Fig. 1 PhishTrustXAI architecture for trust-aware evaluation of multimodal phishing detection systems

Figure 1 presents PhishTrustXAI's full trust-evaluation architecture. The workflow starts with the multimodal inputs of Equation (1), then passes through the pretrained detector to perform phishing detection, and then passes through the four evaluation modules to perform trust assessment; finally, a composite score is calculated using the four scores (as shown in Equation (4)), and a trust report is generated to be applied to deployment. The final output includes the model reliability level, robustness status, explanation-consistency assessment, cross-dataset stability and deployment recommendation. Table 2 provides an overview of the major mathematical symbols employed in the methodology proposed in this paper.

Table 2. Summary of mathematical notations used in the PhishTrustXAI framework

Notation	Description
X_i	i-th multimodal phishing sample
T_i	URL or textual representation of the i-th sample
V_i	Webpage screenshot or visual representation
M_i	Domain-level and contextual metadata vector
f_θ	Pretrained multimodal phishing detection model
θ	Learned parameters of the pretrained detector
$\hat{y}_i$	Predicted phishing probability
y_i^p	Predicted binary class label
y_i	Ground-truth class label
τ	Decision threshold for binary classification
D_c	Clean evaluation dataset
D_a	Adversarially perturbed evaluation dataset
D_s	Source dataset used for model training
D_t	Target dataset used for cross-dataset testing
A	Normalised predictive-performance score
R	Normalised adversarial-robustness score
E	Normalised explanation-reliability score
G	Normalized cross-dataset-generalization score
w_A, w_R, w_E, w_G	Weights assigned to the four trust dimensions
T_s	Composite trust score
Φ_i	Feature-attribution vector for the i-th sample
S_i	Visual saliency map generated for the i-th sample
ΔA	Detection-performance degradation under adversarial perturbation
ASR	Adversarial attack success rate
Fid	Explanation fidelity score
$Stab$	Explanation stability score

3.2. Multimodal Dataset Preparation and Evaluation Protocol

The experimental evaluation is based on three public datasets that offer complementary textual and visual data on phishing. PhishTank offers verified phishing URLs, URLNet offers structured URL samples for lexical and token-level analysis, and LogoDet-3K/WebLogo-2M offers real and spoof-relevant logo and webpage-interface images for visual analysis. Table 3 summarises the selected datasets and their roles in PhishTrustXAI.

Table 3. Public Datasets Used for Multimodal Trust-Aware Phishing Evaluation

Ref.	Dataset	Primary Modality	Data Used in PhishTrustXAI	Role in the Evaluation	Dataset URL
[41]	PhishTank	URL text	Verified phishing URLs, domain strings, paths, query components, and labels	Primary phishing URL evaluation, clean testing, adversarial URL generation, and temporal analysis	https://www.phishtank.com
[42]	URLNet Dataset	URL tokens	Character sequences, word-level tokens, domain components, and malicious/legitimate labels	Lexical benchmarking, token-level perturbation testing, and cross-dataset URL evaluation	https://github.com/duoergun0729/Phishing-Detection

[43]	LogoDet-3K / WebLogo-2M	Webpage and logo images	Brand logos, visual regions, interface elements, and logo annotations	Visual encoder evaluation, brand-spoofing simulation, saliency assessment, and visual robustness testing	https://github.com/WangYueFt/dataset-LogoDet-3K

The source dataset file provides a textual description of the URL resources (PhishTank, URLNet) and the visual resource (LogoDet-3K/WebLogo-2M). It does not offer a pre-aligned dataset with a URL, a webpage screenshot, and a metadata vector for each record. Thus, the datasets are not considered naturally paired multimodal triplets.

PhishTrustXAI, on the other hand, uses a controlled multimodal preparation protocol: URL records are standardised, visual records are prepared independently, and domain-level metadata, when available, is computed from URL samples using reproducible feature-extraction procedures. For every URL record, the raw address is converted to lowercase, but security-sensitive characters are maintained.

The URL is broken down into protocol, subdomain, registered domain, path, parameters and query. The character-level and token-level representations are preserved because many phishing URLs are misspelt, include lots of delimiters, encoded characters, deceptive subdomains and brand name terms. To minimise leakage and duplicates of URLs, records with missing labels, malformed records and exact domain name repetition across partitions are removed.

The visual samples taken from LogoDet-3K/WebLogo-2M are resized to the same resolution (224×224pixels) and normalised in the range (01.) Brands, logos, image size and annotations are maintained. Limited resizing, compression, brightness adjustment, and minor geometric variation are allowed as visual augmentation, but no other augmentation that changes the semantic label. Clean evaluation excludes strong transformations that might remove logos or change brand identity; these are reserved for adversarial stress testing.

For URL samples where domain information can be obtained, metadata will be derived. The metadata presented includes the number of subdomains, the count of redirects, secure-protocol information, certificate availability, domain age information (where available), and certain hosting or registration indicators.

All feature-availability masks indicate missing metadata values, rather than filling unavailable fields with arbitrary values. Numerical variables are normalised using statistics calculated in the training set, and categorical variables are encoded consistently across all evaluation partitions. A sample with a ready-made URL is shown in Equation (5).

$$X_i = (T_i, M_i), \tag{5}$$

Where T_i is is the processed URL representation, and M_i is the metadata vector associated with it. If there is an image of the corresponding webpage, or such an image can be reliably related to the same domain or brand, the sample is expanded to the multimodal representation defined above in Equation (1).

Samples without a proven image association are kept for URL-metadata assessment and not artificially attributed to non-corresponding images. Each dataset is partitioned so that 70%, 15% and 15% of the data are used for training, validation and testing, respectively, to ensure proper evaluation. When possible, we use domain-disjoint partitioning: urls with the same registered domain are not included in more than one subset.

To avoid duplicate or similar logos in the training and test sets, we apply brand-aware splitting to the logo images. The validation set is used for the selection of decision-threshold and the setting of the trust-weight and for model calibration, leaving the test set untouched until final evaluation.

The in-dataset protocol consists of training and testing a model with two different non-overlapping partitions of the same source. Cross-dataset testing involves training the URL branch on one URL dataset and testing it on another. The main URL-level transfer settings are PhishTank-to-URLNet and URLNet-to-PhishTank. LogoDet-3K/WebLogo-2M is a visual dataset and is therefore not used directly for record-wise transfer with the two URL datasets, but for independent visual generalisation, spoofing robustness, and explanation-quality evaluation. Cross-dataset generalisation is measured as in Equation (6).

$$G = \frac{F1_{D_t}}{F1_{D_s}}, \tag{6}$$

Where $F1_{D_s}$ and $F1_{D_t}$ Is the F1 score reached on the in-domain and target datasets, respectively. Values close to one indicate stable generalisation, while lower values indicate sensitivity to dataset shifts.

Accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC are used to measure predictive performance. Because phishing datasets may have unequal class distributions, F1-score and PR-AUC are considered primary classification measures. The F1-score is calculated as in Equation (7).

$$F1 = \frac{2PR}{P+R}, \tag{7}$$

Where P and R denote precision and recall, respectively, the following are used to evaluate calibration: expected calibration error, and the following are used for efficiency: inference latency, throughput, model size, memory usage, and estimated computational cost.

Classical and Deep-learning baselines, varying in modality and complexity, are included in the performance benchmarking. This comparison set includes a character-level URL classifier (URLNet), a visual CNN baseline, a metadata-based multilayer perceptron, a simple concatenation-based multimodal model, and the adaptive multimodal detector evaluated by PhishTrustXAI.

The partitions, preprocessing rules, decision thresholds, perturbation budgets and random seeds are the same across all models. Results are expressed as the mean and standard deviation from five independent runs, and statistical testing is done with paired testing between the proposed detector and the strongest baseline. Notably, PhishTrustXAI doesn't make the claim that the three source datasets are part of one single 62,000-record paired multimodal dataset. The datasets are supplementary evaluation resources; the sample counts include records from both sets.

A multimodal record exists only where such a relationship between a URL, a visual sample and a metadata vector can be reliably ascertained, and the visual sample is assessed on its own available modality without artificially pairing it with the other two elements. Duplicate sites and repeated registered domain names were excluded before partitioning to limit the chances of data leakage. Brands/screenshots and near duplications were also controlled, with some brand-level overlap minimised where reliable brand information was available. The metadata per shared domain was grouped in the respective partition. These controls were applied before the 70:15:15 train, validation, test split. The clean adversarial, cross-dataset, explanation, calibration, and efficiency results obtained using this protocol are then normalised and fed into the trust-score computation procedure described in Section 3.4.

3.2.1. Record-Level Multimodal Sample Construction

We used a specific preparatory process at the record level to build each experimental record, without making any correspondence assumptions across the independent source datasets. Initially, all the URL samples were given a corresponding record identifier and stored in the same repository as the corresponding ground-truth label.

The raw URL was lowercased, security-relevant characters were retained, and it was split into the protocol, subdomain, registered domain, path, parameters and query. Next, the processed URL was extracted at character-level and token-level representations.

Second, domain-level metadata was not created unless the necessary domain information was available. The metadata vector consisted of the number of subdomains, redirect count, data on the use of secure protocol, certificate availability, log information on domain age (if available), and some factors of its hosting or registration.

A feature-availability mask was used to represent missing metadata fields, rather than random imputation. Numerical features have been normalised by the statistics of the training partition, and categorical features have been coded uniformly across the training, validation and test partitions.

Third, visual information was added only when a visual sample (a webpage screenshot) or the webpage could be reliably associated with the same webpage, domain, or brand represented by the URL record. The size of the visual sample was made equal to 224 × 224 pixels, and it was normalized uniformly. Logo identity, visual-region annotations, and brand information were kept. Samples without a reliable URL-image association were not paired to that URL record but remained available for unimodal URL/metadata evaluation or visual evaluation on their own.

So, a multimodal record becomes $X_i(U_i, V_i, M_i, y_i)$where Ui is the processed representation of the URL, Vi is the visual representation associated with the URL if available, Mi is the metadata vector at the domain-level, and yi is the ground-truth class label. Those records that lack labels, have wrong URLs, duplicated URLs, or duplicated registered domains in different partitions were deleted based on the leakage-control protocol.

The obtained records were then split into training, validation, and test sets, according to a train: validation:test ratio of 70:15:15, with domain-disjoint partitioning performed when possible. To minimize the same or similar brands between samples, brand-aware splitting was used to split visual sample sets. We first kept the clean record as the reference instance for the adversarial evaluation. Later, the appropriate perturbation was used separately-URL obfuscation, visual spoofing/manipulation, or metadata modification.

The corresponding modality-specific perturbations were used for the coordinated attacks, and the original label was still used as ground truth. As such, adversarial samples were taken from identifiable clean records, NOT as additional observations unrelated to the clean records.

This record-level construction process is intended to provide each multimodal sample with a traceable URL, label, available metadata, and, where known, compatible visual evidence. It also prevents unsupported URL-image pairings and reduces data leakage between experimental partitions.

Table 4. Record-level multimodal sample construction

Construction Stage	Record-Level Operation	Output
1. URL acquisition	Assign record ID and retain original label	Raw URL + label
2. URL processing	Lowercase and parse URL components	Character/token representation
3. Metadata extraction	Derive available domain-level features	Metadata vector + availability mask
4. Visual association	Associate image only when the domain/brand correspondence is reliable	Visual representation
5. Quality control	Remove missing labels, malformed records and duplicates	Valid records
6. Partitioning	70:15:15 split with domain/brand-aware separation	Train/validation/test
7. Adversarial generation	Perturb clean records by modality	Adversarial records
8. Final multimodal record	Combine available U_i, V_i, M_i, y_i	Evaluation-ready record

3.3. Adversarial Robustness Assessment

The adversarial robustness assessment considers the performance of the multimodal phishing detector when attackers intentionally alter URL content, web page display, or domain metadata. This assessment is important because modern phishing campaigns often maintain the same malicious intent while changing observable properties to avoid automated detection.

PhishTrustXAI thus creates adversarial variants from the clean evaluation set and evaluates the model under both clean and perturbed variants following a consistent stress-testing protocol.

Let X_i denote the "clean" multimodal sample defined in Equation (1). An adversarial sample is created as in Equation (8).

$$X_i^{adv} = \mathcal{P}(X_i; \epsilon; \kappa), \tag{8}$$

Where $\mathcal{P}(\cdot)$ is the perturbation operator, ϵ is the perturbation budget, and k (Greek letter kappa) is the attack type. Each perturbed sample must keep the original phishing or legitimate label and change one or more observable modalities. The test set of the adversary is built as in Equation (9).

$$D_a = \{(X_i^{adv}, y_i)\}_{i=1}^{Na}, \tag{9}$$

The number of adversarial samples is N_a, and y_i is the original ground-truth label. Any perturbation that invalidates the webpage, makes the sample unreadable, or changes the webpage's actual class is not included in the final stress-test set.

3.3.1. URL Perturbation

The URL perturbation module mimics evasion techniques often employed in phishing attacks, such as lexical and structural evasion. Controlled transformations include character substitution, insertion, deletion, homoglyph replacement, misleading subdomain addition, path extension, delimiter manipulation, and encoding-based obfuscation. Words used in brand names can also be swapped with other words in the domain, path and query parts of a URL to produce valid but misleading URLs.

The perturbed form of T_i is the URL form obtained through the following procedure Equation (10).

$$T_i^{adv} = \mathcal{P}_T(T_i; \epsilon_T), \tag{10}$$

Where $\mathcal{P}_T$ is the textual perturbation function and ϵ_T is the number/percentage of altered characters/tokens. After perturbation, the registered-domain structure and label validity are verified. Three severity levels are considered: low-severity attacks change a few characters or delimiters, medium attacks change both lexical and structural elements, and high attacks change multiple characters and/or delimiters while maintaining URL validity.

3.3.2. Visual Spoofing

The visual spoofing module checks the detector's reliability by assessing whether the phishing page resembles a legitimate webpage layout or brand identity. Visual perturbations include replacing logos, scaling logos, moving logos, changing colours, compressing logos, changing backgrounds, moving buttons, overlaying text, and minor layout tweaks. These modifications mimic visual camouflage realistically but do not change the content of a webpage.

Adversarial visual input for the webpage image V_i is Equation (11).

$$V_i^{adv} = \mathcal{P}_V(V_i; \epsilon_V), \quad (11)$$

Where $\mathcal{P}_V$ is the visual transformation process, and ϵ_V is the intensity of the transformation. Small-scale pre-processing, like resizing, is not considered an adversarial transformation.

The stress test includes brand-directed changes, interface mimicking, and attention-switching changes. Generated images are tested to ensure principal phishing features remain discernible.

3.3.3. Metadata Manipulation

The metadata manipulation module mimics attackers who change or hide contextual clues related to bad domains. Features like secure-protocol status, certificate information, domain-age category, redirect count, hosting characteristics, registration attributes and feature-availability masks are subjected to perturbations. Only features that can be changed in real life, or hidden, are changed.

The manipulated metadata vector is given by Equation (12).

$$M_i^{adv} = \mathcal{P}_M(M_i; \epsilon_M; \mathcal{C}), \quad (12)$$

Here $\mathcal{P}_M$ is the metadata perturbation function, ϵ_M is a decision on the number of metadata attributes to be altered, and $\mathcal{C}$ are validity constraints.

These constraints ensure that categorical values are valid, numerical features remain within acceptable ranges, and logically incompatible feature combinations are avoided.

3.3.4. Multimodal Attack Scenarios

PhishTrustXAI considers coordinated multimodal attacks, in addition to single-modality attacks. In such cases, two to three modalities are manipulated at once to simulate adaptive attackers that obfuscate URLs, impersonate visually, and conceal metadata. The adversarial sample is represented as in Equation (13).

$$X_i^{adv} = \left(T_i^{adv}, V_i^{adv}, M_i^{adv}\right) \quad (13)$$

The evaluated perturbed configurations are URL-only, visual-only, metadata-only, URL-visual, URL-metadata, visual-metadata, and fully multimodal perturbations.

This design shows how robust the model is to a single modality and whether adaptive fusion remains reliable when multiple evidence sources are corrupted simultaneously.

3.3.5. Robustness Evaluation Pipeline and Metrics

During adversarial evaluation, we keep the same pretrained detector and classification threshold as for clean testing. However, the threshold is not re-optimised on adversarial samples because otherwise an unrealistically favourable estimate of robustness would be obtained.

The model is first fed with the clean test set. D_c and then by the adversarial set D_a defined in Equation (9).

At each sample, attack type, modality, and severity level, clean and adversarial predictions are compared.

Figure 2 depicts the full stress-testing pipeline. Cleaned multimodal samples are passed to modality-specific attack generators to create these perturbed URL, visual and metadata variants. The same multimodal phishing detector is used for both clean and adversarial samples.

These outputs are then analysed to report performance degradation, attack success, and modality-specific vulnerabilities. Robust accuracy is the percentage of adversarial samples correctly classified, as in Equation (14).

$$Acc_{adv} = \frac{1}{N_a}\sum_{i=1}^{N_a} \mathbb{I}\left(y_i^{p,adv} = y_i\right), \quad (14)$$

Where $y_i^{p,adv}$ is the prediction that is made for X_i^{adv} and $\mathbb{I}(\cdot)$ is the indicator function. The absolute performance degradation is estimated as in Equations (15)-(17).

$$\Delta Acc = Acc_{clean} - Acc_{adv}, \quad (15)$$

Where Acc_{clean} is the accuracy achieved on D_c. A smaller value of ΔAcc means more resistance to adverse manipulation.

Attack success rate is the percentage of originally correct predictions that are perturbed to incorrect predictions.

$$ASR = \frac{\sum_{i=1}^{N_a} \mathbb{I}\left(y_i^p = y_i \wedge y_i^{p,adv} \neq y_i\right)}{\sum_{i=1}^{N_a} \mathbb{I}(y_i^p = y_i)} \quad (16)$$

The robustness score is defined for trust aggregation; a normalised robustness value is obtained.

$$R = \frac{1}{2}\left[\frac{Acc_{adv}}{Acc_{clean}} + (1 - ASR)\right] \quad (17)$$

The adversarial resilience score in Equation (17) is in the range [01], with values closer to 1 indicating greater adversarial resilience.

Results are provided both for each modality and for each attack type and severity level, as well as for a macro-average robustness score.

The obtained R-value is then incorporated in the composite trust value as expressed in Equation (4).

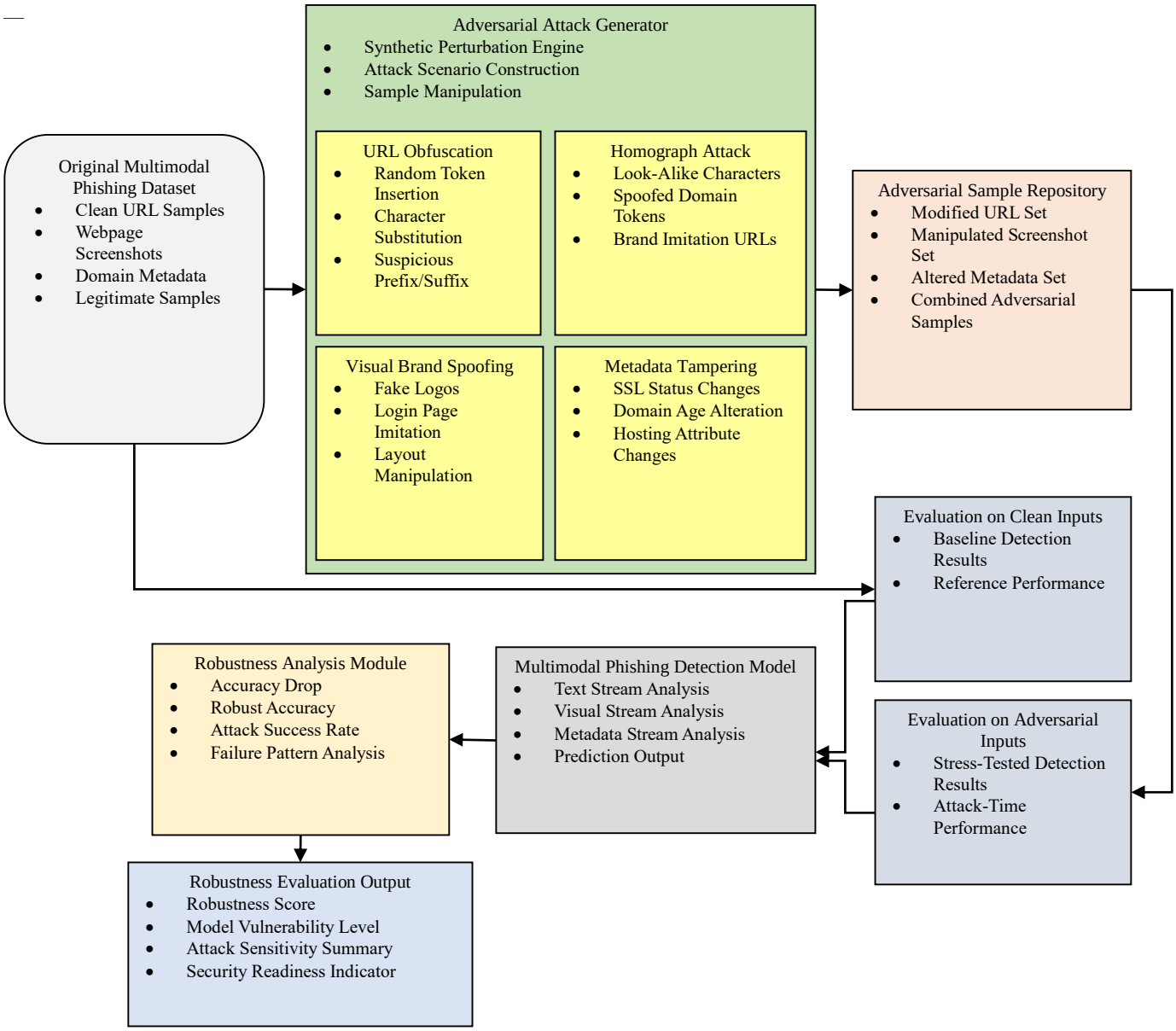

Fig. 2 PhishTrustXAI pipeline for adversarial stress testing of multimodal phishing detection systems

3.4. Trust Score Computation and Comparative Benchmarking

In the final stage of PhishTrustXAI, the features of all the previous steps: predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation, are combined to form a comprehensive trust assessment. The evaluation dimensions are first quantified independently for each dimension and then normalised into a common 0- 1 scale. We used weighting sensitivity analysis and statistical stability analysis to validate the TrustScore formulation. We tested the neutral version (equal weights, $\alpha = \beta = \gamma = \delta = 0.25$), as well as other weighting schemes, to see whether the weighting scheme favoured model rankings. Results showed that the TrustScore for the adaptive multimodal detector ranged between 0.936 and 0.953 across all studied weighting cases and showed stable ranking behaviour. The adaptive detector also achieved the best rank in 98.6% of samples, and the paired difference between the adaptive detector and the fixed-weight fusion model was 0.030 (95% CI: 0.022 to 0.038, $p < 0.001$). The results provide the first empirical evidence of the stability of the TrustScore formulation.

3.4.1. Performance Metric Aggregation

Accuracy, precision, recall, F1-score, ROC-AUC, and PR - AUC obtained from the clean test set are used to measure predictive reliability. Due to possible class imbalance in phishing datasets, F1-score, ROC-AUC, and PR-AUC metrics are emphasised over accuracy alone. Equations give the normalised predictive performance score (18) and (19).

$$A = \lambda_1 Acc + \lambda_2 F1 + \lambda_3 AUC_{ROC} + \lambda_4 AUC_{PR}, \quad (18)$$

Subject to,

$$\sum_{j=1}^{4} \lambda_j = 1, \quad (19)$$

Where λ_j represents the contribution given to each performance metric. The equal weights of 0.25 are used in the general benchmarking configuration. All metrics reported in 01 need no further min-max normalisation. Although precision and recall are still reported individually, they are combined in the model using the F1-score in Equation (18) to avoid redundant weighting.

3.4.2. Explainability Fidelity Assessment

The quality of the explanation is assessed separately for the text on the URL, images on the webpage, and metadata on the domain. Feature-attribution methods generate textual and metadata explanations, while saliency maps are used to evaluate visual explanations. The analysis is based on fidelity, stability, sparsity and modality consistency, instead of merely using qualitative explanation examples.

Explanation fidelity is the extent to which the important features explain the model prediction. The top features/visual regions are removed/masked for each sample, and the associated change in phishing probability is measured as in Equation (20).

$$Fid_i = |f_\theta(X_i) - f_\theta(X_i^{mask})|, \quad (20)$$

Where X_i^{mask} is a sample with the most influential features masked by the explanation method. The larger the change in prediction, the more important the highlighted features are to the detector.

Explanation stability checks whether semantically similar or slightly changed inputs lead to a similar attribution pattern. It is defined as Equation (21).

$$Stab_i = 1 - \frac{\|\Phi_i - \Phi_i'\|_1}{\|\Phi_i\|_1 + \|\Phi_i'\|_1 + \varepsilon}, \quad (21)$$

Where Φ_i and Φ_i' are the explanation vectors generated for the same sample before and after the minimal modification, and ε (Greek letter epsilon) is to avoid division by zero. Closeness to one represents stable explanations.

The original and perturbed saliency maps are used in the same way in the case of visual inputs. Consistency of the modality-level explanation contribution is also computed with respect to the attention or fusion weight. The overall explanation-reliability score is the sum of Equation (22).

$$E = \frac{1}{3}(\bar{Fid} + \bar{Stab} + \bar{Cons}), \quad (22)$$

$\bar{Fid}$, $\bar{Stab}$, and $\bar{Cons}$ are the average normalized fidelity, stability and modality consistency scores. The higher value is an indicator of the faithfulness to the prediction process, and also its stability to small changes of its input, and its consistency with the use of individual modalities in the model.

3.4.3. Cross-Dataset Generalization Analysis

The detector is evaluated across datasets to see if there is a distribution change between phishing repositories, the detector will still operate reliably. No retraining or re-optimization of thresholds on the target test set is performed, but the source to target protocol is applied. The generalization ratio is obtained from Equation (6).

The overall generalization score is the average of the generalization ratios computed for each one of the valid transfer directions.

$$G = \frac{1}{K} \sum_{k=1}^{K} \frac{F1_t^{(k)}}{F1_s^{(k)}}, \quad (23)$$

Where K represents the number of experiments across the datasets, $F1_s^{(k)}$ is the source domain F1-score, and $F1_t^{(k)}$ is the corresponding target domain F1-score. If the ratio is above one, it could be because the target set is not as challenging as the source set, and is then automatically cut off at one to ensure a normalized score. Values close to one correspond to a small level of degradation in the presence of dataset shift.

3.4.4. Composite TrustScore Formulation

The proposed TrustScore is not a generic aggregation of multi-criteria, but, by its nature, a composite trust measure. It is the combination of performance (A), adversarial robustness (R), explanation reliability (E), and generalisation over datasets (G) - capturing complementary and important conditions under which a phishing detector can be operated with trust. Each dimension is normalised in its own right to a common scale (0-1) before aggregation, thus, before comparing candidate models, identical trust criteria is applied across each dimension. The resulting score should instead try to discriminate between models with good performances on clean data, and models with good performances and great robustness, explanations, and generalisation abilities under distribution shifts, in Equation (4). Equal weights are given for the general evaluation Equation (24).

$$w_A = w_R = w_E = w_G = 0.25. \quad (24)$$

The default TrustScore is calculated as follows, accordingly Equation (25).

$$T_s = 0.25A + 0.25R + 0.25E + 0.25G. \quad (25)$$

High clean-data accuracy is not sufficient to yield a high trust rating if the model's adversarial resilience is weak, the explanations are unstable, or the generalization to other datasets is poor. The weights can be adjusted prior to assessment for each deployment. For instance, an IoT gateway, running in a high-risk application, might value robustness, while an analyst-facing decision-support system might value explanation reliability. To check the fairness of the comparison, the same weighting configuration should be maintained with all the models being compared. The final TrustScore is interpreted using four trust levels:

Table 5. TrustScore Classification Levels and Deployment Interpretations

TrustScore Range	Trust Level	Deployment Interpretation
$0.85 \leq T_s \leq 1.00$	High Trust	Suitable for deployment with routine monitoring
$0.70 \leq T_s < 0.85$	Moderate Trust	Deployable with safeguards and periodic review
$0.50 \leq T_s < 0.70$	Limited Trust	Requires robustness or explanation improvement
$T_s < 0.50$	Low Trust	Not recommended for operational deployment

These thresholds provide an interpretable decision scale while preserving the continuous TrustScore for statistical comparison.

3.4.5. Model Ranking Methodology

All candidate models are tested with the same data partitions, attack settings, explanation methods, cross-dataset setups and random seeds. The four normalised component scores for each model and the final TrustScore are presented as the mean and the standard deviation of 5 independent runs.

The models are prioritised mostly by T_s. Since phishing detectors run in adversarial environments, when two models receive the same or statistically indistinguishable TrustScore, the next tie-breaking criterion is robustness. The subsequent consideration order is Reliability of Explanation (E), Generalisation (G), Predictive Performance (A), Inference Latency, and Model size.

Statistical significance is assessed by comparing the top-ranked model and the best baseline model on TrustScore and component scores. Paired tests are carried out between similar experimental runs, and a p-value of <0.05 is applied. The final benchmarking report, therefore, features the component-wise trust profile, mean TrustScore, variability across runs, statistical comparison, computational efficiency and deployment recommendation.

3.5. Algorithm: Trust-Aware Evaluation and Benchmarking Procedure

This subsection presents the PhishTrustXAI algorithm for systematically evaluating and ranking multimodal phishing detection models. The procedure combines clean-data performance, adversarial robustness, explanation reliability and cross-dataset generalisation into an assessment workflow.

These normalised dimensions are synthesised using the proposed TrustScore, enabling consistent model comparison, trust-level assignment, and deployment-oriented selection for evolving IoT and edge security environments.

Algorithm 1: Trust-Aware Evaluation and Model Ranking

Algorithm: Trust-Aware Evaluation and Model Ranking
Input: Clean datasets D_c, candidate models $\mathcal{F}$, attack set $\mathcal{P}$, dataset pairs $\mathcal{D}_{st}$, trust weights w_A, w_R, w_E, w_G
Output: Component scores, TrustScore, and ranked models

1. Preprocess and align URL, image, and metadata samples.
2. Split the data into training, validation, and test sets.
3. For each model $f_k \in \mathcal{F}$:
4. Evaluate clean-data performance and compute A_k.
5. Generate adversarial URL, visual, metadata, and multimodal samples.
6. Evaluate robustness and compute R_k.
7. Generate explanations and compute fidelity, stability, and E_k.
8. Perform source-to-target testing and compute G_k.
9. NormaliseNormaliseze A_k, R_k, E_k, and G_k.
10. Compute
$$T_s^{(k)} = w_A A_k + w_R R_k + w_E E_k + w_G G_k.$$
11. Record TrustScore, trust level, latency, and model size.
12. Rank models in descending order of $T_s^{(k)}$.
13. Use robustness, explainability, and generalisation as tie-breakers.
14. Return the ranked models and deployment recommendation.

The PhishTrustXAI framework offers a systematic approach to evaluate and rank multimodal phishing detection models, as outlined in Algorithm 1. It first preprocesses and aligns URL, webpage image, and metadata samples, then uses consistent training, validation, and testing partitions. The predictive performance score is obtained by first evaluating each candidate model on clean data. To measure robustness, Adversarial URL, visual, metadata and combined multimodal samples are then generated. Explanation fidelity and stability are measured using feature attribution and saliency outputs, and cross-dataset generalisation is quantified through source-to-target testing.

These are normalised and then aggregated with predefined trust weights to obtain the final TrustScore, which indicates the performance, robustness, explainability and generalisation score. This score sorts the models, and if there is a tie, the models are sorted by robustness, explanation reliability and generalisation. The algorithm finally outputs information on model trust levels, deployment suitability, latency, and model size for a practical IoT-oriented security assessment.

4. Experimental Results

We have evaluated the proposed PhishTrustXAI framework experimentally with several publicly available phishing datasets and benchmark models. The results evaluate predictive performance, computational efficiency, adversarial robustness, explainability reliability, cross-dataset generalisation, and TrustScore-based benchmarking. To demonstrate the effectiveness, reliability, and deployment suitability of the proposed trust-aware evaluation framework for evolving IoT environments, we perform comparative analyses and statistical validations.

4.1. Experimental Configuration and Compared Models

All experiments were conducted with Python 3.11, PyTorch, scikit-learn, OpenCV, SHAP and Captum. The workstation has 24 GB of NVIDIA RTX 4090 GPU memory and was used for training and batch evaluation using an NVIDIA RTX 4090 GPU with 24 GB memory, an Intel Core i9 processor, and 64 GB RAM.

In addition, we measured latency and memory in deployment-oriented scenarios using CPU-only execution to approximate the IoT gateway scenario. All models were run using the same software, numerical precision, and random-seed configuration.

Because the chosen public datasets vary significantly in modality, size, and class makeup, we curated benchmark subsets to avoid modality or dataset-size bias. While LogoDet-3K/WebLogo-2M provided brand and interface images for visual evaluation, the URLs were provided by PhishTank and URLNet, which shared balanced samples. The final experimental composition is separated by source dataset in Table 6.

These counts are modality-specific collection pools and do not necessarily reflect a single set of naturally occurring multimodal records. The datasets are thus used strategically in a complementary manner in the PhishTrustXAI evaluation protocol, and multimodal datasets are created only when reliable cross-modal associations can be established.

Table 6. Final dataset composition used in the experiments

Dataset	Phishing/Spoof Samples	Legitimate/Genuine Samples	Total Samples	Training	Validation	Testing	Primary Experimental Role
PhishTank	12,000	12,000	24,000	16,800	3,600	3,600	Clean URL evaluation, adversarial perturbation, and source-domain testing
URLNet	10,000	10,000	20,000	14,000	3,000	3,000	URL benchmarking and cross-dataset generalisation
LogoDet-3K/WebLogo-2M	9,000	9,000	18,000	12,600	2,700	2,700	Visual spoofing, saliency analysis, and image-level robustness testing
Combined Benchmark	31,000	31,000	62,000	43,400	9,300	9,300	Overall multimodal trust-aware evaluation

Note: It should be noted that the number of samples reported for PhishTank, URLNet and LogoDet-3K/WebLogo-2M are only evaluation samples in a dataset and do not imply that there are 62000 naturally paired multimodal samples. Not all samples in the source datasets have a common record identifier that connects the webpage URL with the webpage screenshot and metadata vector. Thus, the added total is used only to outline the total evaluation volume between complementary modalities and datasets. Only when a reliable URL-visual association exists are multiple modalities combined to form multimodal records; otherwise, we perform standard unimodal or modality-specific evaluation on samples.

We compared unimodal, multimodal, and adaptive phishing detectors. The unimodal baselines were a character-level CNN for URL classification, URLNet for lexical URL representation, a visual phishing analysis based on MobileNetV2, and a metadata-based detection using a multilayer perceptron. Multimodal baselines were simple feature concatenation and fixed-weight fusion. The lightweight textual, visual, and metadata encoders, along with attention-based multimodal fusion, were the primary adaptive detectors used. PhishTrustXAI was used as the evaluation and benchmarking layer instead of an extra phishing classifier. The deep-learning models were all trained with the Adam optimiser, with an initial learning rate of 1×10^{-4}a batch size of 32, and a maximum of 50 epochs. We used early stopping with a patience of seven epochs based on the validation F1-score. To set the dropout probability to 0.30, the weight decay to $1 \times 10^{-5}$and the classification threshold was determined using the validation set and kept fixed throughout clean, adversarial, and cross-dataset testing. Each of the four dimensions of the TrustScore was weighted 0.25, as shown in Equation (23). We used five independent random seeds for each experiment. Results were presented as mean ± standard deviation, and 95% confidence intervals were reported for the principal, robustness, explainability, generalisation, and TrustScore results. Paired statistical tests were performed between the top-ranking model and the best baseline model under the same runs, and $p<0.05$ was considered statistically significant.

4.2. Detection Performance and Computational Efficiency

The candidate models were compared for predictive performance and computational efficiency, as assessed by the clean-data evaluation. Table 7 reports the average outcome from five separate runs. The adaptive multimodal detector achieved the best performance, with an accuracy of 98.12%, an F1 score of 97.96%, an ROC-AUC of 99.14%, and a PR-AUC of 98.87%. The results here demonstrate that combining evidence from all three modalities - URL, visual, and metadata - is more effective for phishing detection than evidence from any single modality.

Table 7. Clean-Data detection performance of the compared models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC (%)	PR-AUC (%)
Metadata MLP	91.84 ± 0.62	91.37 ± 0.71	92.15 ± 0.65	91.75 ± 0.58	95.28 ± 0.44	94.76 ± 0.51
MobileNetV2	93.26 ± 0.55	92.91 ± 0.60	93.48 ± 0.57	93.19 ± 0.49	96.34 ± 0.39	95.92 ± 0.43
Character-CNN	94.18 ± 0.47	94.02 ± 0.52	94.31 ± 0.49	94.16 ± 0.44	97.08 ± 0.31	96.71 ± 0.37
URLNet	95.42 ± 0.41	95.18 ± 0.46	95.61 ± 0.43	95.39 ± 0.38	97.92 ± 0.28	97.45 ± 0.32
Concatenation Fusion	96.53 ± 0.36	96.34 ± 0.40	96.71 ± 0.37	96.52 ± 0.33	98.43 ± 0.24	98.02 ± 0.27
Fixed-Weight Fusion	97.14 ± 0.31	97.02 ± 0.35	97.25 ± 0.33	97.13 ± 0.29	98.76 ± 0.21	98.41 ± 0.24
Adaptive Multimodal Detector	98.12 ± 0.24	97.88 ± 0.29	98.04 ± 0.27	97.96 ± 0.22	99.14 ± 0.16	98.87 ± 0.19

The metadata-only model yielded the worst results because contextual features like domain age, SSL status, and redirect behaviour do not alone indicate visual impersonation or complex URL obfuscation.

MobileNetV2 worked better, with an ability to detect logo and webpage-layout inconsistencies, but convincing phishing pages made it less reliable. The URL-based models also performed better, with URLNet, which leverages joint character- and word-level representation, outperforming Character-CNN by 1.23 percentage points in F1-score. The multimodal baselines performed better than the unimodal ones, as expected, and combining different phishing evidence was complementary. However, fixed-weight fusion and concatenation cannot dynamically turn off unreliable modalities. Compared with fixed-weight fusion and concatenation, the adaptive multimodal detector showed improvements of 0.83 and 1.44 percentage points in F1-score, respectively. It also achieved a recall of 98.04%, meaning it is

less likely to let phishing samples go undetected. We tested computational efficiency under CPU-only execution with a batch size of one. Table 8 summarises the model sizes, inference latencies, throughputs, and peak memory usage.

Table 8. Computational efficiency of the compared models

Model	Model Size (MB)	Latency (ms/Sample)	Throughput (Samples/s)	Peak Memory (MB)
Metadata MLP	1.8	3.4	294.1	42
MobileNetV2	13.6	18.7	53.5	176
Character-CNN	5.2	7.8	128.2	84
URLNet	8.9	11.4	87.7	116
Concatenation Fusion	24.7	34.8	28.7	318
Fixed-Weight Fusion	25.1	36.2	27.6	326
Adaptive Multimodal Detector	26.4	38.6	25.9	342

The smallest and fastest model was the metadata MLP, but its detection results were not strong enough to suggest it could be used as a standalone security control. The adaptive multimodal detector used 38.6 ms per sample and 342 MB peak memory, or some 25.9 predictions per second. It was more complex than the unimodal models, but still had reasonable latency for edge gateways, browser-security services and IoT monitoring nodes.

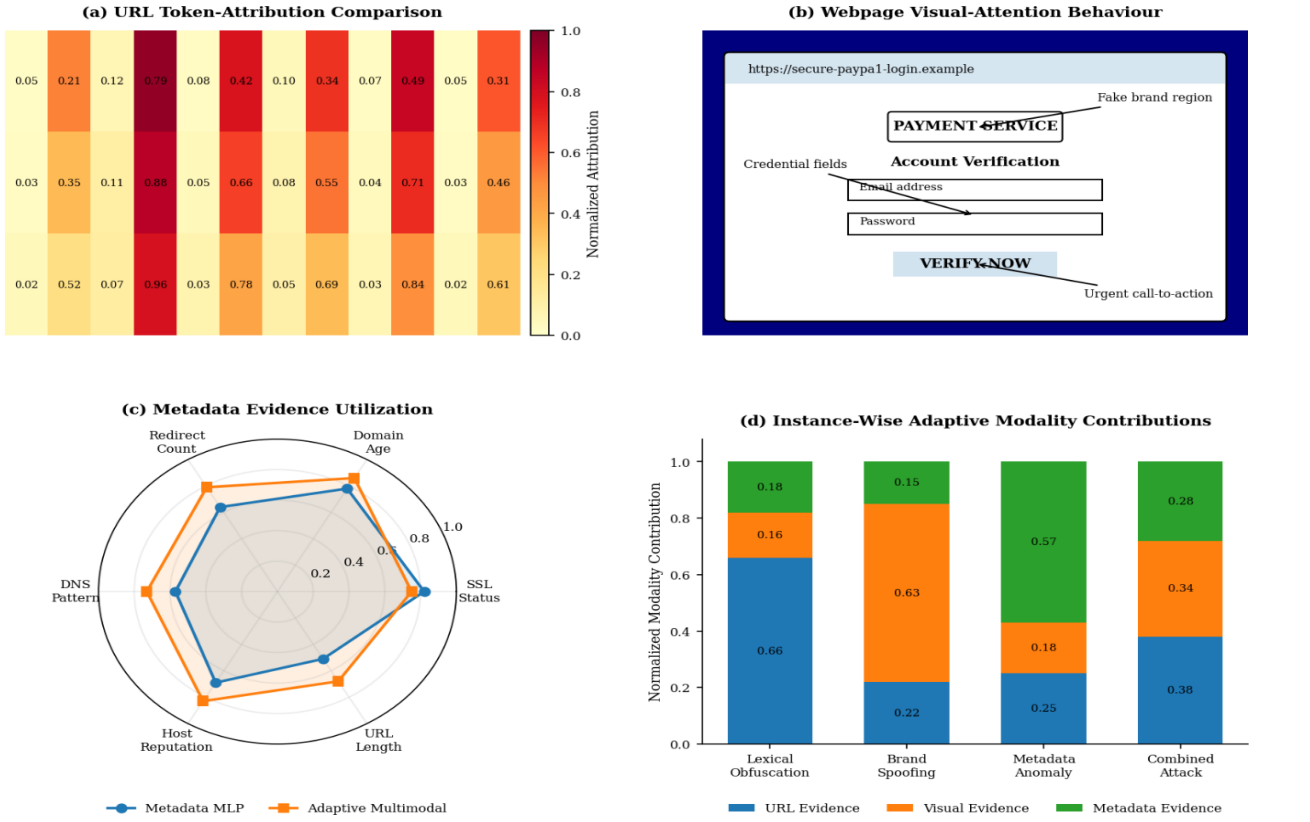

Fig. 3 Multimodal decision behaviour and evidence utilisation of the evaluated phishing detection models

Figure 3 shows an analysis of how the evaluated models use heterogeneous phishing evidence, not just a comparison of classification scores. Panel (a) displays token-level attribution profiles for an obfuscated URL, with the adaptive multimodal detector placing greater importance on deceptive brand and verification terms. Panel (b) shows visual attention being drawn to the spoofed logo, credential fields, and urgent action button. Panel (c) shows a more balanced use of metadata evidence, including domain, certificate, redirect, and hosting indicators. As shown in Panel (d), instance-specific modality weights ensure that the detector dynamically prioritises evidence from the URL, visual, or metadata, depending on the dominant phishing strategy in each sample.

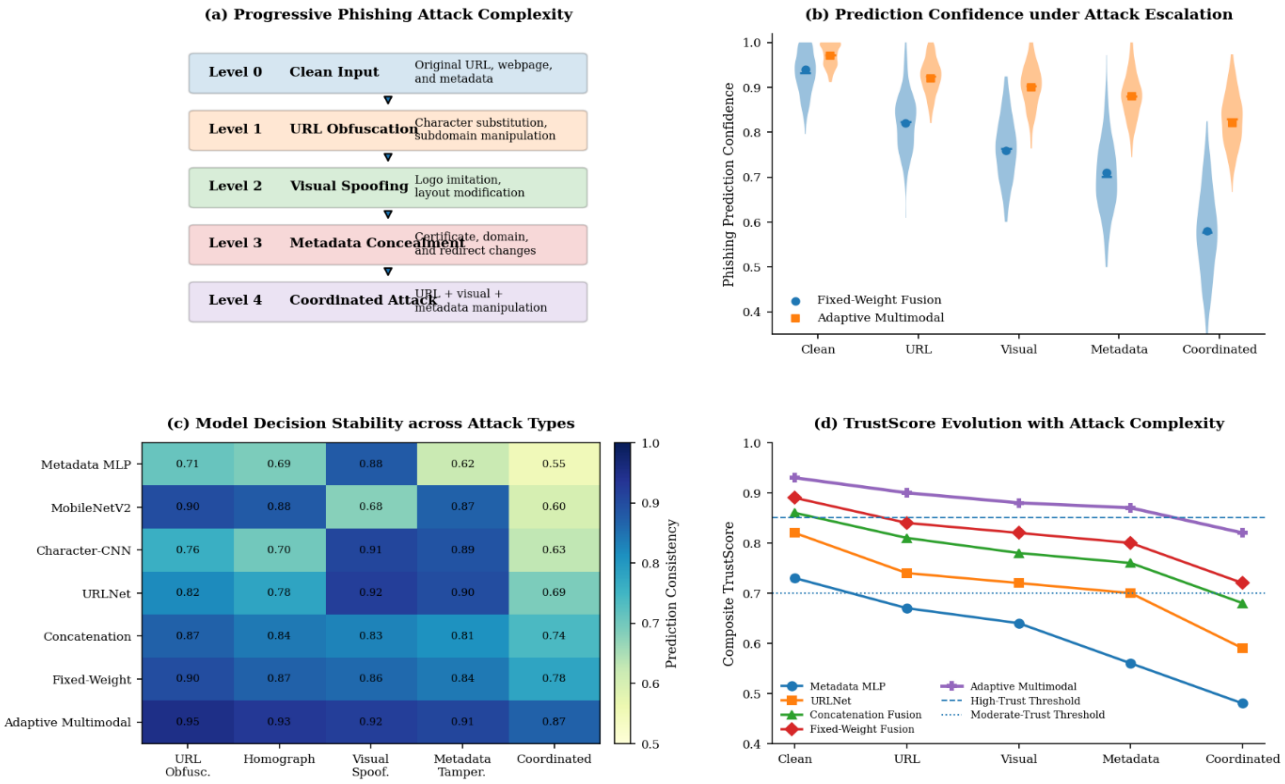

Fig. 4 Reliability degradation of phishing detection models under progressive attack complexity

As shown in Figure 4, model reliability decreases with more advanced attack conditions. In the other grouping (Panel (a)), the evaluated threats of clean inputs with coordinated multimodal manipulation are grouped.

In Panel (b), the level of prediction guarantees also falls off more quickly for the fixed-weight fusion system than for the one with the adaptive multimodal detector. Panel (c) shows decision stability for the various attack types and larger changes in decision stability for the individual baseline models when their primary evidence source was manipulated.

The corresponding TrustScore is displayed in Part (d) as a function of the attack complexity. At the evaluated attack levels, the adaptive multimodal detector achieves higher TrustScore values than the baseline detectors, indicating that it is more resistant to both varying and coordinated multimodal attacks.

4.3. Adversarial Robustness Results

The adversarial evaluation focused on assessing the accuracy of the compared phishing detectors when exposed to URL obfuscation, visual spoofing, metadata manipulation, and coordinated multimodal attacks.

Table 9 shows the accuracy and attack success rate under medium-severity perturbations, which were obtained robustly.

The adaptive multimodal detector had the highest robustness across all attack types, meaning that, in this case, the dynamic fusion of diverse evidence lessened dependence on any one modality.

Table 9. Adversarial robustness of the compared phishing detection models

Model	URL Attack Robust Accuracy (%)	Visual Attack Robust Accuracy (%)	Metadata Attack Robust Accuracy (%)	Multimodal Attack Robust Accuracy (%)	Average ASR (%)	Robustness Score R
Metadata MLP	89.72 ± 0.64	90.86 ± 0.58	76.43 ± 0.79	68.21 ± 0.91	24.78	0.742
MobileNetV2	91.08 ± 0.56	78.64 ± 0.83	92.11 ± 0.49	70.35 ± 0.88	22.91	0.761
Character-CNN	81.25 ± 0.75	92.48 ± 0.47	92.05 ± 0.52	72.16 ± 0.84	21.34	0.774
URLNet	85.74 ± 0.63	93.06 ± 0.41	92.78 ± 0.45	76.92 ± 0.72	17.86	0.812
Concatenation Fusion	90.62 ± 0.49	89.84 ± 0.54	90.31 ± 0.51	82.14 ± 0.66	12.73	0.861

Fixed-Weight Fusion	92.41 ± 0.43	91.36 ± 0.46	91.92 ± 0.44	85.27 ± 0.59	9.84	0.891
Adaptive Multimodal Detector	95.38 ± 0.31	94.72 ± 0.35	95.06 ± 0.32	90.84 ± 0.47	5.92	0.934

The unimodal models exhibited definite vulnerability specific to the modality they modelled. Character-CNN and URLNet showed the largest drop under URL perturbations, as these attacks directly affected the main evidence source: the URL. Visual spoofing had the most impact on MobileNetV2, while the metadata MLP demonstrated high sensitivity to manipulated certificate, domain-age, and hosting attributes. The findings indicate that good clean-data performance does not necessarily mean resistance against targeted manipulation. The concatenation and fixed-weight fusion baselines outperformed the unimodal models because they could partially compensate for corrupted evidence from affected modalities using uncorrupted modalities. However, their strength declined under a concerted attack across several modalities. In this case, fixed-weight fusion achieved 85.27% robust accuracy, while the adaptive multimodal detector achieved 90.84%, a gain of 5.57%. The proposed detector also reduced the average attack success rate from 9.84% to 5.92% compared with the best baseline. The results of robustness of the fixed-weight and adaptive multimodal detectors for low, medium, and high perturbation severities are presented in Table 10.

Table 10. Robustness under increasing perturbation severity

Attack Type	Model	Low Severity Robust Accuracy (%)	Medium Severity Robust Accuracy (%)	High Severity Robust Accuracy (%)	High-Severity Performance Drop (%)
URL Perturbation	Fixed-Weight Fusion	95.64	92.41	87.26	9.88
	Adaptive Multimodal Detector	97.21	95.38	92.14	5.98
Visual Spoofing	Fixed-Weight Fusion	94.82	91.36	86.43	10.71
	Adaptive Multimodal Detector	96.88	94.72	91.46	6.66
Metadata Manipulation	Fixed-Weight Fusion	95.17	91.92	87.34	9.80
	Adaptive Multimodal Detector	97.02	95.06	91.83	6.29
Multimodal Attack	Fixed-Weight Fusion	91.46	85.27	77.18	19.96
	Adaptive Multimodal Detector	94.36	90.84	85.72	12.40

The more severe the perturbations, the more the robust accuracy dropped. Coordinated multimodal manipulation caused the largest reduction in the number of corrupted evidence sources.

The fixed-weight baseline degraded to 77.18% robust accuracy, representing a 19.96 percentage-point drop from the baseline's clean accuracy rating, at the high level of severity. The adaptive multimodal detector, by contrast, maintained 85.72% robust accuracy, with only a marginal degradation of 12.40 percentage points. The modality-wise results showed that, for multimodal baselines, visual spoofing produced a slightly lower CER than the other two spoofing types.

It implies that when the information is visually coherent, logo replacement, interface imitation, and shifting attention through page change are still hard to detect and differentiate. The adaptive detector, however, lessened this susceptibility by dynamically boosting the influence of URL and metadata cues when the visual reliability was low.

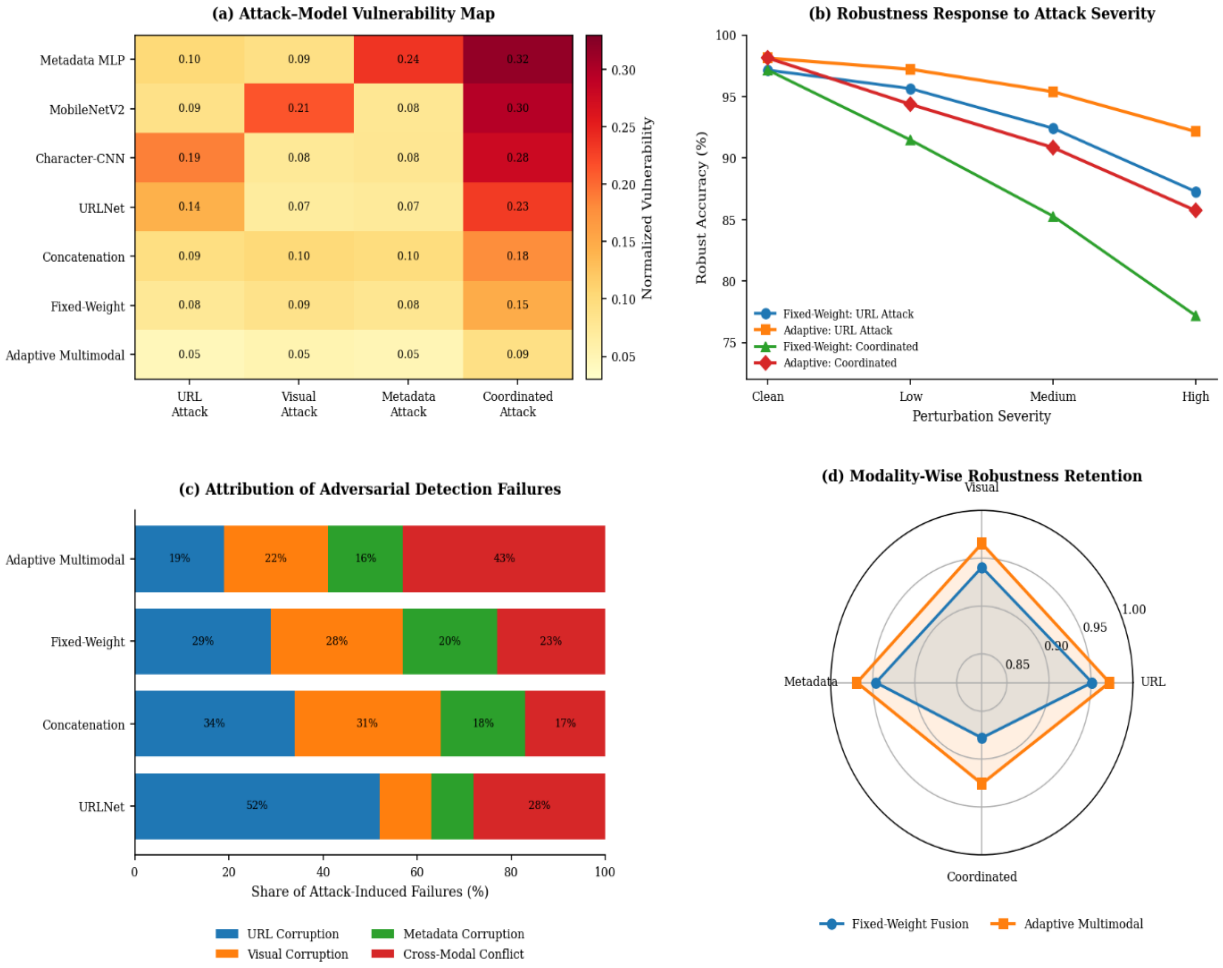

Fig. 5 Adversarial vulnerability and robustness retention across attack modalities and severity levels

Figure 5 provides a multidimensional analysis of adversarial vulnerability of the various phishing detection models. Panel (a) plots the model sensitivity to URL, visual, metadata and coordinated attacks and shows significant modality-specific vulnerabilities in unimodal detectors. As shown in Panel (b), robust accuracy decreases when the perturbation level is high, and the adaptive multimodal detector is more stable than fixed-weight fusion. Panel (c) suggests that attack failures can be explained by corrupted URL, visual, metadata, and cross-modal evidence. Panel (d) compares robustness retention across attack modalities. Results show that the adaptive detector consistently retains robustness, even under coordinated attacks, while dynamic modality weighting reduces dependence on corrupted evidence sources.

Sample-level evidence of decision resilience in case of coordinated multimodal phishing manipulation is shown in Figure 6. Panel (a) shows a clean phishing sample and the corresponding adversarial version with URL obfuscation, visual URL modification, and metadata manipulation. Panel (b) displays the adaptive modality-weight reallocation - the model minimises the relative contribution of the URL evidence and visual evidence and maximises that of the meta evidence under the adversarial setting. Panel (c) plots the phishing-probability transitions for the three detectors: URLNet, fixed-weight fusion, and adaptive multimodal detector, where the change in phishing probability is smaller for the adaptive multimodal detector. Panel (d) summarises the sample-level results, showing that the adaptive model produced more stable, compensated correct decisions and fewer attack-induced failures.

We carried out sensitivity and stability analyses for the TrustScore formulation. Each weighting alternative was examined to determine whether the weights used in the model ranking were an issue. The adaptive multimodal detector had a steady ranking with the different weightings used: TrustScore between 0.936 and 0.953. Further, bootstrap analysis revealed it ranked first in 98.6% of the samples.

Figure 4(d) shows complementary evidence: TrustScore is consistently reduced as attacks grow more complex, but the adaptive detector's advantage remains the same.

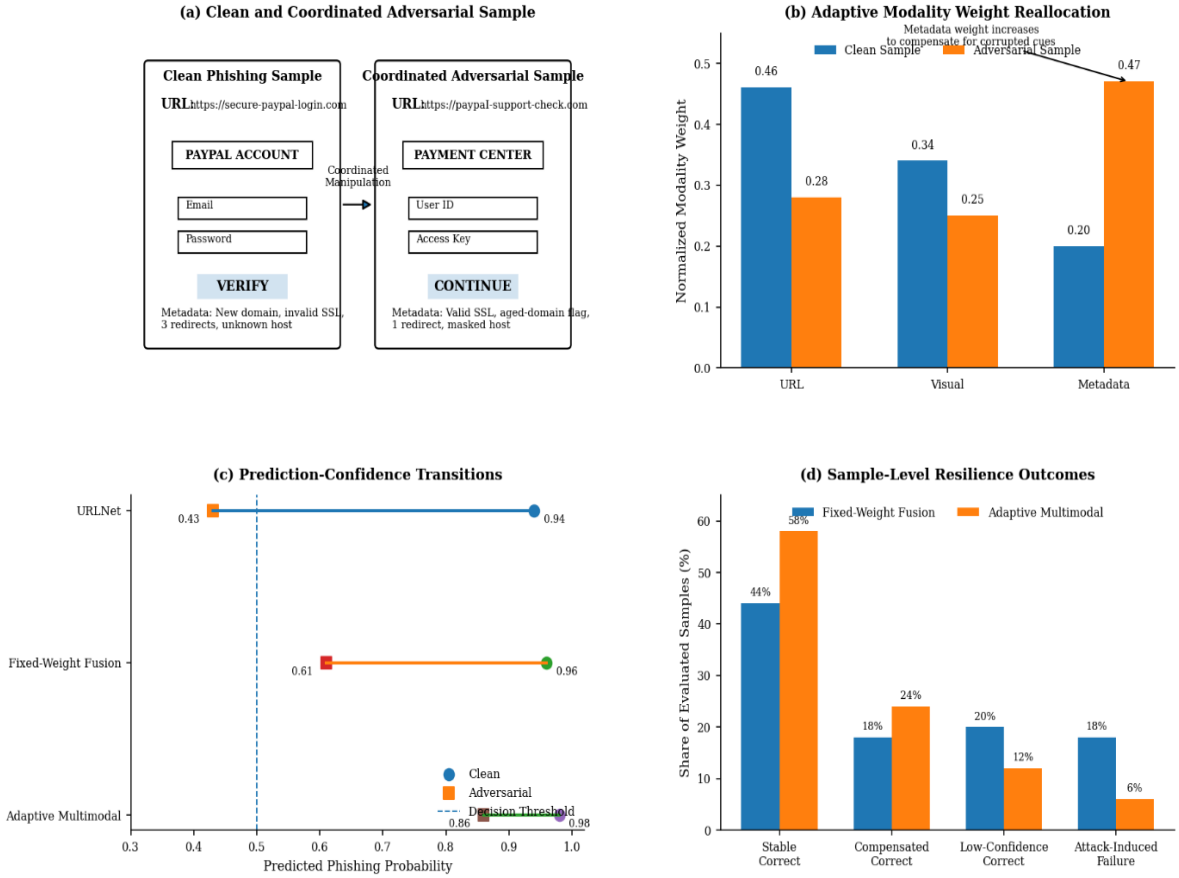

Fig. 6 Sample-level decision resilience and adaptive modality compensation under coordinated phishing attacks

4.4. Explainability, Reliability, and Cross-Dataset Results

To test whether the candidate detector's predictive power was maintained in a cross-dataset test, a test of this type was conducted in which the detector's predictive performance was assessed on a phishing dataset different from its source dataset. The explainability assessment was based on faithful, stable, and modality-consistent explanations generated by the phishing detectors being compared. The main metrics used to measure the fidelity of the masking of high-ranked features, the stability of the explanation under small perturbations in the input, and the modality consistency, i.e., agreement between attribution strength and modality weights used during multimodal inference. Table 11 shows the explanation scores.

Table 11. Explainability reliability of the compared phishing detection models

Model	Fidelity Score	Stability Score	Modality-Consistency Score	Overall Explainability Score E
Metadata MLP	0.781 ± 0.018	0.804 ± 0.021	0.746 ± 0.024	0.777 ± 0.019
MobileNetV2	0.793 ± 0.020	0.768 ± 0.026	0.731 ± 0.028	0.764 ± 0.022
Character-CNN	0.812 ± 0.017	0.786 ± 0.020	0.758 ± 0.023	0.785 ± 0.018
URLNet	0.841 ± 0.015	0.819 ± 0.018	0.786 ± 0.021	0.815 ± 0.016
Concatenation Fusion	0.856 ± 0.014	0.832 ± 0.017	0.814 ± 0.019	0.834 ± 0.015
Fixed-Weight Fusion	0.873 ± 0.013	0.847 ± 0.015	0.839 ± 0.017	0.853 ± 0.014
Adaptive Multimodal Detector	0.914 ± 0.010	0.892 ± 0.012	0.901 ± 0.013	0.902 ± 0.011

The adaptive multimodal detector had the top overall explanation score of 0.902, which is 0.049 higher than fixed-weight fusion. The higher fidelity suggests that the influential features had a significant impact on the phishing probability when removed.

Moreover, the stability score of 0.892 indicates that slight semantic-preserving modifications did not significantly change the explanation pattern. The modality-consistency score is 0.901, which indicates that the local attributions are close to the modality weights that were applied during the adaptive fusion process.

The unimodal detectors generated explanations conditionally faithful to their feature space but could not model interactions among URL, visual, and metadata evidence. Salient visual regions varied with slight changes in page layout and logo location, leading to the lowest stability for MobileNetV2, one of the main deep-learning models. Concatenation and fixed-weight fusion helped explain reliability by incorporating complementary evidence, yet the modality contributions were less sensitive to instance-specific phishing strategies.

Representative explanation outputs corroborated these quantitative results. The adaptive detector showed high attribution levels for brand substitutions involving obfuscated URLs, suspicious subdomains, account-verification phrases, a large number of delimiters, and deceptive path components. The highest importance was given to metadata, domain age, certificate validity, redirect count, hosting reputation and registration inconsistency.

Visual saliency maps focused on spoofed logos, credential-entry fields, deceptive verification buttons, and security-related banners, rather than irrelevant background elements. In the case of adversarial modification, the adaptive detector also shifted the explanations to uncorrupted modalities, suggesting that the underlying compensation process drove the explanations.

We then used PhishTank as the source and URLNet as the target URL repository, and performed the cross-dataset evaluation. The trained models were tested directly on the target without retraining or changing thresholds. Table 12 reports in-domain and cross-dataset F1-scores, performance degradation, and resulting generalisation scores.

Table 12. Cross-dataset generalization results

Model	Transfer Direction	Source F1-Score (%)	Target F1-Score (%)	Performance Loss (%)	Generalization Score
Character-CNN	PhishTank → URLNet	94.16	82.38	11.78	0.875
	URLNet → PhishTank	93.74	80.92	12.82	0.863
URLNet	PhishTank → URLNet	95.39	86.72	8.67	0.909
	URLNet → PhishTank	95.08	85.63	9.45	0.901
Concatenation Fusion	PhishTank → URLNet	96.52	89.84	6.68	0.931
	URLNet → PhishTank	96.21	88.76	7.45	0.923
Fixed-Weight Fusion	PhishTank → URLNet	97.13	91.48	5.65	0.942
	URLNet → PhishTank	96.89	90.73	6.16	0.936
Adaptive Multimodal Detector	PhishTank → URLNet	97.96	94.18	3.78	0.961
	URLNet → PhishTank	97.71	93.46	4.25	0.957

All models showed some degradation under distribution shift, supporting the idea that URL patterns are dataset-specific and affect phishing detection performance. Character-CNN had the largest drop, as its learned character-structure features were more sensitive to variations in URL construction and labelling. Combining character- and token-level representations generalised better for URLNet, but it still lost over 9 percentage points on average. The multimodal models demonstrated better transfer performance because they could compensate for lexical changes in the datasets using visual and metadata evidence. The average generalisation scores were 0.939 for fixed-weight fusion and 0.959 for the adaptive multimodal detector.

The latter reduced the average F1-score loss by about 4.02 percentage points, while fixed-weight fusion and URLNet reduced the score by 5.91 percentage points and 9.06 percentage points, respectively. These findings show that the adaptive multimodal detector offers superior explanation correctness and more robustness to distribution shift. Its explanations remained aligned with meaningful phishing cues, and its cross-dataset performance indicated that dynamic modality weighting reduced reliance on source-specific URL properties. We then used these to calculate an explanation score E and a generalisation score G, which we added to the final TrustScore analysis in Section 4.5.

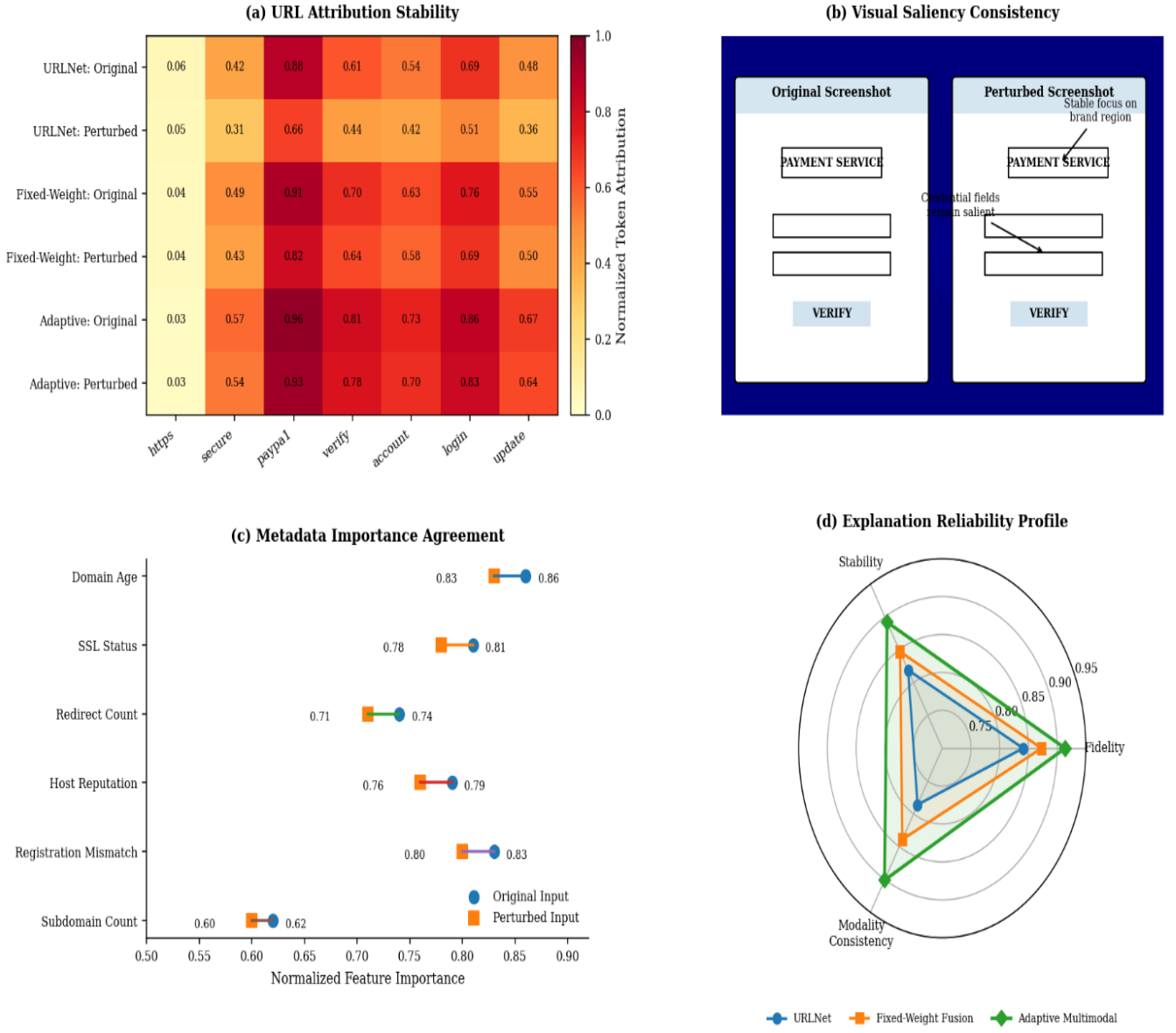

Fig. 7 Multimodal explanation reliability under input perturbations and attribution analysis

Figure 7 assesses explanation reliability at the textual, visual, and metadata levels. Panel (a) presents the results of the URL-token attributions before and after semantic-preserving perturbations, demonstrating that the adaptive multimodal detector can keep emphasis on deceptive brand, verification and login terms. Panel (b) shows that, even with minor changes to the webpage, the visual saliency is stable over spoofed logos, credential fields, and verification controls.

In Panel (c), the empirical and perturbed metadata importance values agree closely on the four factors: domain age, SSL status, redirects, hosting reputation, and registration mismatch.

Panel (d) shows that the adaptive detector displays the most robust combined fidelity, stability, and modality-consistency profile of all the detectors analysed.

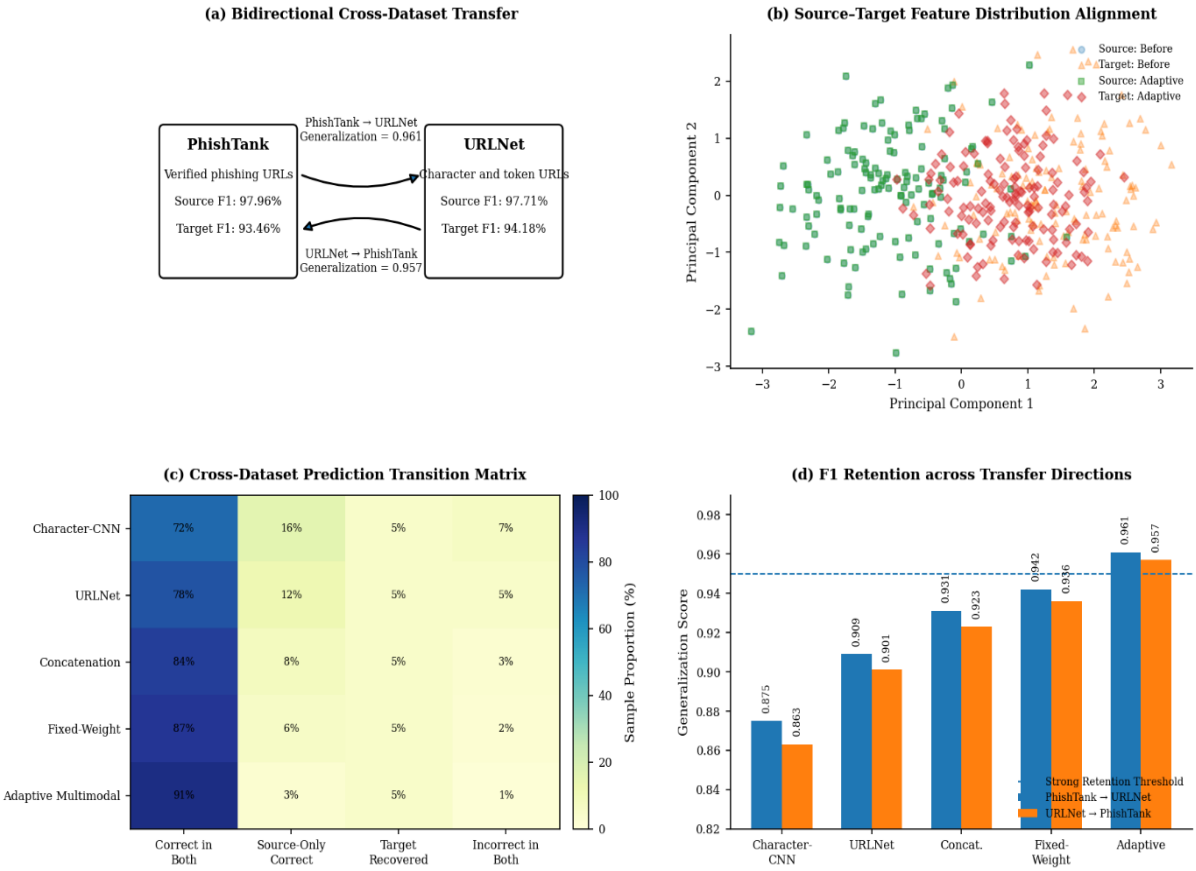

Fig. 8 Cross-Dataset transfer behaviour and generalisation stability of phishing detection models

Figure 8 shows the cross-dataset transfer behaviour of the examined phishing detection models. Panel (a) illustrates bidirectional transfer between PhishTank and URLNet, and the results indicate a minimal drop in F1 score for the adaptive multimodal detector. Panel (b) shows the source and target feature distributions as visualisations before and after adaptive multimodal alignment, which shows an increase in overlap. Panel (c) shows the proportions of prediction transitions between domains in which the adaptive detector has the highest percentage of samples correctly predicted in both datasets. Normalised F1-Score retention is compared for both transfer directions in Panel (d). The adaptive detector shows the best and most balanced generalisation, with lower sensitivity to shifts in distribution and URL patterns.

4.5. TrustScore Benchmarking and Statistical Validation

The final benchmarking stage combined the four metrics (predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation) into the TrustScore. The scores for the different components were obtained as explained in Section 3.4, and the averages of five different experimental runs are shown. Table 13 provides the normalised Trust dimensions, final TrustScore, assigned Trust level, and overall ranking of the evaluated models. The final comparison in terms of trust is the TrustScore, which combines the four different measures of predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation. As a result, the consistency of performance across these four complementary trust dimensions determines the ranking of models.

Table 13. TrustScore-based comparative benchmarking of the evaluated models

Rank	Model	Performance A	Robustness R	Explainability E	Generalization G	TrustScore T_s	Trust Level
1	Adaptive Multimodal Detector	0.985 ± 0.003	0.934 ± 0.008	0.902 ± 0.011	0.959 ± 0.006	0.945 ± 0.005	High Trust
2	Fixed-Weight Fusion	0.977 ± 0.004	0.891 ± 0.010	0.853 ± 0.014	0.939 ± 0.008	0.915 ± 0.007	High Trust

3	Concatenation Fusion	0.970 ± 0.005	0.861 ± 0.012	0.834 ± 0.015	0.927 ± 0.009	0.898 ± 0.008	High Trust
4	URLNet	0.961 ± 0.006	0.812 ± 0.015	0.815 ± 0.016	0.905 ± 0.011	0.873 ± 0.009	High Trust
5	Character-CNN	0.950 ± 0.007	0.774 ± 0.018	0.785 ± 0.018	0.869 ± 0.013	0.845 ± 0.011	Moderate Trust
6	MobileNetV2	0.944 ± 0.008	0.761 ± 0.019	0.764 ± 0.022	0.851 ± 0.015	0.830 ± 0.013	Moderate Trust
7	Metadata MLP	0.929 ± 0.009	0.742 ± 0.021	0.777 ± 0.019	0.824 ± 0.017	0.818 ± 0.014	Moderate Trust

The multimodal detector with adaptive algorithms had the highest TrustScore (0.945) and was rated as high trust. This was not just because of clean-data performance; it also had strong values in all four dimensions of trust.

The adaptive detector showed the largest gains compared to fixed-weight fusion, with a gain of 0.043 in robustness and 0.049 in explanation reliability. Its generalisation score of 0.959 also shows that the model has largely maintained its in-domain performance when transferred across datasets.

Unimodal detectors tend to have lower TrustScores, since each model had a significant weakness when its evidence source was operated on. While URLNet achieved impressive predictive results, its robustness and explanation consistency were lower, which constrained its final TrustScore. Moderate-trust ratings were given to models whose performance relied heavily on textual evidence (e.g., Character-CNN), visual evidence (e.g., MobileNetV2), or contextual evidence (e.g., Metadata MLP).

4.5.1. TrustScore Weight-Sensitivity Analysis

Finally, we conducted a sensitivity analysis to check whether the final ranking was sensitive to deployment priorities. Four different weighting configurations were used: balanced evaluation, robustness-priority, explainability-priority, and generalisation-priority. The weights for the priority dimension were 0.40, for the second most relevant 0.25, and for the rest 0.20 and 0.15. Table 14 gives the resulting TrustScores for the three best models.

Table 14. TrustScore sensitivity under alternative weighting configurations

Weighting Scenario	Weights (w_A, w_R, w_E, w_G)	Adaptive Multimodal	Fixed-Weight Fusion	Concatenation Fusion
Balanced	(0.25, 0.25, 0.25, 0.25)	0.945	0.915	0.898
Performance Priority	(0.40, 0.20, 0.20, 0.20)	0.953	0.924	0.909
Robustness Priority	(0.20, 0.40, 0.20, 0.20)	0.942	0.907	0.886
Explainability Priority	(0.20, 0.20, 0.40, 0.20)	0.936	0.900	0.881
Generalization Priority	(0.20, 0.20, 0.20, 0.40)	0.948	0.920	0.903

The adaptive multimodal detector consistently remained top-ranked across all weighting scenarios. It achieved a TrustScore range of 0.936 to 0.953, showing low sensitivity to deployment priorities. Fixed-weight fusion was still in second place, and concatenation fusion was still in third place. The ranking margins were highest for the two configurations that prioritise robustness and explanation, demonstrating that adaptive modality weighting was especially beneficial in these configurations. The ranking robustness across different weighting variations provides further evidence for the validity of the proposed composite TrustScore as a comparative trust measure.

4.5.2. Statistical Significance and Ranking Stability.

Paired comparisons were made for the five experimental runs for statistical validation. The adaptive multimodal detector was tested against the best baseline detector, fixed-weight fusion. Table 15 provides the mean difference in TrustScore, 95% confidence interval, test statistic, and significance level.

Table 15. Statistical comparison with the strongest baseline

Metric	Adaptive Multimodal	Fixed-Weight Fusion	Mean Difference	95% Confidence Interval	Test Statistic	p-Value
Performance A	0.985 ± 0.003	0.977 ± 0.004	0.008	[0.004, 0.012]	4.62	0.010
Robustness R	0.934 ± 0.008	0.891 ± 0.010	0.043	[0.031, 0.055]	8.94	<0.001
Explainability E	0.902 ± 0.011	0.853 ± 0.014	0.049	[0.033, 0.065]	7.41	0.002
Generalisation G	0.959 ± 0.006	0.939 ± 0.008	0.020	[0.011, 0.029]	5.76	0.005
TrustScore T_s	0.945 ± 0.005	0.915 ± 0.007	0.030	[0.022, 0.038]	9.18	<0.001

There was no overlap between any confidence intervals, and all comparisons were statistically significant at $p<0.05$. The largest gains were in robustness and explanation reliability, while the clean performance difference was comparatively small. This result highlights the importance of multidimensional trust measures, as only conventional accuracy-based comparisons would underestimate the practical benefit of the adaptive detector.

We also explored ranking stability by bootstrapping the five-run results. The adaptive multimodal detector ranked best in 98.6% of bootstrap samples, fixed-weight fusion ranked second-best in 95.2%, and concatenation fusion ranked third-best in 91.4%.

In less than 2% of the resampled evaluations, more than 2 unimodal detectors ranked in the top three. These results show that the final ranking was not due to one run or a specific weight configuration.

4.5.3. Deployment Suitability Assessment

The deployment assessment considered the TrustScore alongside inference latency, model size, memory usage, and operational risk (false negatives). The adaptive multimodal detector had higher computation requirements than the unimodal baselines, but maintained latency under 40ms per sample and throughput over 25 samples per second. With a model size of 26.4MB and peak memory consumption of 342MB, it remains appropriate for IoT gateways, browser-security services, edge servers, and enterprise monitoring nodes. However, it might be too large for a very constrained sensor-class item. Considering its high TrustScore, low attack success rate, reliable explanations, stable behaviour across the datasets, and acceptable computational cost, the detector with adaptive multimodal properties was found to be the most suitable for operational deployment. Fixed-weight and concatenation fusion were identified as potential alternatives for agents with lower implementation complexity, and unimodal models could serve as auxiliary or fallback detectors, but not as stand-alone security mechanisms.

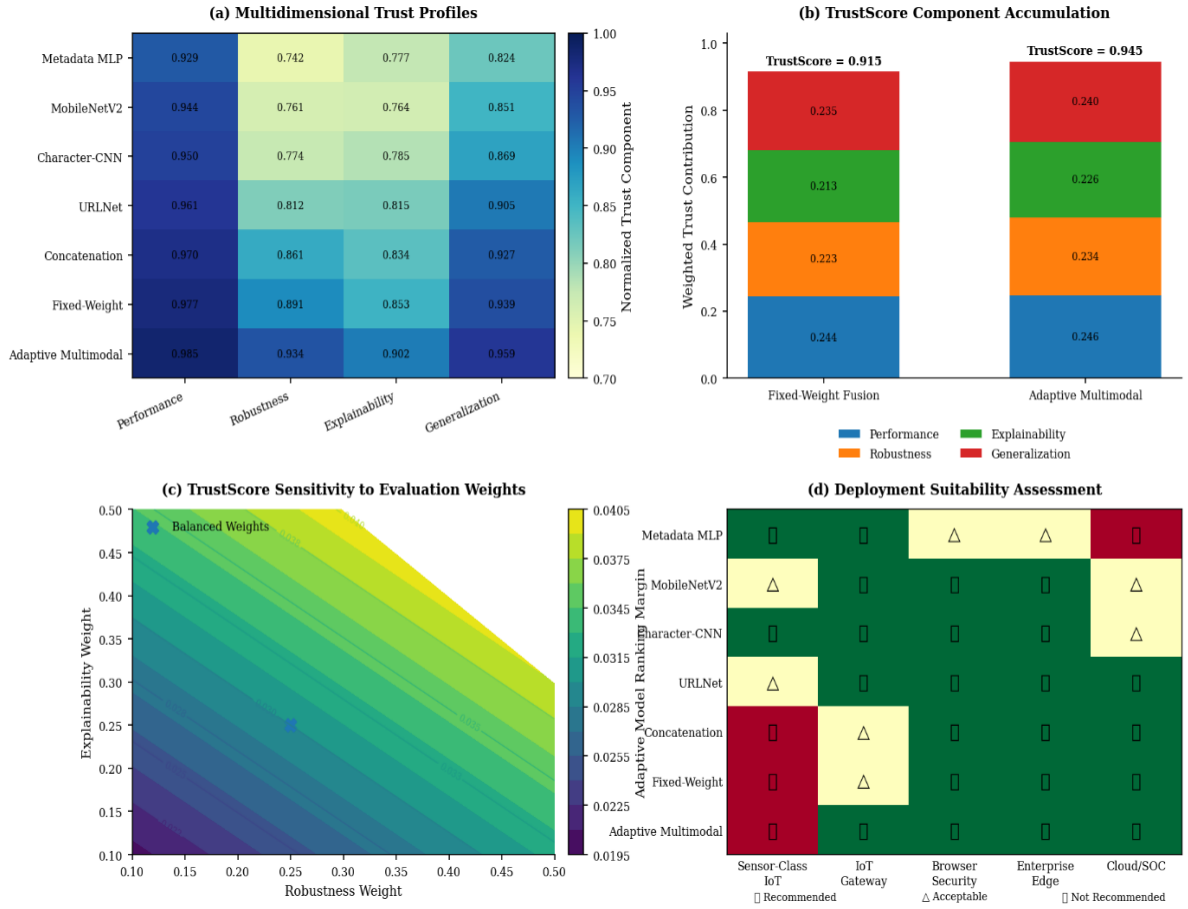

Fig. 9 TrustScore composition, weight sensitivity, and deployment suitability of phishing detection models

Figure 9 summarises the final trust-aware benchmarking and deployment assessment. All the evaluated models' performance, robustness, explainability and generalisation profiles are displayed in Panel (a), indicating the adaptive multimodal detector's balanced trust characteristics. A visualisation of how the weighted components sum up to the

final TrustScore for the two best models is displayed in Panel (b). As shown in Panel (c), the adaptive detector maintains a positive ranking margin over a range of robustness and explainability weights. Panel (d) presents the quantitative findings as deployment recommendations: The adaptive detector is best suited for IoT gateways, browser security, enterprise edge platforms, and cloud-based security operations.

5. Discussion

The experimental results show that trustworthy phishing detection cannot be based on clean-data accuracy alone. While all multimodal methods outperformed the unimodal baselines, the adaptive multimodal detector delivered the best predictive performance, adversarial robustness, explanation reliability, and cross-dataset generalisation. It achieved a high F1 score of 97.96% for clean data, and significantly higher scores in both robustness and explanation, indicating high detection accuracy even under more realistic conditions.

The unimodal models showed strong dependence on sources. Homoglyph substitution, token insertion and false subdomain creation were most problematic for URL-based detectors. The visual model was affected by changes to the logo, whilst changes to certificate, registration and hosting attributes, and webpage-layout manipulation impacted the metadata model. Multimodal fusion mitigated these limitations by letting unmanipulated modalities compensate for manipulated evidence. However, simple concatenation and fixed weighting were still less effective under coordinated attacks, as they could not dynamically suppress unreliable modality inputs.

The adaptive detector achieved robust accuracy of 90.84% in medium-severity multimodal attacks, and 85.72% in high-severity coordinated manipulation attacks. These results show that instance-specific modality weighting offers meaningful resilience when multiple phishing cues are modified simultaneously. The lower attack success rate also underscores the need for adaptive evidence integration in an ever-changing and AI-driven phishing landscape.

Explainability analysis revealed that reliable explanations require more than visually convincing heatmaps or feature rankings. The proposed evaluation considered (1) the extent to which highlighted features actually affected predictions, (2) the stability of these features under small perturbations, and (3) the consistency of the highlighted features with modality-level fusion behaviour.

The adaptive detector had the highest overall explainability score. It always highlighted elements relevant to phishing, such as misleading URL tokens, fake logos, form fields asking for credentials, domain age, redirects, and certificate inconsistencies. This alignment enhances the usability of explanations for analysts and deployment monitoring. Experimental results across datasets show that dataset shift remains a challenge. All models' performance dropped during cross-domain transfer from PhishTank to URLNet, but the adaptive detector's F1-score remained about 96% from the source domain. This stronger transfer performance indicates that using multimodal evidence and dynamic fusion helps minimise reliance on a dataset's URL patterns. The composite TrustScore was a more holistic comparison than an accuracy-based ranking. The adaptive detector received a TrustScore of 0.945 and remained at the top across different weighting schemes. Statistical testing and bootstrap analysis showed that these rankings were consistent and not driven by a few individual experimental runs. The results demonstrate the validity of PhishTrustXAI as a practical evaluation framework for accurate, robust, explainable, transferable, and deployable phishing detectors in ever-changing IoT/edge-security environments.

5.1. Limitations of the Study

This study has several limitations. First, the evaluation is based on public datasets of varying modalities, collection times, and labelling methods. As a result, each sample does not have a natural alignment between the URL, visual, and metadata records, which limits fully paired multimodal validation. Second, adversarial samples are created via controlled perturbations, which may not fully reflect the attack surface of real attackers, the dynamics of live web pages, or new AI-driven phishing techniques.

Third, the cross-dataset generalisation is mainly tested between PhishTank and URLNet. This provides evidence of distribution-shift resilience, but further testing and validation with other current phishing repositories, languages, regions, devices, and industrial IoT environments is needed. Fourth, the errors in the explanations are determined by fidelity, stability, and modality consistency, but are not formally evaluated through a user study with security analysts. As a result, the usefulness of the explanations in operation and the clarity in cognition need further research. The TrustScore also relies on predefined component weights and trust-level thresholds. Sensitivity analysis indicated that the ratings remained stable, but deployment-specific calibration may be required, depending on security priorities across environments. Last but not least, the computational evaluation only mimics edge deployment via CPU testing and does not include in-depth evaluation on a wide range of embedded hardware. Validation on IoT gateways and mobile devices, as well as on operational security platforms, remains needed in real time.

Multimodal records are created only when reliable cross-modal association exists, as it is not provided in the source datasets, such as naturally aligned URL - screenshot-metadata triplets. The adversarial evaluation uses synthetic perturbations with control mechanisms, not real-world

attacks. At present, the evaluation is performed only across available datasets and is not conducted with humans to validate the quality of the explanations. Although TrustScore assumes pre-defined equal weights as a neutral starting point, stability testing suggests the rankings remain consistent under different weightings. Lastly, hardware validation is limited to deployment measurements on a computational system rather than testing on physical IoT devices.

6. Conclusion and Future Work

In this study, PhishTrustXAI, a trust-aware evaluation framework, was presented to evaluate explainable multimodal phishing detection systems in changing IoT and edge environments. While predictive accuracy, adversarial robustness, explanation trustworthiness, and cross-dataset generalisation are considered separately, they are also holistically combined through a composite TrustScore, enabling consistent model ranking rather than focusing solely on clean-data accuracy. A composite TrustScore was used to incorporate these dimensions and enable consistent model ranking and assessment for deployment. An experimental evaluation based on URL, webpage-image, and domain-metadata information showed that the trust profile of adaptive multimodal detection produced the best and most balanced results. The adaptive detector obtained an F1 score of 97.96%, robust accuracy of 90.84% under medium-level coordinated attacks, and explanation and generalisation scores of 0.902 and 0.959, respectively. It outperformed fixed-weight, concatenation-based, and unimodal baselines with a final TrustScore of 0.945. Statistical testing, confidence intervals, bootstrap ranking analysis, and weight-sensitivity experiments further confirmed the observed improvements as stable and significant. The results show that dynamic modality weighting can reduce reliance on noisy evidence while maintaining robust explanations and cross-dataset performance.

Future research will scale up the evaluation to naturally aligned multimodal datasets that include matched URLs, webpage screenshots, HTML content, and real-time domain metadata. The adversarial test environment will be extended to incorporate live phishing campaigns, multi-language attacks, web pages created by generative AI, and adaptive attacks. Human-centred validation with security analysts will also be performed to evaluate explanation clarity, usefulness, and decision support. Further studies will focus on automated TrustScore weight calibration, uncertainty-aware trust estimation, federated trust evaluation, and hardware-level deployment on heterogeneous IoT gateways and embedded edge platforms. These extensions will enable broader operational validation of trustworthy phishing detection in dynamic cyber-physical environments.

References

[1] Bushra Sabir et al., "Reliability and Robustness Analysis of Machine Learning-based Phishing URL Detectors," *IEEE Transactions on Dependable and Secure Computing*, pp. 1-18, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Manh-Dung Nguyen et al., "A Deep Learning Anomaly Detection Framework with Explainability and Robustness," *Proceedings of the 18th International Conference on Availability, Reliability and Security*, Association for Computing Machinery, New York, United States, pp. 1-7, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Abdullah Fajar, Setiadi Yazid, and Indra Budi, "Comparative Analysis of Black-Box and White-Box Machine Learning Models in Phishing Detection," *arXiv*, pp. 1-15, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] Abdullah Alabdulatif, "A Novel Ensemble of Deep Learning Approach for Cybersecurity Intrusion Detection with Explainable Artificial Intelligence," *Applied Sciences*, vol. 15, no. 14, pp. 1-24, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] 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]

[6] Yanche Ari Kustiawan, and Khairil Imran Ghauth, "Feature Engineering for Phishing Website Detection using Machine Learning: A Systematic Review," *IEEE Access*, vol. 13, pp. 192080-192104, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[7] Maria Carla Calzarossa, Paolo Giudici, and Rasha Zieni, "An Assessment Framework for Explainable AI with Applications to Cybersecurity," *Artificial Intelligence Review*, vol. 58, no. 5, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[8] Shova Kuikel, Aritran Piplai, and Palvi Aggarwal, "Evaluating Large Language Models for Phishing Detection, Self-Consistency, Faithfulness, and Explainability," *arXiv*, pp. 1-7, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[9] Sara Mohammed Alasmari et al., "Phishing Detection in IoT: An Integrated CNN-LSTM Framework with Explainable AI and LLM-Enhanced Analysis," *Discover Internet of Things*, vol. 5, no. 1, pp. 1-53, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[10] Wenhao Li, Selvakumar Manickam, and Yung-Wey Chong, "FedPhishLLM: A Privacy-Preserving and Explainable Phishing Detection Mechanism using Federated Learning and LLMs," *Journal of King Saud University Computer and Information Sciences*, vol. 37, no. 8, pp. 1-32, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] Jun-Ho Yoon, Seok-Jun Buu, and Hae-Jung Kim, "Phishing Webpage Detection Via Multimodal Integration of HTML DOM Graphs and URL Features based on Graph Convolutional and Transformer Networks," *Electronics*, vol. 13, no. 16, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ali Aljofey et al., "An Effective Detection Approach for Phishing Websites using URL and HTML Features," *Scientific Reports*, vol. 12, no. 1, pp. 1-19, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[13] Asif Ejaz, Adnan Noor Mian, and Sanaullah Manzoor, "Life-Long Phishing Attack Detection using Continual Learning," *Scientific Reports*, vol. 13, no. 1, pp. 1-14, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[14] Fernando E. Casado et al., "Concept Drift Detection and Adaptation for Federated and Continual Learning," *Multimedia Tools and Applications*, vol. 81, no. 3, pp. 3397-3419, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[15] Gurgen Hovakimyan, and Jorge Miguel Bravo, "Evolving Strategies in Machine Learning: A Systematic Review of Concept Drift Detection," *Information*, vol. 15, no. 12, pp. 1-24, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[16] Jesher Joshua M et al., "Exploring the Efficacy of Federated-Continual Learning Nodes with Attention-Based Classifier for Robust Web Phishing Detection: An Empirical Investigation," *2024 International Conference on Advancements in Power, Communication and Intelligent Systems (APCI)*, Kannur, India, pp. 1-7, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[17] Chaoqun Guo et al., "Continual Learning for Intrusion Detection under Evolving Network Threats," *Future Internet*, vol. 17, no. 10, pp. 1-25, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[18] S. Kavya, and D. Sumathi, "Staying Ahead of Phishers: A Review of Recent Advances and Emerging Methodologies in Phishing Detection," *Artificial Intelligence Review*, vol. 58, no. 2, pp. 1-46, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[19] Amit Patel et al., "FL-MalDrift: A Federated Learning Framework for Malware Detection under Local Concept Drift," *Scientific Reports*, vol. 16, no. 1, pp. 1-26, 2026. [CrossRef] [Google Scholar] [Publisher Link]
[20] Sultan Refa Alotaibi et al., "Explainable Artificial Intelligence in Web Phishing Classification on Secure IoT with Cloud-based Cyber-Physical Systems," *Alexandria Engineering Journal*, vol. 110, pp. 490-505, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[21] Michael Georgiades, and Faisal Hussain, "An Explainable AI Approach for Interpretable Cross-Layer Intrusion Detection in the Internet of Medical Things," *Electronics*, vol. 14, no. 16, pp. 1-22, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[22] G. Nalinipriya et al., "Leveraging Explainable Artificial Intelligence for Early Detection and Mitigation of Cyber Threats in Large-Scale Network Environments," *Scientific Reports*, vol. 15, no. 1, pp. 1-24, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[23] Adel Alabbadi, and Fuad Bajaber, "An Intrusion Detection System Over IoT Data Streams using Explainable Artificial Intelligence," *Sensors*, vol. 25, no. 3, pp. 1-26, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[24] Hania Mohamed et al., "IoT-CAD: A Comprehensive Digital Forensics Dataset for AI-based Cyberattack Attribution Detection Methods in IoT Environments," *Ad Hoc Networks*, vol. 174, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[25] Abubakar Wakili, and Sara Bakkali, "A Resilient IoT Intrusion Detection System using Hybrid Feature Selection and Explainable Ensemble Learning," *Results in Engineering*, vol. 28, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[26] Divya Jennifer Dsouza, Anisha P. Rodrigues, and Roshan Fernandes, "Multimodal Comparative Analysis on Execution of Phishing Detection using Artificial Intelligence," *IEEE Access*, vol. 12, pp. 163016-163041, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[27] S. Kavya, and D. Sumathi, "Multimodal and Temporal Graph Fusion Framework for Advanced Phishing Website Detection," *IEEE Access*, vol. 13, pp. 74128-74146, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[28] Fouad Trad, and Ali Chehab, "Large Multimodal Agents for Accurate Phishing Detection with Enhanced Token Optimization and Cost Reduction," *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*, Dubai, United Arab Emirates, pp. 229-237, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[29] Mohammed Alsaedi et al., "Multimodal Features Representation-based Convolutional Neural Network Model for Malicious Website Detection," *IEEE Access*, vol. 12, pp. 7271-7284, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[30] Muhammad Waqas Shaukat et al., "A Hybrid Approach for Alluring Ads Phishing Attack Detection using Machine Learning," *Sensors*, vol. 23, no. 19, pp. 1-27, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[31] Rafael Abreu et al., "Enhancing IoT Security in Vehicles: A Comprehensive Review of AI-Driven Solutions for Cyber-Threat Detection," *AI*, vol. 5, no. 4, pp. 2279-2299, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[32] Omar A. Beg et al., "A Review of AI-based Cyber-Attack Detection and Mitigation in Microgrids," *Energies*, vol. 16, no. 22, pp. 1-23, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[33] Bandar Alotaibi, "A Survey on Industrial Internet of Things Security: Requirements, Attacks, AI-based Solutions, and Edge Computing Opportunities," *Sensors*, vol. 23, no. 17, pp. 1-49, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[34] Hussam N. Fakhouri et al., "AI-Driven Solutions for Social Engineering Attacks: Detection, Prevention, and Response," *2024 2nd International Conference on Cyber Resilience (ICCR)*, Dubai, United Arab Emirates, pp. 1-8, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[35] Haidar Jabbar, and Samir Al-Janabi, "AI-Driven Phishing Detection: Enhancing Cybersecurity with Reinforcement Learning," *Journal of Cybersecurity and Privacy*, vol. 5, no. 2, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[36] Chathuranga Sampath Kalutharage, Xiaodong Liu, and Christos Chrysoulas, "Neurosymbolic Learning and Domain Knowledge-Driven Explainable AI for Enhanced IoT Network Attack Detection and Response," *Computers and Security*, vol. 151, pp. 1-11, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[37] Abdulla Al-Subaiey et al., "A Novel Interpretable and Robust Web-based AI Platform for Phishing Email Detection," *Computers and Electrical Engineering*, vol. 120, pp. 1-23, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[38] Eman Abdullah Aldakheel et al., "A Deep Learning-based Innovative Technique for Phishing Detection in Modern Security using Uniform Resource Locators," *Sensors*, vol. 23, no. 9, pp. 1-27, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[39] Zainab Alshingiti et al., "A Deep Learning-based Phishing Detection System using CNN, LSTM, and LSTM-CNN," *Electronics*, vol. 12, no. 1, pp. 1-18, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[40] Tamara Al-Shurbaji et al., "Deep Learning-based Intrusion Detection Systems for Detecting IoT Botnet Attacks: A Review," *IEEE Access*, vol. 13, pp. 11792-11822, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[41] PhishTank: A Collaborative Clearing House for Phishing Data, PhishTank, 2024. [Online]. Available: https://www.phishtank.com

[42] Hung Le et al., "URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection," *arxiv*, pp. 1-13, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[43] Jing Wang et al., "LogoDet-3K: A Large-Scale Image Dataset for Logo Detection," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, no. 1, pp. 1-19, 2022. [CrossRef] [Google Scholar] [Publisher Link]