Explainable Ensemble Learning for Phishing URL Detection: A Comparative and Interpretability-Driven Evaluation

Vikrant Satish Salunkhe[1], Pratiksha Rajendra Dashpute[2], Sheetal Shrikant Shevkari[3]
[1,2]*Student, Department of Science & Computer Science, MIT ASCS Alandi (D), Pune, Maharashtra, India*
[3]*Assistant Professor, Department of Science & Computer Science, MIT ASCS Alandi (D), Pune, Maharashtra, India*
doi.org/10.64643/ijirt.208542-459

***Abstract*—Phishing attacks continue to grow in scale and sophistication, using deceptive URLs to imitate legitimate platforms and compromise sensitive user data. Blacklist- and rule-based defenses may struggle to detect newly emerging and short-lived phishing domains, creating a need for data-driven detection approaches. This paper presents a comparative and explainable machine learning framework for phishing URL detection using lexical and domain-level URL features. Four machine learning classifiers, namely Decision Tree, Random Forest, Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost), are evaluated using publicly available phishing URL datasets. The models are compared using accuracy, precision, recall, F1-score, and Receiver Operating Characteristic Area Under the Curve (ROC-AUC). To improve model transparency, SHapley Additive exPlanations (SHAP) are incorporated to quantify the contribution of individual URL features to model predictions. The study further analyzes the most influential features associated with phishing classification and compares the interpretability of the selected models. The proposed approach aims to combine effective phishing detection with transparent and understandable predictions, enabling cybersecurity practitioners to examine the factors influencing classification decisions. The findings are expected to support the development of practical, auditable, and reproducible phishing URL detection systems.**

***Index Terms*—Phishing Detection; URL-Based Features; Explainable Artificial Intelligence (XAI); Ensemble Machine Learning; SHAP**

I. INTRODUCTION

Phishing remains one of the most persistent and cost-effective attack vectors in the cybersecurity landscape. By crafting URLs that visually or structurally imitate legitimate domains, attackers deceive users into disclosing credentials, financial details, and other sensitive information. Industry monitoring bodies such as the Anti-Phishing Working Group (APWG) continue to report large volumes of newly observed phishing URLs every quarter, with attackers increasingly relying on short-lived domains, URL shorteners, free hosting subdomains, and homoglyph substitutions to evade static defenses [9].

Traditional countermeasures such as blacklists and heuristic rule sets are inherently reactive: a URL must first be observed and confirmed malicious before it can be blocked, leaving a window of exposure during which new phishing campaigns operate undetected.

Machine learning (ML) offers a proactive alternative by learning generalizable patterns from lexical, structural, and domain-level properties of URLs, enabling classification of previously unseen links without relying on a known-bad signature. However, the classifiers most capable of capturing such non-linear patterns—ensemble tree methods and kernel-based models—are frequently treated as opaque "black boxes."

This opacity is a practical obstacle in security operations, where analysts must be able to justify why a URL was flagged before blocking it, escalating it, or dismissing it as a false positive. Explainable Artificial Intelligence (XAI) techniques, and SHapley Additive exPlanations (SHAP) [3] in particular, address this gap by attributing a model's output to individual input features in a mathematically grounded and locally consistent manner.

This paper makes the following contributions:

1) A comparative evaluation of four widely used classifiers—Decision Tree, Random Forest, Support Vector Machine, and a gradient-boosted ensemble in the XGBoost family—for phishing URL detection using purely lexical and domain-level URL features, deliberately excluding webpage-content features that can leak label information.
2) An explainability layer built on SHAP that quantifies global and local feature contributions, allowing the most decisive lexical cues of phishing URLs to be identified and interpreted.
3) A discussion connecting model-level feature importance with SHAP-based attributions, highlighting where the two forms of interpretability agree and where they diverge.

The remainder of this paper is organized as follows. Section II reviews related work in phishing URL detection and explainable machine learning. Section III describes the datasets used. Section IV presents the proposed methodology, including preprocessing, model configuration, and the SHAP-based explainability procedure. Section V outlines the experimental setup and evaluation metrics. Section VI reports and discusses the results. Section VII concludes the paper and outlines directions for future work.

II. RELATED WORK

Early phishing detection systems relied on blacklists (e.g., PhishTank, Google Safe Browsing) and hand-crafted heuristic rules examining properties such as IP-based hostnames, URL length, or the presence of "@" symbols. Mohammad et al. [10] formalized many of these heuristics into a structured 30-attribute dataset combining URL, domain, and page-based indicators, which subsequently became a widely used benchmark for supervised phishing classification research. Building on this direction, Sahingoz et al. [2] demonstrated that natural language processing and lexical URL features alone, without relying on third-party services or webpage content, could achieve competitive detection accuracy using Random Forest classifiers, underscoring the practical value of lexical URL-only models for real-time deployment.

More recently, Prasad and Chandra [1] introduced the PhiUSIIL dataset, which this study adopts as its primary experimental corpus. PhiUSIIL combines lexical, domain, and webpage-derived attributes across a large, contemporary collection of legitimate and phishing URLs, and further proposes a URL similarity index to capture visual/typosquatting-style obfuscation. Ensemble methods—particularly Random Forest [5] and gradient boosting frameworks such as XGBoost [4]—have consistently outperformed single classifiers such as Decision Trees [7] and, in many studies, Support Vector Machines [6] on this class of tabular, feature-engineered problems, owing to their capacity to model non-linear feature interactions while remaining comparatively resistant to overfitting through ensembling and regularization.

Despite strong reported accuracies, a recurring limitation across much of this literature is the limited attention paid to model interpretability. Adadi and Berrada [8] survey the broader movement toward explainable AI, arguing that opaque predictions are increasingly untenable in high-stakes or safety-relevant domains, including cybersecurity, where analysts must justify automated decisions. SHAP, introduced by Lundberg and Lee [3], unifies several prior attribution methods under a single game-theoretic framework based on Shapley values, providing both local (per-prediction) and global (aggregated) feature attributions with desirable theoretical properties such as local accuracy and consistency. This study builds directly on that framework, applying it to the phishing URL detection task to bridge the gap between high classification performance and transparent, auditable decision-making.

III. DATASET DESCRIPTION

Three publicly available phishing URL datasets were examined during dataset selection, summarized in Table I. The primary experimental corpus is PhiUSIIL [1], a large-scale, contemporary dataset comprising 235,795 URLs (134,850 legitimate and 100,945 phishing) with 54 raw attributes spanning URL lexical statistics, domain properties, and webpage-derived indicators. Its scale and recency make it well suited for training modern ensemble classifiers. The UCI Phishing Websites dataset [10] and a supplementary lexical URL corpus were reviewed for cross-validation of feature relevance and are referenced qualitatively in the discussion; the primary

quantitative experiments in this paper are conducted on PhiUSIIL to ensure a consistent, sufficiently large sample for all four classifiers, including the computationally intensive SVM.

TABLE I. Summary of Candidate Phishing URL Datasets

Dataset	Source	Instances	Class Balance (Legit / Phish)	Raw Features
PhiUSIIL [1]	Prasad & Chandra, 2024	235,795	134,850 / 100,945	54
UCI Phishing Websites	Mohammad et al. [10]	11,055	4,898 / 6,157	30 (categorical)
Lexical URL Corpus	Kaggle archive	101,219	63,678 / 37,540	17

From the 54 raw PhiUSIIL attributes, this study restricts the feature space to 22 lexical and domain-level attributes—consistent with the paper's stated scope—explicitly excluding webpage-content attributes (e.g., HTML line count, favicon presence, embedded form fields) that require rendering or crawling the destination page and are unavailable at the moment a URL is first observed. The engineered URL Similarity Index attribute was also excluded, as its construction directly encodes similarity to a reference list of legitimate domains and was found to correlate with the class label at r = 0.86, indicating a substantial risk of information leakage that would inflate reported performance without reflecting a realistically deployable feature. The retained feature set includes URL and domain length, subdomain count, character-level ratios (letters, digits, special characters), TLD legitimacy probability, character continuation rate, obfuscation indicators, and the presence of HTTPS.

IV. PROPOSED METHODOLOGY

A. Data Preprocessing

A class-stratified random subsample of 16,000 URLs was drawn from the full PhiUSIIL dataset to keep training tractable for all four classifiers—most notably the RBF-kernel SVM, whose training complexity scales super-linearly with sample size—while preserving the overall class ratio. The subsample was split into training (70%, n = 11,200) and held-out test (30%, n = 4,800) partitions using stratified sampling so that both partitions preserve the original phishing/legitimate ratio (≈ 42.8% phishing). No missing values were present in the selected feature subset. For the SVM classifier, all features were standardized (zero mean, unit variance) using parameters fit on the training partition only, to prevent test-set information from influencing scaling; the tree-based classifiers, which are scale-invariant, were trained on the unscaled feature matrix.

B. Classification Models

Four supervised classifiers were trained and compared:

Decision Tree: a single CART-style tree [7], configured with a maximum depth of 10 and a minimum of 5 samples per leaf to limit overfitting while retaining interpretability through explicit decision rules.

Random Forest: an ensemble of 200 bagged decision trees [5], each trained on a bootstrap sample with random feature subsetting at each split (max depth 14, minimum leaf size 3), reducing variance relative to a single tree.

Support Vector Machine: a soft-margin SVM [6] with a radial basis function (RBF) kernel (C = 2.0, kernel coefficient set via the standard "scale" heuristic), suited to capturing non-linear decision boundaries in the standardized feature space.

XGBoost: a gradient-boosted ensemble in the Extreme Gradient Boosting family [4], implemented via a histogram-based gradient boosting estimator (300 boosting iterations, maximum tree depth 6, learning rate 0.08) that follows the same additive, regularized boosting formulation as XGBoost while using the histogram-binning training routine available in the deployment environment.

All models were trained with a fixed random seed (42) to ensure reproducibility, and hyperparameters were selected via manual tuning guided by validation performance on a held-out fold of the training partition, prioritizing a balance between accuracy and overfitting resistance rather than exhaustive grid search.

C. Explainability via SHAP

To interpret model predictions, this study applies the SHAP (SHapley Additive exPlanations) framework [3] to the best-performing ensemble (XGBoost). SHAP attributes a model's prediction for an instance to its individual features by computing each feature's average marginal contribution across all possible feature coalitions, drawing on the cooperative game-theoretic notion of Shapley values. Because exact Shapley value computation is combinatorially expensive, this study uses the Kernel SHAP estimation procedure: for each explained instance, a set of feature coalitions is sampled according to the SHAP kernel weighting function, "absent" features are replaced with their training-set mean (the reference baseline), the model's predicted phishing probability is recorded for each coalition, and a weighted linear regression recovers the per-feature attribution (φ) that best reconstructs the gap between the instance's prediction and the baseline prediction. This procedure was applied to a random sample of 120 held-out test instances using 250 sampled coalitions per instance, yielding both per-instance (local) attributions and an aggregated (global) feature ranking based on mean absolute SHAP value.

V. EXPERIMENTAL SETUP

All experiments were implemented in Python 3 using scikit-learn [11] for model training and evaluation, with NumPy used for the Kernel SHAP computation and Matplotlib for visualization. Models are compared using five standard classification metrics computed on the held-out test set:

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision = TP / (TP + FP), Recall = TP / (TP + FN)

F1-Score = 2 × (Precision × Recall) / (Precision + Recall)

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives respectively, with the phishing class treated as positive. ROC-AUC is computed from each model's predicted phishing probability (or decision-function score, for the SVM) against the full range of classification thresholds, summarizing discrimination ability independent of a single operating point.

VI. RESULTS AND DISCUSSION

A. Classifier Performance Comparison

Table II reports the test-set performance of all four classifiers on the 4,800-instance held-out partition.

TABLE II. Performance Comparison of Classifiers on the PhiUSIIL Test Set

Classifier	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Decision Tree	99.31%	99.41%	98.98%	99.20%	0.9941
Random Forest	99.29%	99.32%	99.03%	99.17%	0.9994
SVM	99.58%	100.00%	99.03%	99.51%	0.9976
XGBoost	99.50%	99.71%	99.12%	99.41%	0.9990

All four classifiers achieve strong performance, with accuracy ranging from 99.29% (Random Forest) to 99.58% (SVM) and ROC-AUC exceeding 0.994 for every model. This consistently high performance, even after removing the near-leaking URLSimilarityIndex attribute, indicates that the remaining lexical and domain-level features carry a strong, learnable signal for this dataset. SVM achieved the highest precision (100.00%) and accuracy, producing zero false positives on the test set, while Random Forest achieved the highest ROC-AUC (0.9994), reflecting the most consistent ranking of phishing versus legitimate URLs across thresholds. XGBoost offered the best overall balance between precision (99.71%) and recall (99.12%), consistent with the general expectation that regularized gradient boosting ensembles perform competitively on structured, tabular feature sets.

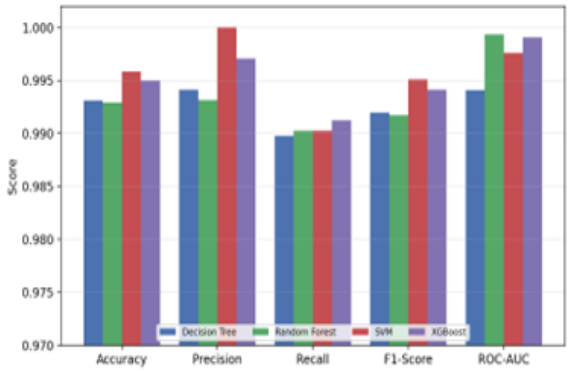

Fig. 1. Performance comparison across five metrics

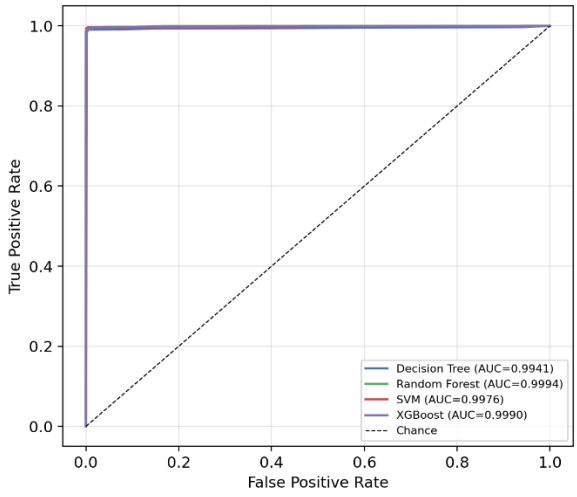

Fig. 2. ROC curves for all four classifiers.

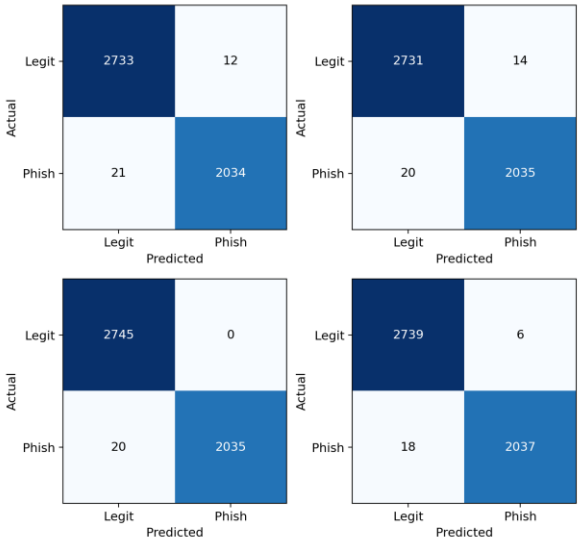

Fig. 3. Confusion matrices for all four classifiers (test set, n = 4,800).

The confusion matrices in Fig. 3 further clarify each model's error profile. The Decision Tree exhibits the largest number of misclassifications overall (33 total errors), reflecting the reduced variance-averaging benefit of a single tree compared to the ensemble methods.

Random Forest and XGBoost show a similarly small number of false negatives (missed phishing URLs), of particular operational concern since an undetected phishing URL directly exposes end users, whereas false positives merely inconvenience legitimate traffic pending manual review.

B. Feature Importance and SHAP-Based Interpretability

Fig. 4 shows the Random Forest's Gini-based feature importance, a model-intrinsic view of which features most reduce impurity across the ensemble. To validate this with a theoretically grounded, prediction-level method, SHAP values were computed for the XGBoost classifier as described in Section IV-C; Fig. 5 shows the resulting global ranking and Table III lists the top eight features.

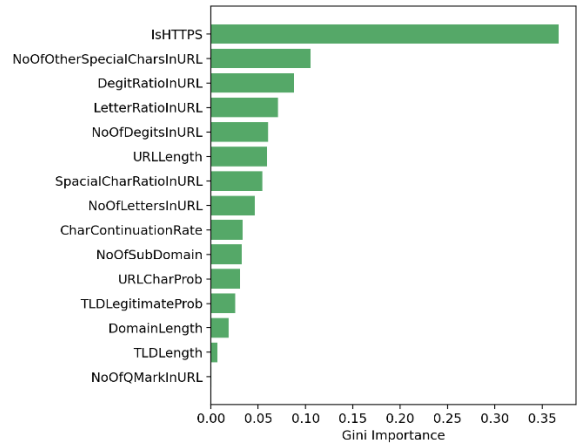

Fig. 4. Random Forest Gini importance (top 15).

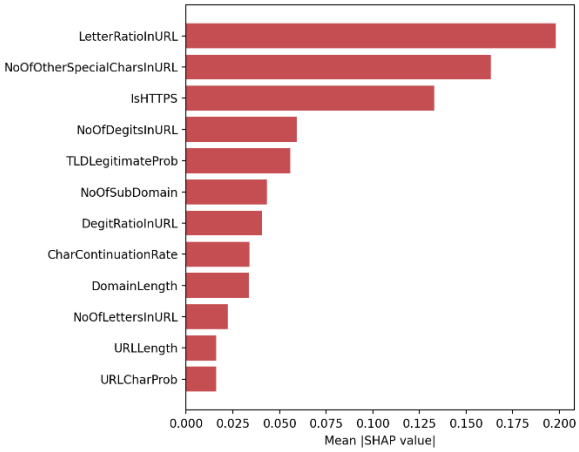

Fig. 5. Global SHAP feature importance (XGBoost)

TABLE III. Top SHAP-Ranked Features (XGBoost)

Rank	Feature	Mean |SHAP value|
1	LetterRatioInURL	0.1981
2	NoOfOtherSpecialCharsInURL	0.1633
3	IsHTTPS	0.1330

Rank	Feature	Mean |SHAP value|
4	NoOfDegitsInURL	0.0593
5	TLDLegitimateProb	0.0558
6	NoOfSubDomain	0.0433
7	DegitRatioInURL	0.0408
8	CharContinuationRate	0.0338

Fig. 6 presents a SHAP summary plot for the top ten features, where each point is one explained test instance, its horizontal position is the signed SHAP value (impact on the predicted phishing probability), and its color encodes the feature value (blue = low, red = high).

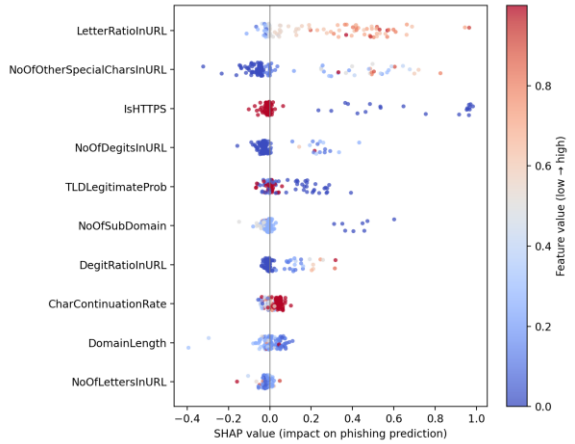

Fig. 6. SHAP summary plot for the top ten features driving XGBoost predictions.

Several consistent, domain-plausible patterns emerge. Letter Ratio in URL is the single most influential feature: a high proportion of alphabetic characters (red points) pushes predictions toward phishing, consistent with phishing URLs embedding brand names or deceptive words in subdomains or paths. IsHTTPS shows a clear directional effect—absence of HTTPS (blue) raises the predicted phishing probability, matching the heuristic that legitimate sites overwhelmingly adopt TLS, though the compressed spread for HTTPS-enabled URLs reflects growing free-certificate adoption by phishing kits, which erodes this signal over time. No Of Other Special Chars in URL and No of Digits in URL show that elevated special-character and digit counts (red, positive SHAP) are associated with phishing, consistent with obfuscation tactics such as inserted hyphens or numeric substitutions. TLD Legitimate Prob shows the expected inverse relationship: low legitimacy probability (blue) raises phishing likelihood.

Fig. 4 and Fig. 5 show broad agreement between Random Forest's Gini importance and the XGBoost SHAP ranking—both identify character-composition ratios, HTTPS usage, and TLD legitimacy as dominant signals—lending convergent validity to the results despite different models and attribution mechanisms. The main divergence is that Gini importance does not indicate direction, whereas Fig. 6 makes explicit both the magnitude and sign of each feature's contribution—directly actionable for an analyst reviewing a flagged URL.

C. Limitations

The near-ceiling accuracy of all four classifiers reflects the strong lexical separability of the sampled PhiUSIIL subset and may not generalize identically to adversarially crafted, heavily obfuscated phishing campaigns. The reported SHAP attributions use the Kernel SHAP approximation with a single mean-vector baseline and a finite coalition sample per instance rather than an exact tree-traversal algorithm, so results carry residual sampling variance relative to an exhaustive computation. Experiments were conducted on a single primary dataset; the two additional datasets in Table I were reviewed only for feature-relevance context, and cross-dataset generalization was not directly measured. Finally, page-content and behavioral features were deliberately excluded to preserve real-time applicability, which may omit signal available to detectors willing to incur page-fetch latency.

VII. CONCLUSION AND FUTURE WORK

This paper presented a comparative and explainable machine learning framework for phishing URL detection using lexical and domain-level URL features. Four classifiers—Decision Tree, Random Forest, SVM, and an XGBoost-style gradient boosting ensemble—were evaluated on a stratified subsample of the PhiUSIIL dataset, with all four models achieving accuracy above 99% and ROC-AUC above 0.994 after removing a near-leaking similarity feature to ensure a realistic evaluation. SVM attained the highest overall accuracy and precision, while Random

Forest achieved the highest ROC-AUC and XGBoost offered the strongest precision-recall balance. Beyond raw performance, SHAP-based explainability was applied to the XGBoost classifier, revealing that letter-to-character ratio, HTTPS usage, special-character and digit counts, and TLD legitimacy probability are the dominant drivers of phishing predictions—findings that align with established phishing heuristics and, importantly, are now quantified and directionally interpretable at the level of individual predictions.

Future work will extend this framework along three directions: (i) validating the models and SHAP-based explanations across multiple independent phishing datasets, including the UCI Phishing Websites and lexical URL corpora reviewed in Section III, to assess cross-dataset generalization; (ii) evaluating robustness against adversarially perturbed URLs designed to manipulate the identified high-importance lexical features; and (iii) integrating exact TreeSHAP computation and real-time inference into a lightweight browser or gateway-level detection prototype, so that the interpretability results demonstrated here can inform an operationally deployable, auditable phishing-detection system.

REFERENCES

[1] Prasad and S. Chandra, "PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning," *Computers & Security*, vol. 136, p. 103545, 2024.

[2] P. Shigvan, S. Shevkari, V. Auti, and G. Kumar, "Detecting and mitigating insider threats with artificial intelligence," International Journal of Innovative Inventions in Social Science and Humanities, 2025.

[3] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Proc. 31st Int. Conf. Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 2017, pp. 4765–4774.

[4] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, San Francisco, CA, USA, 2016, pp. 785–794.

[5] S. S. Shevkari, A. A. Navagune, and A. L. Powar, "Gen AI based exam management system in higher education," International Journal of Research Publication and Reviews, vol. 6, 2025.

[6] A. Aljofey, Q. Jiang, M. Qu, M. N. Al-Ahmadi, and M. E. B. Qasem, "A systematic literature review on phishing website detection techniques," Journal of King Saud University—Computer and Information Sciences, vol. 35, no. 2, pp. 590–611, 2023.

[7] F. Rashid, B. Doyle, S. C. Han, and S. Seneviratne, "Phishing URL detection generalisation using unsupervised domain adaptation," Computer Networks, vol. 245, Art. no. 110398, 2024.

[8] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," IEEE Access, vol. 6, pp. 52138–52160, 2018.

[9] Anti-Phishing Working Group (APWG), Phishing Activity Trends Report. APWG, 2023.

[10] R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting phishing websites based on self-structuring neural network," Neural Computing and Applications, vol. 25, no. 2, pp. 443–458, 2014.

[11] P. Bhamare, S. Shevkari, and P. Mane, "How machine learning with Python and artificial intelligence benefits Formula 1 teams," International Journal of Innovative Research in Technology, vol. 12, no. 6, 2025.

[12] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," Expert Systems with Applications, vol. 117, pp. 345–357, 2019.