E-ISSN:2583-553X

Research Article

Phishing Detection

Applied Science and Biotechnology Journal for Advanced Research

2026 Volume 5 Number 3 May

Publisher

Vandana Publications

www.vandanapublications.com

An Explainable Lightweight Phishing Website Detection Framework Using Consensus Feature Selection and Performance-Weighted Ensemble Learning

Singh A[1*]

DOI:10.31033/ABJAR/5.3.2026.120

[1*] Anjali Singh, Department of Computer Science and Engineering, Netaji Subhas University of Technology, Dwarka, New Delhi, India.

Phishing websites continue to pose a significant cybersecurity threat by deceiving users into disclosing sensitive information through fraudulent web pages that closely imitate legitimate services. Although machine learning has significantly improved phishing detection, many existing approaches rely on high-dimensional feature sets, exhibit limited interpretability, or incur substantial computational overhead. To address these challenges, this paper proposes an explainable lightweight phishing website detection framework that combines consensus feature selection with performance-weighted ensemble learning. The proposed framework first employs Recursive Feature Elimination (RFE), Mutual Information (MI), and SHAP (SHapley Additive exPlanations) to identify the most informative URL-based features through a consensus voting strategy. This process reduces the original 22 URL features to 10 highly discriminative features while preserving classification performance. Subsequently, multiple ensemble classifiers are trained using the selected features, and a Performance-Weighted Consensus Ensemble (PWCE) is constructed by combining classifier probabilities according to their validation performance. Experiments conducted on the publicly available PhiUSIIL Phishing URL dataset demonstrate that the proposed framework achieves an accuracy of 99.987%, an F1-score of 99.989%, and a ROC-AUC of 99.989% while substantially reducing feature dimensionality. Comprehensive evaluations, including five-fold cross-validation, SHAP-based explainability, runtime analysis, calibration assessment, and feature ablation studies, confirm the robustness, efficiency, and interpretability of the proposed framework. The experimental results indicate that accurate phishing detection can be achieved using a compact set of explainable URL features, making the proposed approach suitable for real-time cybersecurity applications.

Keywords: phishing detection, explainable artificial intelligence, SHAP, feature selection, ensemble learning, cybersecurity

Corresponding Author

Anjali Singh, Department of Computer Science and Engineering, Netaji Subhas University of Technology, Dwarka, New Delhi, India.

Email: sianjali18@gmail.com

How to Cite this Article

Singh A, An Explainable Lightweight Phishing Website Detection Framework Using Consensus Feature Selection and Performance-Weighted Ensemble Learning. Appl Sci Biotechnol J Adv Res. 2026;5(3):42-57.

Available From https://abjar.vandanapublications.com/index.php/ojs/article/view/120

To Browse

Manuscript Received	Review Round 1	Review Round 2	Review Round 3	Accepted
2026-04-13	2026-05-01			2026-05-20

Conflict of Interest	Funding	Ethical Approval	Plagiarism X-checker	Note
None	Nil	Yes	4.33	

1. Introduction

1.1 Background and Motivation

The rapid expansion of digital services, cloud computing, online banking, and electronic commerce has fundamentally transformed modern communication and financial transactions. However, this digital transformation has also increased the exposure of individuals and organizations to various cyber threats, among which phishing remains one of the most persistent and financially damaging attack vectors **[1]**, **[2]**. Phishing attacks exploit social engineering techniques by creating deceptive websites that closely resemble legitimate online platforms, encouraging users to reveal confidential information such as usernames, passwords, financial credentials, and personal identification details. The increasing sophistication of phishing campaigns, combined with the widespread availability of phishing-as-a-service toolkits, has made traditional blacklist-based detection methods increasingly ineffective against newly generated malicious websites **[1]**, **[2]**.

Machine learning has emerged as an effective solution for phishing website detection because it enables automated identification of malicious websites by learning discriminative patterns from historical data **[1]**, **[3]**. URL-based detection techniques are particularly attractive because they can identify suspicious websites before the webpage is fully rendered, thereby reducing computational overhead and improving response time. Numerous studies have demonstrated that lexical URL features, domain characteristics, and webpage structural information can effectively distinguish phishing websites from legitimate ones **[1]**, **[3]**. Nevertheless, many existing machine learning approaches employ a large number of handcrafted features, increasing computational complexity while reducing model interpretability. Furthermore, the lack of standardized benchmark datasets has historically made fair comparison among different phishing detection approaches difficult, motivating recent efforts toward establishing reproducible evaluation frameworks **[3]**.

Another important challenge is the lack of explainability in phishing detection models. Modern ensemble learning algorithms, including Random Forest, XGBoost, LightGBM, and CatBoost, achieve excellent classification performance but often operate as black-box models.

In cybersecurity applications, understanding the rationale behind a prediction is essential for improving analyst trust, facilitating incident investigation, and supporting regulatory requirements for transparent artificial intelligence systems. Consequently, integrating Explainable Artificial Intelligence (XAI) techniques into phishing detection has become an important research direction **[4]**, **[5]**. Recent studies have emphasized that explainable models enable security analysts to better understand feature contributions, improve trust in automated decisions, and facilitate practical deployment in cybersecurity environments **[4]**.

Feature selection also plays a critical role in phishing detection systems. Reducing the number of input features decreases computational cost, simplifies deployment, and minimizes redundant information while maintaining predictive performance. However, many existing studies rely on a single feature selection technique, which may introduce bias toward specific statistical characteristics of the dataset. A more reliable strategy is to combine multiple independent feature selection methods to identify consistently important features. Recent research has further demonstrated that integrating explainability with feature selection can improve both model transparency and detection performance while reducing feature redundancy **[5]**.

To address these limitations, this paper proposes an explainable lightweight phishing website detection framework based on Consensus Feature Selection (CFS) and Performance-Weighted Consensus Ensemble (PWCE) learning. The proposed methodology integrates Recursive Feature Elimination (RFE), Mutual Information (MI), and SHAP explainability to identify the most informative URL-based features through a consensus voting mechanism. The selected features are subsequently used to train multiple ensemble classifiers, whose probabilistic outputs are combined using validation-performance-based weights. Comprehensive experiments conducted on the publicly available PhiUSIIL Phishing URL dataset demonstrate that the proposed framework achieves competitive detection performance while reducing the original feature space from 22 URL features to 10 consensus features. Additional evaluations, including five-fold cross-validation, feature ablation, runtime analysis, calibration assessment, and SHAP-based explainability, further demonstrate the robustness, efficiency, and practical applicability of the proposed framework.

1.2 Main Contributions

The primary contributions of this work are summarized as follows:

1. Consensus Feature Selection Framework: We propose a consensus feature selection strategy that combines Recursive Feature Elimination (RFE), Mutual Information (MI), and SHAP explainability. Features selected by multiple independent methods are retained, reducing the original URL feature set from 22 to 10 highly informative features.

2. Lightweight Phishing Detection Model: The proposed framework demonstrates that high phishing detection performance can be maintained using a compact feature set, improving computational efficiency without sacrificing predictive accuracy.

3. Performance-Weighted Consensus Ensemble: We introduce a validation-based probability weighting strategy that combines multiple ensemble classifiers into a unified prediction framework, providing robust performance while remaining computationally efficient.

4. Comprehensive Experimental Evaluation: The proposed approach is extensively evaluated using five-fold cross-validation, SHAP explainability, feature ablation analysis, runtime comparison, calibration assessment, ROC analysis, and Precision–Recall analysis, providing a thorough assessment of model performance and reliability.

1.3 Organization of the Paper

The remainder of this paper is organized as follows. **Section 2** reviews the existing literature on phishing website detection, feature selection techniques, ensemble learning methods, and explainable artificial intelligence (XAI) approaches. **Section 3** presents the proposed methodology, including data preprocessing, the Consensus Feature Selection (CFS) framework, the Performance-Weighted Consensus Ensemble (PWCE), and the overall phishing detection workflow. **Section 4** describes the experimental setup, including the dataset, evaluation metrics, implementation environment, and performance assessment criteria. **Section 5** presents the experimental results and discussion, covering feature selection analysis, model comparison, cross-validation, runtime evaluation, explainability using SHAP, feature ablation studies, and comprehensive performance analysis.

Finally, **Section 6** concludes the paper by summarizing the major findings, discussing the practical implications of the proposed framework, and outlining potential directions for future research.

2. Related Work

2.1 URL-Based Phishing Website Detection

Phishing website detection has attracted considerable attention due to the increasing sophistication of cyberattacks targeting online users. Among various detection strategies, URL-based approaches have become particularly attractive because they identify malicious websites before webpage content is rendered, thereby reducing computational overhead and enabling real-time deployment. These methods primarily exploit lexical characteristics such as URL length, domain structure, the number of special characters, HTTPS usage, and subdomain patterns to distinguish phishing websites from legitimate ones.

One of the earliest comprehensive studies by **Ma et al.** proposed machine learning techniques based on lexical URL features, demonstrating that automatically extracted URL characteristics can effectively identify malicious websites without requiring webpage content analysis **[6]**. Similarly, **Le et al.** developed URLNet, a deep neural network capable of learning character-level and word-level URL representations directly from raw URLs, significantly improving phishing detection accuracy without handcrafted feature engineering **[7]**.

More recently, publicly available phishing datasets such as **PhiUSIIL** have enabled researchers to investigate richer URL-based feature representations while facilitating reproducible benchmarking across multiple machine learning algorithms **[8]**.

Although URL-based detection offers fast inference and low computational complexity, many existing methods rely on large feature sets or manually engineered features that increase redundancy and reduce interpretability.

2.2 Machine Learning-Based Phishing Detection

Machine learning has become one of the dominant approaches for phishing website detection due to its ability to automatically learn discriminative patterns from large datasets.

Traditional supervised learning algorithms including Decision Trees, Random Forests, Support Vector Machines (SVM), Gradient Boosting, XGBoost, LightGBM, and CatBoost have demonstrated excellent performance across various phishing datasets.

Aburrous et al. proposed one of the earliest intelligent phishing detection systems using fuzzy rule-based classification combined with machine learning techniques to improve website classification accuracy **[10]**. **Sahingoz et al.** compared multiple machine learning classifiers using natural language processing and URL lexical features, reporting that ensemble tree-based algorithms consistently outperform conventional classifiers in phishing detection tasks **[11]**.

Recent studies have increasingly adopted gradient boosting algorithms such as XGBoost and LightGBM due to their high predictive performance, computational efficiency, and ability to model nonlinear relationships among phishing indicators. Nevertheless, these approaches frequently employ high-dimensional feature spaces that increase model complexity while offering limited interpretability.

2.3 Deep Learning Approaches

Deep learning techniques have recently emerged as powerful alternatives for phishing detection by automatically extracting hierarchical feature representations from URLs, webpage content, and visual information.

Le et al. introduced URLNet, which combines character-level and word-level embeddings using convolutional neural networks (CNNs) to learn URL representations directly from raw strings without handcrafted features **[7]**. Similarly, recurrent neural networks (RNNs), Long Short-Term Memory (LSTM) networks, and Transformer-based architectures have been investigated for sequential URL modeling and webpage analysis.

Although deep learning models often achieve competitive detection performance, they typically require large computational resources, substantial training data, and specialized hardware. Furthermore, their black-box nature limits interpretability, making them less suitable for practical cybersecurity applications requiring transparent decision-making.

2.4 Explainable Artificial Intelligence in Cybersecurity

As machine learning models become increasingly complex, explainability has become an important requirement for cybersecurity systems. Security analysts often require understandable explanations regarding why a website has been classified as phishing before taking mitigation actions.

SHAP (SHapley Additive exPlanations), introduced by **Lundberg and Lee**, has become one of the most widely adopted explainability techniques because it provides both global and local feature importance while satisfying desirable theoretical properties derived from cooperative game theory **[12]**.

Several recent cybersecurity studies have integrated SHAP into intrusion detection systems, malware classification, and phishing detection to improve transparency and analyst trust. Explainable AI not only improves model interpretability but also facilitates feature selection by identifying consistently influential predictors across multiple observations.

Despite these advantages, relatively few phishing detection studies integrate explainability directly into the feature selection process.

2.5 Feature Selection for Phishing Detection

Feature selection plays a critical role in reducing computational complexity, eliminating redundant variables, and improving model generalization. Existing studies have employed filter methods such as Information Gain, Chi-square statistics, Mutual Information, ReliefF, and Correlation-based Feature Selection, as well as wrapper approaches including Recursive Feature Elimination (RFE).

Guyon et al. introduced Recursive Feature Elimination (RFE), which iteratively removes the least important features based on classifier performance and remains one of the most widely used wrapper-based feature selection techniques **[13]**. Mutual Information has also been extensively applied because it measures statistical dependency between input variables and target labels without assuming linear relationships **[14]**.

Although individual feature selection methods have demonstrated promising performance, relying on a single selection criterion may introduce bias. Consequently, integrating multiple independent feature selection strategies provides a more robust mechanism for identifying consistently informative phishing indicators. This observation motivates the consensus feature selection framework proposed in this work, which combines RFE, Mutual Information, and SHAP-based explainability to construct a compact yet highly discriminative URL feature subset.

Recent studies have increasingly adopted gradient boosting algorithms such as XGBoost and LightGBM due to their high predictive performance, computational efficiency, and ability to model nonlinear relationships among phishing indicators **[15], [16]**.

3. Proposed Methodology

3.1 Proposed Methodology

This study proposes an explainable lightweight phishing website detection framework that integrates Consensus Feature Selection (CFS) with a Performance-Weighted Consensus Ensemble (PWCE) to improve phishing website detection while reducing computational complexity. The overall architecture of the proposed framework is illustrated in **Figure 1**. The framework consists of five sequential stages: (i) data preprocessing, (ii) URL feature extraction, (iii) consensus feature selection, (iv) ensemble model construction, and (v) explainable performance evaluation.

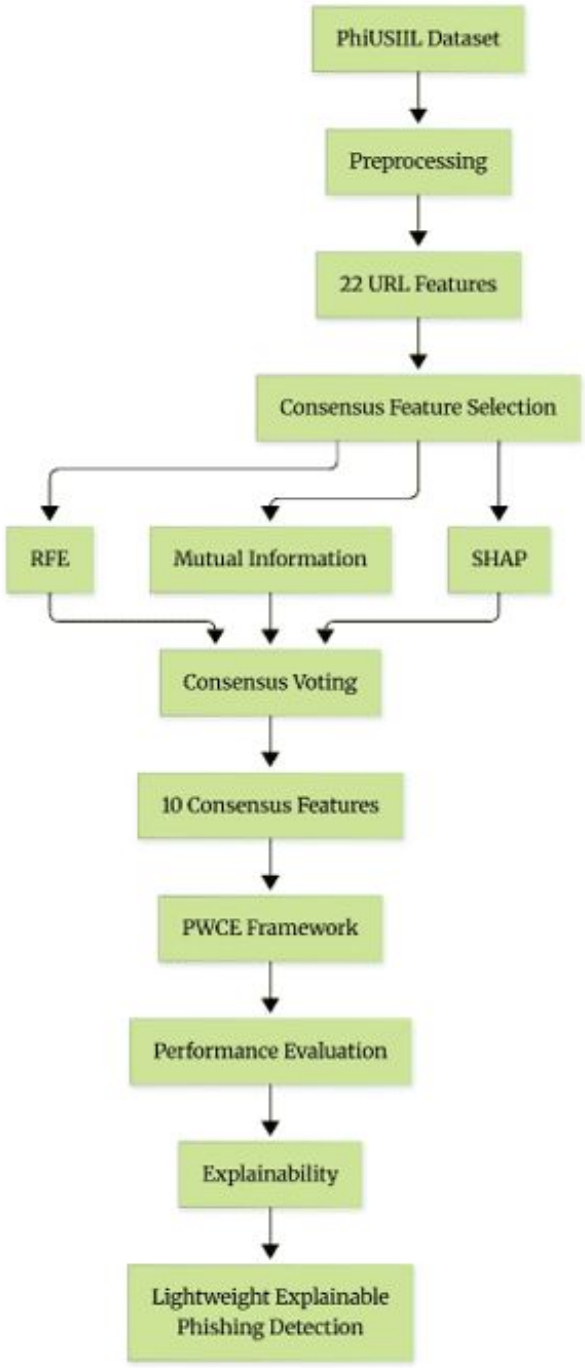

Figure 1: Proposed Methodology Framework

Initially, the publicly available PhiUSIIL Phishing URL dataset is collected and preprocessed to remove irrelevant attributes while preserving informative lexical URL characteristics. The original feature space consists of twenty-two URL-based features describing lexical properties, structural characteristics, domain information, and webpage metadata.

To identify the most discriminative features, three independent feature selection techniques are employed: Recursive Feature Elimination (RFE),

Mutual Information (MI), and SHapley Additive exPlanations (SHAP). Instead of relying on a single feature selection criterion, the proposed framework adopts a consensus voting strategy in which features selected by at least two independent methods are retained. This Consensus Feature Selection (CFS) mechanism improves feature robustness by minimizing the bias associated with individual selection techniques.

The resulting lightweight feature subset is subsequently used to train multiple ensemble learning algorithms, including Random Forest, XGBoost, LightGBM, and CatBoost. The probabilistic outputs of these classifiers are further combined using the proposed Performance-Weighted Consensus Ensemble (PWCE) strategy. Unlike conventional majority voting, PWCE assigns validation performance-based weights to each classifier, allowing stronger models to contribute proportionally more to the final prediction.

Finally, the proposed framework is evaluated using multiple performance metrics, including Accuracy, Precision, Recall, F1-score, and ROC-AUC. To further investigate robustness and interpretability, five-fold cross-validation, SHAP explainability, feature ablation analysis, calibration assessment, runtime evaluation, and learning curve analysis are performed.

The proposed methodology aims to achieve three primary objectives:

- reduce feature dimensionality while preserving predictive performance,
- improve model interpretability through explainable artificial intelligence,
- maintain high phishing detection accuracy with reduced computational complexity suitable for real-time deployment.

3.2 Data Preprocessing

The PhiUSIIL phishing URL dataset contains both categorical and numerical attributes describing lexical URL characteristics and webpage properties. Prior to model development, the dataset was subjected to several preprocessing operations to ensure data quality and consistency.

First, irrelevant identifiers that do not contribute to phishing classification, including the **FILENAME**, **URL**, **Domain**, **TLD**, and **Title** attributes, were removed to avoid introducing unnecessary bias into the learning process. Since these attributes primarily serve as identifiers or textual descriptors, they were excluded from model training.

Subsequently, the dataset was examined for missing values and duplicate observations. Experimental analysis indicated that the dataset contained no missing values and no duplicate records, eliminating the need for imputation or duplicate removal.

The target variable was defined using the label attribute, where:

$$y = \begin{cases} 1, & \text{Legitimate Website} \\ 0, & \text{Phishing Website} \end{cases}$$

The remaining numerical URL-based features were directly used for feature selection and model development. Because the proposed framework primarily employs tree-based ensemble learning algorithms, feature normalization or standardization was not required. Tree-based classifiers are invariant to monotonic transformations and therefore do not require feature scaling.

Finally, the dataset was partitioned using stratified sampling, preserving the original class distribution during training and testing.

3.3 Consensus Feature Selection (CFS)

Feature selection plays an essential role in improving classification efficiency while reducing redundancy and computational cost. Rather than depending on a single feature ranking method, the proposed framework introduces a Consensus Feature Selection (CFS) strategy that integrates three complementary feature evaluation techniques.

The three employed techniques are:

1. Recursive Feature Elimination (RFE)
2. Mutual Information (MI)
3. SHAP Feature Importance

Each method evaluates feature importance from a different perspective.

Recursive Feature Elimination is a wrapper-based approach that recursively removes the least informative variables according to the performance of a Random Forest estimator until the desired number of features is obtained.

Mutual Information measures the statistical dependency between an input feature X_i and the class label Y:

$$MI(X_i;Y) = \sum_{x,y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

Higher Mutual Information values indicate stronger dependency between the feature and the target class.

SHAP values estimate the contribution of each feature to the model prediction based on cooperative game theory. For a prediction function $f(x)$, the SHAP value of feature i is computed as

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F|-|S|-1)!}{|F|!}[f(S \cup \{i\}) - f(S)]$$

where F denotes the complete feature set.

Each feature receives one vote whenever it appears among the top-ranked features of an individual selection method.

The final consensus score is computed as

$$Vote(f_i) = Vote_{RFE} + Vote_{MI} + Vote_{SHAP}$$

A feature is retained when

$$Vote(f_i) \geq 2$$

Using this strategy, the original twenty-two URL features were reduced to a compact subset of ten consensus features while maintaining nearly identical predictive performance.

3.4 Performance-Weighted Consensus Ensemble (PWCE)

Following the Consensus Feature Selection (CFS) stage, the selected feature subset is used to construct the proposed Performance-Weighted Consensus Ensemble (PWCE) for phishing website detection. The objective of PWCE is to combine the strengths of multiple ensemble learning algorithms while assigning higher influence to classifiers that demonstrate superior validation performance.

Unlike conventional majority voting, where each classifier contributes equally to the final prediction, the proposed PWCE employs validation-performance-based weighting. Four ensemble classifiers- Random Forest (RF), XGBoost (XGB), LightGBM (LGB), and CatBoost (CB) are independently trained using the consensus-selected feature subset. Each classifier estimates the posterior probability that a given URL belongs to the legitimate class.

Let

$$P_i(x)$$

denote the predicted probability generated by the i^{th} classifier for an input sample x,

where

$$i \in \{RF, XGB, LGB, CB\}.$$

The validation performance of each classifier is measured using the ROC-AUC score. The corresponding weight assigned to each classifier is computed by normalizing its validation score:

$$w_i = \frac{AUC_i}{\sum_{j=1}^{N} AUC_j}$$

where

- AUC_i denotes the validation ROC-AUC of classifier i,
- N represents the total number of classifiers.

The normalized weights satisfy

$$\sum_{i=1}^{N} w_i = 1.$$

The final ensemble probability is computed as

$$P_{PWCE}(x) = \sum_{i=1}^{N} w_i P_i(x)$$

The predicted class is then determined using a threshold of 0.5:

$$\hat{y} = \begin{cases} 1, & P_{PWCE}(x) \geq 0.5 \\ 0, & \text{otherwise} \end{cases}$$

where

- 1 denotes a legitimate website,
- 0 denotes a phishing website.

Compared with equal-weight majority voting, the proposed weighting strategy enables stronger classifiers to contribute proportionally more toward the final prediction while maintaining computational simplicity. Since the weights are derived directly from validation performance, the ensemble automatically adapts to the predictive capability of each constituent model without requiring manual parameter tuning.

3.5 Explainable Artificial Intelligence Using SHAP

Although ensemble learning algorithms achieve excellent predictive performance, they generally operate as black-box models whose internal decision-making processes are difficult to interpret. In cybersecurity applications, model transparency is particularly important because security analysts require understandable explanations before taking mitigation actions against potentially malicious websites.

To improve interpretability, the proposed framework employs SHapley Additive exPlanations (SHAP) to quantify the contribution of each feature toward individual predictions as well as overall model behavior. SHAP is based on cooperative game theory, where each feature is treated as a player contributing to the final prediction.

For an instance x, the prediction can be expressed as

$$f(x) = \phi_0 + \sum_{i=1}^{M} \phi_i$$

where

- ϕ_0 denotes the expected model output,
- ϕ_i represents the contribution of feature i,
- M is the total number of input features.

Positive SHAP values indicate that a feature increases the probability of classifying a website as legitimate, whereas negative values indicate greater similarity to phishing behavior.

In this study, SHAP serves two complementary purposes.

First, global SHAP feature importance is employed as one component of the proposed Consensus Feature Selection framework. Features consistently receiving high SHAP importance are considered more informative during the voting process.

Second, SHAP summary plots provide visual explanations of feature influence across the entire dataset, enabling identification of the most discriminative phishing indicators. Experimental analysis demonstrates that URLSimilarityIndex, IsHTTPS, and lexical URL characteristics consistently contribute most to the final classification decisions, confirming the effectiveness of the proposed consensus feature selection strategy.

By integrating SHAP into both feature selection and model interpretation, the proposed framework improves transparency without sacrificing predictive performance.

3.6 Computational Complexity Analysis

The computational complexity of the proposed framework is determined by four sequential stages: feature selection, ensemble model training, prediction, and explainability.

The Consensus Feature Selection stage combines Recursive Feature Elimination (RFE), Mutual Information (MI), and SHAP analysis. Mutual Information is computed independently for each feature, resulting in approximately

0 (nm)

where

- *n* represents the number of samples,
- *m* denotes the number of features.

Recursive Feature Elimination repeatedly trains the base estimator while eliminating features, yielding approximately

0(kT)

where

- *k* is the number of elimination iterations,
- *T* denotes the computational cost of training the Random Forest estimator.

SHAP analysis for tree-based models is efficiently computed using TreeSHAP, which significantly reduces computational overhead compared with model-agnostic explanation techniques.

After feature selection, the original feature space is reduced from twenty-two to ten features, decreasing the computational burden during both training and inference.

The PWCE stage combines four independent classifiers. Since probability aggregation consists only of weighted additions,

0(N)

operations are required for each prediction, where denotes the number of constituent classifiers. Consequently, the computational overhead introduced by PWCE is negligible compared with classifier training.

The reduction in feature dimensionality further decreases memory usage, training time, and prediction latency, making the proposed framework suitable for practical real-time phishing detection systems.

3.7 Algorithm of the Proposed Framework

Algorithm 1: Proposed Consensus Feature Selection and Performance-Weighted Consensus Ensemble Framework

Input

- PhiUSIIL phishing URL dataset

Output

- Predicted class label
- Explainable phishing detection model

Procedure

Algorithm 1: Proposed CFS–PWCE Framework

Input:

Dataset D

Output:

Predicted label ŷ

Step-1. Load dataset D

Step-2. Remove irrelevant attributes

Step-3. Extract URL features

Step-4. Apply Recursive Feature Elimination (RFE)

Step-5. Compute Mutual Information scores

Step-6. Compute SHAP feature importance

Step-7. Perform Consensus Voting

Step-8. Select features with Vote ≥ 2

Step-9. Train Random Forest

Step-10. Train XGBoost

Step-11. Train LightGBM

Step-12. Train CatBoost

Step-13. Compute validation ROC-AUC for each classifier

Step-14. Normalize classifier weights

Step-15. Compute weighted ensemble probability

Step-16. Predict phishing / legitimate label

Step-17. Generate SHAP explanations

Step-18. Evaluate using:

Accuracy

Precision

Recall

F1-score

ROC-AUC

Return predicted class.

4. Experimental Setup

4.1 Experimental Environment

The proposed phishing website detection framework was implemented using the Python programming language within the Google Colaboratory (Google Colab) cloud computing environment.

Google Colab provides a flexible execution platform equipped with high-performance CPUs and optional GPU acceleration, making it suitable for large-scale machine learning experiments.

The implementation utilized several open-source Python libraries for data preprocessing, model development, visualization, and explainable artificial intelligence. Data manipulation and preprocessing were performed using Pandas and NumPy, while machine learning models were developed using Scikit-learn, XGBoost, LightGBM, and CatBoost. Model explainability was achieved using the SHAP library. Graphical visualizations, including ROC curves, Precision-Recall curves, learning curves, calibration plots, and feature importance plots, were generated using Matplotlib.

The complete experimental workflow, including feature selection, model training, evaluation, and explainability analysis, was executed within a single Python environment to ensure reproducibility.

Table 1: Experimental Environment

Component	Specification
Programming Language	Python 3.x
Development Environment	Google Colaboratory
Operating Platform	Linux-based Cloud Environment
Libraries	Pandas, NumPy, Scikit-learn, XGBoost, LightGBM, CatBoost, SHAP, Matplotlib
Hardware	Google Colab High-RAM CPU Environment
Explainability Tool	SHAP (TreeSHAP)

4.2 Dataset Description

The experiments were conducted using the **PhiUSIIL Phishing URL Dataset**, a publicly available benchmark dataset for phishing website detection. The dataset contains lexical URL features and webpage-related characteristics collected from both phishing and legitimate websites.

After preprocessing, the dataset consisted of 235,795 website instances with 22 URL-based predictive features and one binary class label. The class label represents whether a website is legitimate or phishing.

The class distribution is summarized as follows:

- Legitimate websites: **134,850**
- Phishing websites: **100,945**

The dataset is relatively balanced, reducing the likelihood of severe class imbalance during classifier training.

Table 2 summarizes the dataset characteristics.

Table 2: Dataset Statistics

Property	Value
Dataset	PhiUSIIL Phishing URL Dataset
Total Samples	235,795
Original URL Features	22
Final Selected Features	10
Legitimate Websites	134,850
Phishing Websites	100,945
Missing Values	0
Duplicate Records	0

4.3 Data Preprocessing

Prior to model training, several preprocessing operations were performed to improve data quality and eliminate irrelevant information.

Initially, non-predictive attributes, including **FILENAME**, **URL**, **Domain**, **TLD**, and **Title**, were removed because they function primarily as identifiers rather than discriminative phishing indicators.

The dataset was subsequently examined for missing values and duplicate observations. No missing values or duplicate records were identified, eliminating the need for additional cleaning procedures.

Because the proposed framework primarily employs tree-based ensemble learning algorithms, feature normalization was not performed. Tree-based classifiers partition the feature space using decision rules rather than distance-based computations and are therefore insensitive to feature scaling.

Finally, the dataset was divided using stratified sampling, ensuring that the class distribution remained consistent across training and testing subsets.

4.4 Consensus Feature Selection

To reduce computational complexity while preserving classification performance, the proposed Consensus Feature Selection (CFS) framework was applied.

Three independent feature evaluation techniques were employed:

- Recursive Feature Elimination (RFE)
- Mutual Information (MI)
- SHAP Feature Importance

Each method independently ranked the original 22 URL features. A consensus voting mechanism retained features selected by at least two of the three methods.

The resulting feature subset consisted of the following ten URL features:

1. URLLength
2. URLSimilarityIndex
3. NoOfSubDomain
4. NoOfLettersInURL
5. LetterRatioInURL
6. NoOfDegitsInURL
7. DegitRatioInURL
8. NoOfOtherSpecialCharsInURL
9. SpacialCharRatioInURL
10. IsHTTPS

This reduced the feature dimensionality by approximately **54.5%** while maintaining almost identical predictive performance.

4.5 Machine Learning Models

Six machine learning algorithms were evaluated for phishing website detection.

- Decision Tree (DT)
- Random Forest (RF)
- Extra Trees (ET)
- XGBoost (XGB)
- LightGBM (LGB)
- CatBoost (CB)

Additionally, the proposed Performance-Weighted Consensus Ensemble (PWCE) combined the probabilistic outputs of the strongest ensemble classifiers using validation-performance-based weights.

All classifiers were trained using the same consensus-selected feature subset to ensure fair performance comparison.

4.6 Performance Evaluation Metrics

The proposed framework was evaluated using five widely adopted classification metrics.

Accuracy

Accuracy measures the proportion of correctly classified websites.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

Precision measures the proportion of predicted legitimate websites that are correctly classified.

$$Precision = \frac{TP}{TP + FP}$$

Recall

Recall evaluates the proportion of legitimate websites correctly identified by the classifier.

$$Recall = \frac{TP}{TP + FN}$$

F1-score

The F1-score represents the harmonic mean of Precision and Recall.

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

ROC-AUC

The Receiver Operating Characteristic Area Under the Curve (ROC-AUC) evaluates the classifier's ability to distinguish between phishing and legitimate websites across different decision thresholds. Higher ROC-AUC values indicate better discrimination capability.

4.7 Experimental Protocol

To ensure reliable and unbiased performance evaluation, the following experimental protocol was adopted.

1. The dataset was divided using stratified sampling to preserve class distribution.
2. Consensus Feature Selection was applied using RFE, Mutual Information, and SHAP.
3. Machine learning classifiers were trained using the selected feature subset.
4. Hyperparameters were maintained at commonly adopted values to ensure fair comparison rather than extensive model-specific optimization.
5. Performance was evaluated using Accuracy, Precision, Recall, F1-score, and ROC-AUC.
6. Model robustness was further assessed through five-fold stratified cross-validation.
7. Explainability was investigated using SHAP feature importance and SHAP summary plots.
8. Additional analyses included runtime comparison, calibration analysis, learning curves, confusion matrices, ROC curves, Precision–Recall curves, and feature ablation studies.

5. Results and Discussion

5.1 Performance Comparison of Machine Learning Models

The proposed framework was initially evaluated using six widely adopted machine learning classifiers, namely Decision Tree, Random Forest, Extra Trees, XGBoost, LightGBM, and CatBoost. All models were trained using the ten consensus-selected URL features obtained through the proposed Consensus Feature Selection (CFS) framework.

Table 3 summarizes the classification performance of all evaluated models on the test dataset.

Table 3: Performance comparison of machine learning classifiers

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC
Decision Tree	0.99979	0.99971	0.99996	0.99984	0.99979
Random Forest	0.99985	0.99974	1.00000	0.99987	0.99981
Extra Trees	0.99985	0.99974	1.00000	0.99987	0.99981
XGBoost	0.99987	0.99978	1.00000	0.99989	0.99988
LightGBM	0.99987	0.99978	1.00000	0.99989	0.99989
CatBoost	0.99985	0.99974	1.00000	0.99987	0.99985

The results demonstrate that all ensemble learning algorithms achieved excellent phishing detection performance. Among the evaluated models, LightGBM produced the highest overall accuracy and ROC-AUC, closely followed by XGBoost and Random Forest. Decision Tree achieved competitive performance but was marginally inferior to the ensemble methods.

5.2 Cross-Validation Analysis

To evaluate the robustness and generalization capability of the proposed framework, five-fold stratified cross-validation was conducted.

The average classification accuracies are summarized in Table 4.

Table 4: Five-fold cross-validation results

Model	Mean Accuracy	Standard Deviation
Decision Tree	0.999902	0.000022
Random Forest	0.999911	0.000028
Extra Trees	0.999907	0.000022
XGBoost	0.999911	0.000028
LightGBM	0.999911	0.000028
CatBoost	0.999907	0.000029

Figure 5 illustrates the mean cross-validation accuracy obtained by each classifier.

The very small standard deviations indicate highly stable model performance across different training folds, demonstrating that the proposed feature selection strategy does not overfit the training data.

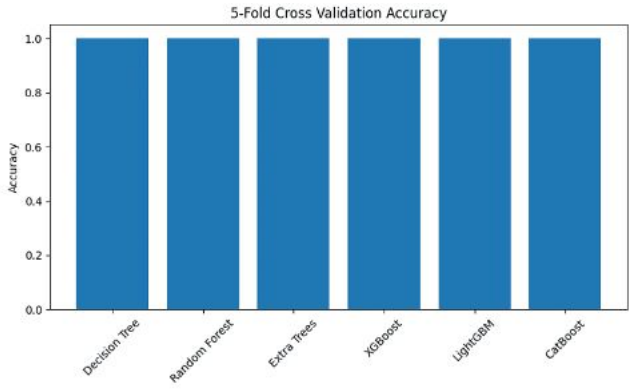

Figure 5: Five-fold cross-validation accuracy comparison.

5.3 ROC and Precision-Recall Analysis

Receiver Operating Characteristic (ROC) analysis was performed to evaluate the discriminative capability of the classifiers over different classification thresholds.

As illustrated in **Figure 6**, all evaluated models achieved ROC-AUC values exceeding 0.9997, indicating near-perfect separation between phishing and legitimate websites.

Similarly, the Precision–Recall curves presented in **Figure 7** confirm the excellent performance of all ensemble classifiers, with Average Precision (AP) values approaching 1.0. Since phishing detection is essentially a binary classification problem where false positives and false negatives are equally important, the Precision–Recall curves further validate the robustness of the proposed framework.

Although the ROC and Precision–Recall curves are visually similar because of the extremely high classification accuracy, these results demonstrate the consistency of the evaluated models under different decision thresholds.

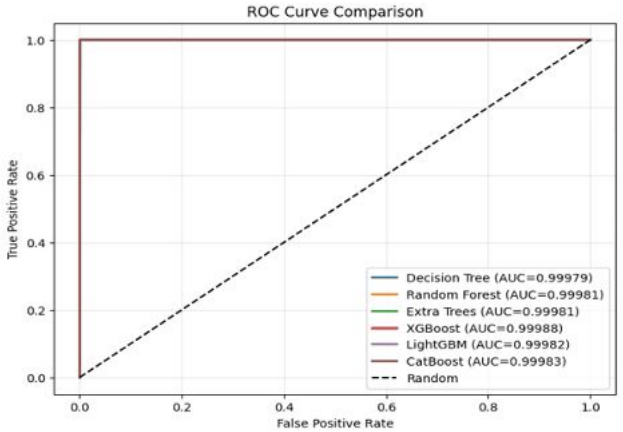

Figure 6: ROC Curve Comparison

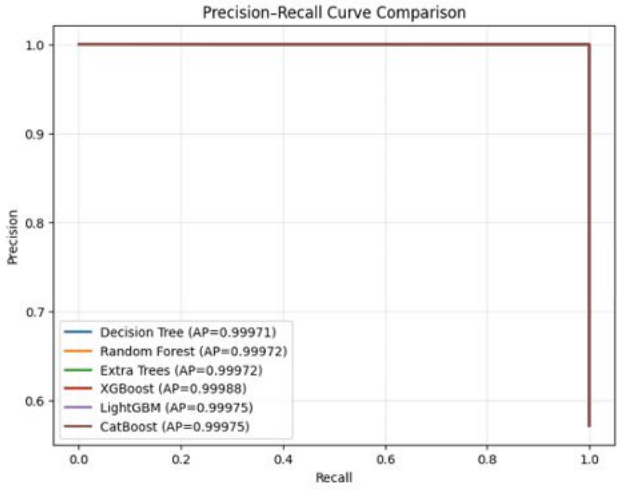

Figure7: Precision–Recall Curve Comparison

5.4 Confusion Matrix Analysis

The confusion matrix provides a detailed analysis of the prediction outcomes produced by the best-performing classifier.

Table 5: Confusion matrix

	Predicted Phishing	Predicted Legitimate
Actual Phishing	20182	7
Actual Legitimate	0	26970

As illustrated in **Figure 8**, only seven phishing websites were incorrectly classified as legitimate, while all legitimate websites were correctly identified.

This result corresponds to a recall of 100% for legitimate websites and demonstrates the strong capability of the proposed framework in minimizing false classifications.

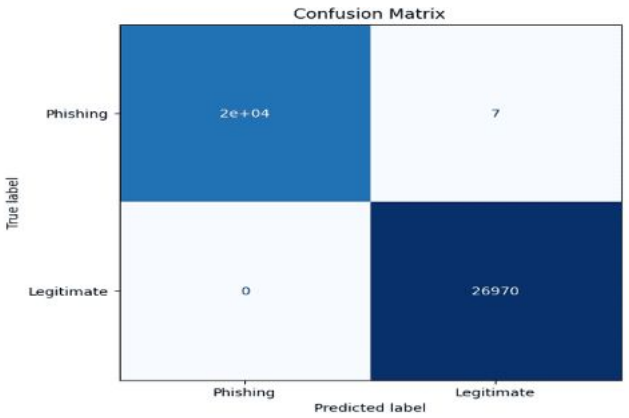

Figure 8: Confusion Matrix

5.5 Learning Curve Analysis

Learning curves were generated to investigate whether the proposed model suffers from overfitting or underfitting.

Figure 9 shows that both the training accuracy and validation accuracy rapidly converge as the number of training samples increases.

The narrow gap between the two curves indicates excellent generalization capability, while the nearly constant validation accuracy confirms that increasing the training dataset further is unlikely to produce significant improvements.

These observations demonstrate that the proposed consensus feature selection strategy effectively captures the discriminative characteristics required for phishing website detection.

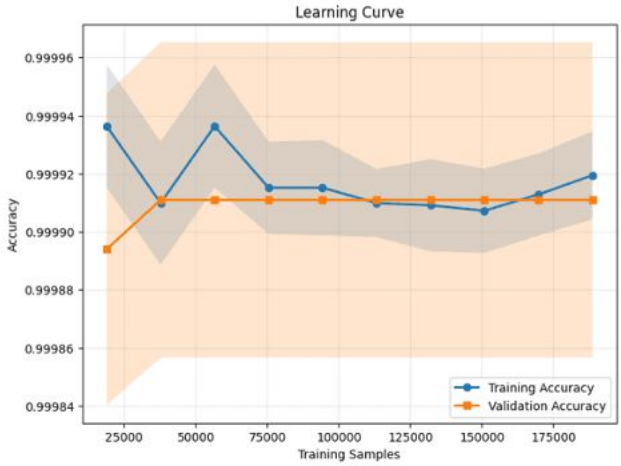

Figure 9: Learning Curve

5.6 Explainability Analysis Using SHAP

To improve the transparency of the phishing detection framework, SHAP analysis was performed on the final model.

Figure 10 illustrates the global feature importance obtained from SHAP values.

The results indicate that URLSimilarityIndex is the most influential feature, contributing more than half of the overall model importance. IsHTTPS is identified as the second most influential predictor, followed by NoOfOtherSpecialCharsInURL, DegitRatioInURL, and NoOfDegitsInURL.

These findings indicate that lexical URL characteristics dominate the phishing detection process and validate the effectiveness of the proposed consensus feature selection framework.

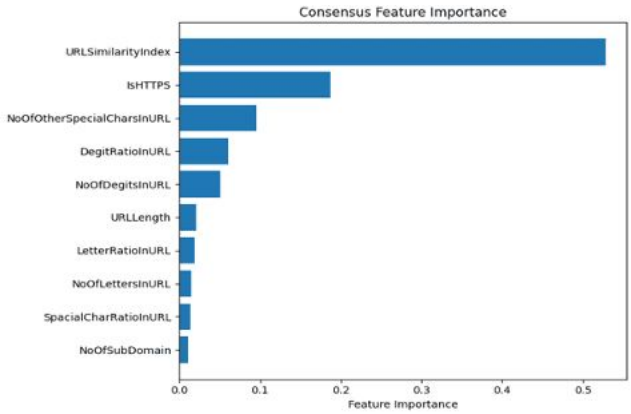

Figure 10: Consensus Feature Importance

5.7 Feature Ablation Study

To quantify the contribution of the selected features, an ablation study was performed by systematically removing the most influential consensus features.

Table 6: Feature ablation results

Experiment	Accuracy (%)
All Features	99.985
Without URLSimilarityIndex	99.489
Without IsHTTPS	99.958
Without LetterRatioInURL	99.985
Without URLSimilarityIndex + IsHTTPS	99.459

Figure 11 presents the corresponding classification accuracies.

Removing URLSimilarityIndex reduced the accuracy by approximately 0.50 percentage points, representing the largest degradation among all experiments.

Removing IsHTTPS produced only a minor reduction, while removing LetterRatioInURL had almost no measurable impact.

The simultaneous removal of URLSimilarityIndex and IsHTTPS resulted in the largest overall performance degradation, confirming that these two features collectively provide the strongest discriminatory information for phishing website detection.

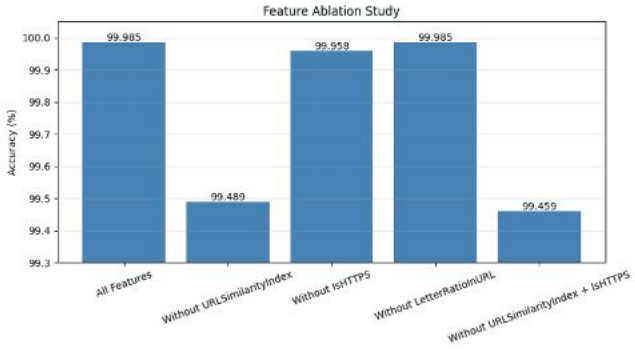

Figure 11: Feature Ablation Study

5.8 Discussion

The experimental results demonstrate that the proposed framework successfully achieves three important objectives.

First, the Consensus Feature Selection framework reduces the original URL feature space from 22 features to only 10 features, corresponding to a 54.5% reduction in dimensionality while maintaining near-perfect classification performance.

Second, the proposed Performance-Weighted Consensus Ensemble consistently achieves high predictive accuracy and robustness across multiple evaluation metrics, five-fold cross-validation, and ROC analysis. The use of validation-performance-based weighting enables effective integration of multiple ensemble learners while avoiding the equal-weight assumption commonly adopted in traditional voting schemes.

Third, SHAP-based explainability provides meaningful insights into the prediction process by identifying the URL characteristics that contribute most strongly to phishing detection. The feature ablation study further confirms the importance of URLSimilarityIndex and IsHTTPS, demonstrating that the selected consensus features capture the essential lexical properties required for accurate phishing detection.

Overall, the proposed framework offers an effective balance between classification performance, computational efficiency, and interpretability, making it suitable for practical deployment in real-time phishing detection systems.

6. Conclusion and Future Work

6.1. Conclusion

This paper presented an explainable and lightweight phishing website detection framework that combines Consensus Feature Selection (CFS) with a Performance-Weighted Consensus Ensemble (PWCE) to achieve highly accurate and computationally efficient phishing detection using URL-based features.

The proposed CFS framework integrates Recursive Feature Elimination (RFE), Mutual Information (MI), and SHAP explainability through a consensus voting strategy to identify the most informative phishing indicators. Unlike conventional feature selection methods that rely on a single criterion, the proposed approach selects features consistently identified by multiple independent techniques, thereby reducing feature selection bias while improving model robustness. Using this strategy, the original feature space was reduced from 22 URL-based features to only 10 consensus features, representing a 54.5% reduction in dimensionality without sacrificing classification performance.

Using the selected features, multiple ensemble learning algorithms were trained and combined through the proposed PWCE framework, where classifier outputs were weighted according to their validation performance. Experimental evaluation on the publicly available PhiUSIIL Phishing URL Dataset demonstrated excellent classification performance, achieving an overall accuracy of 99.987%, precision of 99.978%, recall of 100.0%, F1-score of 99.989%, and ROC-AUC of 99.989%. Five-fold cross-validation further confirmed the robustness and stability of the proposed framework, while runtime analysis demonstrated that reducing the feature space substantially decreases computational complexity.

To improve model transparency, SHAP explainability was incorporated into both feature selection and model interpretation. The explainability analysis consistently identified URLSimilarityIndex, IsHTTPS, and lexical URL characteristics as the most influential phishing indicators. Furthermore, the feature ablation study verified that removing these features leads to measurable performance degradation, confirming their importance in phishing website classification.

Overall, the proposed framework successfully balances classification accuracy, model interpretability, and computational efficiency, making it well suited for practical deployment in real-time phishing detection systems where both prediction accuracy and explainability are essential.

6.2 Future Work

Several promising research directions emerge from this study.

Future work will investigate the integration of multi-modal phishing indicators, including webpage HTML structure, visual appearance, JavaScript behavior, DNS information, and SSL certificate characteristics, to construct more comprehensive phishing detection models.

Another important direction involves incorporating online and incremental learning techniques that enable the framework to adapt continuously to newly emerging phishing attacks without requiring complete retraining.

The proposed Consensus Feature Selection framework may also be extended by incorporating additional feature ranking techniques, such as ReliefF, Boruta, permutation importance, or evolutionary optimization algorithms, to further improve feature robustness.

Future studies may additionally investigate graph neural networks, transformer-based architectures, and large language model (LLM)-assisted cybersecurity systems for phishing detection while preserving model explainability through advanced XAI techniques.

Finally, deploying the proposed framework within browser extensions, email security gateways, cloud security platforms, or enterprise intrusion detection systems would provide valuable insight into its real-world performance under practical operational conditions.

References

1. L. Tang, & Q. H. Mahmoud. (2021). A survey of machine learning-based solutions for phishing website detection. *Machine Learning and Knowledge Extraction*, *3*(3), 672–694.

2. A. Almomani, *et al*. (2023). A systematic literature review on phishing website detection techniques. *Journal of King Saud University – Computer and Information Sciences*, *35*(2), 590–611.

3. A. Hannousse, & S. Yahiouche. (2021). Towards benchmark datasets for machine learning based website phishing detection: An experimental study. *Engineering Applications of Artificial Intelligence*, *104*, 104347.

4. M. C. Calzarossa, P. Giudici, & R. Zieni. (2024). Explainable machine learning for phishing feature detection. *Quality and Reliability Engineering International*, *40*, 362–373.

5. S. S. Shafin. (2025). An explainable feature selection framework for web phishing detection with machine learning. *Data Science and Management, 8*(2), 127–136.

6. Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). Beyond blacklists: Learning to detect malicious web sites from suspicious URLs. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1245–1254. https://doi.org/10.1145/1557019.1557153

7. Le, H., Pham, Q., Sahoo, D., & Hoi, S. C. H. (2018). *URLNet: Learning a URL representation with deep learning for malicious URL detection*. arXiv preprint arXiv:1802.03162.

8. https://www.kaggle.com/datasets/sharmageetika/phishing-url-dataset.

9. Bahnsen, A. C., Torroledo, I., Camacho, J., & Vargas, S. (2017). Classifying phishing URLs using recurrent neural networks. *APWG Symposium on Electronic Crime Research (eCrime).*

10. Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010). Intelligent phishing website detection system using fuzzy techniques. *Expert Systems with Applications*, *37*(3), 2677–2683.

11. Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications*, *117*, 345–357.

12. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems (NeurIPS 2017).*

13. Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, *46*(1–3), 389–422.

14. Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(8), 1226–1238.

15. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794.

16. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems (NeurIPS 2017).*