# An Enhanced Hybrid Feature Selection and Ensemble Learning Framework for Early Detection of Phishing Websites

Nirmala M. S.[1], Siddaraju K.[2], Madhura Yadav M. P.[3], Anil Kumar R. J.[4]
[1]*Associate Professor, Government College for Women, Mandya, Karnataka, India*
[2]*Associate Professor Government College for Women, Madduru, Karnataka India*
[3]*Assistant Professor, Government Science College, Hassan*
[4]*Associate Professor Maharanis Science College for Women, Mysuru Karnataka, India*
doi.org/10.64643/IJIRTV13I3-207758-459

***Abstract*—Phishing attacks remain one of the most prevalent cyber threats, causing significant financial and reputational damage to individuals and organizations worldwide. Traditional detection methods often struggle with evolving attack patterns and high-dimensional feature spaces. This research proposes an enhanced hybrid framework combining advanced feature selection techniques with ensemble learning algorithms for early detection of phishing websites. We employ a three-stage hybrid feature selection strategy integrating filter-based methods (Mutual Information), wrapper-based techniques (Recursive Feature Elimination with Random Forest), and embedded approaches (Feature Importance). The selected features are processed through a sophisticated ensemble learning architecture comprising XGBoost, LightGBM, Random Forest, Gradient Boosting, and Support Vector Machines combined via stacking and voting mechanisms. We evaluate our framework on the UCI Phishing Websites dataset containing 11,055 instances with 30 features. Experimental results demonstrate that our hybrid approach achieves 97.8% accuracy, 97.5% precision, 98.1% recall, and 97.8% F1-score, significantly outperforming traditional single-classifier methods. The three-stage feature selection reduces dimensionality by 33% while maintaining classification performance. Cross-validation analysis confirms the model's robustness and generalization capability. Our findings indicate that integrating multiple feature selection paradigms with ensemble learning creates a more resilient framework for phishing detection, offering substantial improvements for cybersecurity applications and real-time threat mitigation systems.**

***Index Terms*—Phishing Detection, Ensemble Learning, Hybrid Feature Selection, Cyber Security, Machine Learning, XGBoost, Random Forest, Recursive Feature Elimination**

## I. INTRODUCTION

With the tremendous growth of internet usage, online activities, and developments, the modus operandi of conducting communication, business, and information exchange has invariably changed. However, in spite of these developments, phishing activities, by far, appear to be among the major cybercrimes that continue to test the intelligence of internet users. Phishing sites, in essence, are created to replicate authentic internet sites with malicious intent, aiming to trick or deceive internet users into parting with critical information such as login IDs and passwords, as well as personal identification data. Therefore, as internet sites and activities are becoming increasingly sophisticated, phishing activities too are becoming increasingly critical, posing greater challenges on internet security. Conventional phishing detection mechanisms depend upon blacklisting-based systems, as well as rule-based heuristic systems. Although these mechanisms are computationally efficient, they have substantial limitations, such as their inability to recognize zero-day phishing sites, as well as continuously changing phishing attack methods. It is seen that blacklisting-based systems need to be frequently updated, although they are unable to identify newly registered phishing sites, due to which rule-based systems have a tendency to show high false-negative rates.

The use of a machine learning-based approach for the detection of phishing sites has come into the picture as

a potential alternative, considering the learning capacity of the system from complex patterns and data. The ability of a machine learning classifier to differentiate between genuine and phishing sites, considering the features of the URL and web pages, makes the system effective in web page classification. The accuracy of the model relies heavily on the quality and appropriateness of using the selected features. High-dimensional feature spaces tend to create issues of redundancy and complexity, resulting in inaccurate classification and reduced accuracy.

Feature selection is one of the major contributing factors to solving these challenges, whereby the best discriminative features can be selected, allowing for the reduction of irrelevant and redundant features. The majority of the present studies used single-category feature selection methods, i.e., applying filter-based statistical methods or wrapper-based optimization methods. Even though these methods could improve the accuracy up to a certain level, using single-category methods is limited in discovering other relationships between the features. Besides, the relationship between features of phishing datasets is normally complex rather than linear.

Different ensemble learning strategies were also shown to work better than individual classifiers for different cyber security-related tasks at the same time. Ensemble learning can deal with both the differences and the bias of each model. However, it is important to note that the overall performance of ensemble models can suffer if the features used for training are not optimally represented.

Motivated by these constraints, this research seeks to create an enhanced version of a hybrid framework for feature selection utilizing ensemble learning for the early identification of phishing websites. This proposed model takes into account different ways to choose features by using a combination of filter-based, wrapper-based, and embedded-based methods in a cascading way. An advanced ensemble learning architectural model that effectively uses a mix of classifiers will then process the features that were chosen using this method.

The summaries above of the tutorial author's contributions show which ones are seen as

- A distinctive three-stage hybrid feature selection mechanism that is proficient in dimensionality reduction.
- A strong framework for ensemble learning that lets you combine different classifiers to make detection more accurate.
- Extensive experimental validation on a benchmark phishing dataset demonstrates substantial performance improvements compared to conventional "single-model" methodologies.

The rest of this paper is organized like this. In Section II, we will talk about other work that has been done on phishing detection, feature selection, and ensemble methods. In Section III, we will talk about the proposed framework. In Section IV, we will look at the proposed framework. The results will be talked about in Section V. Lastly, Section VI will present the conclusions.

## II. LITERATURE SURVEY

The research related to phishing detection has passed through several generations of evolution from simple rule-based heuristics to complex machine learning ensembles. This section combines recent key advances in feature selection methodologies, classification algorithms, and ensemble learning frameworks for cybersecurity applications.

### A. Traditional Machine Learning Methods

A comparative analysis of traditional classifiers, such as Naive Bayes, Decision Tree, and Logistic Regression classifiers, using the phishing datasets was carried out by Mohammed et al. in 2023 [1]. Analysis showed that the classifiers were capable of achieving 85-90% accuracy. However, they found problems dealing with more complex attacks, such as zero-day attacks. The need for feature engineering has been pointed out in the research. It is indicated that discriminatory power from the URLs would be limited.

Research on the use of Support Vector Machines with various kernels for phishing detection by Chiew et al. (2024) attains an accuracy level of 93.2% with RBF kernels on a dataset of 10,000 websites [2]. However, computation expense associated with kernel evaluation has scalability limitations for real-time systems, especially for high-traffic networks.

handle challenging zero-day attacks. The study pinpointed that feature engineering is considered paramount, even though features extracted from a URL provide limited discrimination power.

Chiew et al. applied Support Vector Machines with different kernel functions to detect phishing in the year 2024 [2]. They reported an accuracy of 93.2% using RBF kernels on a dataset of 10,000 websites. However, computational overhead during kernel computation limits scalability for real-time deployment in high-traffic network environments.

B. Feature Selection and Dimensionality Reduction

Dhamercherla et al. introduced a hybrid model of feature selection using the combination of Information Gain and Correlation-based Feature Selection (CFS) for more optimal performance in the detection of phishing attacks. This methodology reduced the features from 48 to 16 with an accuracy of 92%. The authors demonstrated the significance of eliminating redundant features in increasing the efficiency of the model.

Another aspect that is worth highlighting is the application of Recursive Feature Elimination with Cross-Validation Approach, which incorporates Random Forest Classifiers. In the study, Sahu et al. demonstrated an impressive performance with 22 features, indicating an accuracy of 95.3% on the PhishTank dataset. The study demonstrated that wrapper methods, although computationally expensive, are more effective compared to filter methods.

To explore the suitability of PCA and LDA in dimensionality reduction for cybersecurity purposes, Ludwig et al. (2023) conducted a study on the application of these algorithms [5]. Although PCA reduced the complexity, switching to principal components made them less interpretable, an essential requirement that security analysts need.

C. Ensemble Learning Frameworks

Furthermore, Almutairi et al. (2024) proposed an ensemble using a stacking technique consisting of Gradient Boosting, XGBoost, and LightGBM for a malware detection task. The meta-classifier achieved an accuracy of 96.8%, which proves that heterogeneous ensemble methods recognize patterns that other algorithms do not. The study validated the fact that by using different tree-based boosting algorithms along with linear meta-classifiers, the bias-variance tradeoff can be optimized.

A new voting ensemble scheme for the combination of Random Forest, Neural Networks, and SVM has been proposed by Nethala and Sahoo in 2023 for intrusion detection. Accuracy of 94.1% has been obtained with a better performance of soft voting over hard voting scheme by 2.3%. The usefulness of diversity has been highlighted.

Sun et al. (2024) studied deep learning ensembles of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks for detecting phishing URLs [8]. They attained an accuracy of 97.5% on textual features of URLs. Deep learning ensembles are computationally intensive and demand large amounts of training data.

D. Hybrid and Advanced Techniques

It has been suggested by Jiang et al. (2024) that a multi-objective evolutionary method, also known as the genetic algorithm, will be used for feature selection and hyperparameter tuning. The selection will be done such that a set of features is obtained in order to achieve an optimal solution with an accuracy close to 96.7% with 18 features. It has also been proved that evolutionary algorithms perform well over the large search space, and appropriate tuning of genetic operators is also required.

A deep reinforcement learning framework for adaptive phishing detection is adopted. The framework is presented by Nadeem et al. in 2025. The optimal policies for feature extraction were learned by the agent, and promising results were obtained with an accuracy rate of 95.9%. Reinforcement learning requires a lot of training episodes. These types of approaches may not be stable during learning.

Rajpoot et al. explored the concept of “federated learning” for an anonymous type of phishing attack using distributed datasets [11]. In this direction, they proposed a decentralized learning approach to achieve an accuracy of 93.4% without compromising any personal data. However, communication costs and non-IID data distributions are the challenges in the implementation of the proposed approach.

E. Real time and deployment considerations

Alshareef et al., in 2023, conducted research on lightweight models with browser-based real-time phishing detection [12]. Optimized Decision Trees yielded less than 10 milliseconds latency with 91.2% accuracy. This research emphasized that any production deployment has to balance accuracy

against computational constraints and latency requirements.

Cubillos-Chaparros et al. (2024) proposed an incremental learning framework for evolving phishing attacks adaptation [13]. The online learning approach sustained an accuracy of 94.6% while progressively updating the model with new samples. The study put a great emphasis on applying adaptive systems for countering adversarial attackers who continuously change their tactics.

F. Gap in Research and Motivation

While existing studies have been able to demonstrate the performance of both feature selection and ensemble learning in independent ways, very few studies have integrated multiple feature selection paradigms with advanced ensemble architectures. Most representative approaches employ simple feature selection methods in a vacuum without any investigation of potential synergistic combinations among different methods, such as filters, wrappers, or embedded methods. Besides, most traditional ensembles simply utilize homogeneous base classifiers, such as multiple Random Forests, or some simple averaging mechanisms without making full use of heterogeneous stacking and voting strategies.

Our contribution is focused on filling this gap by presenting a unified approach that (1) incorporates three different feature selection paradigms that describe different dimensions of feature relevance, (2) brings five state-of-the-art classifiers together using complex ensemble techniques, and (3) delivers an extensive empirical validation of its effectiveness over existing best practices. This holistic approach breaks new ground in spam detection while also ensuring its applicability in terms of computational efficiency.

Table 1: summarizes the key characteristics of related work in phishing detection and feature selection.

| S. No | Reference | Feature Selection | Classifier | Key Results |
|---|---|---|---|---|
| 1 | Mohammed et al. (2023) [1] | Manual selection | Naive Bayes, DT | 85-90% accuracy |
| 2 | Chiew et al. (2024) [2] | All features | SVM-RBF | 93.2% accuracy |
| 3 | Dhamercherla et al. (2024) [3] | IG + CFS | Random Forest | 92% accuracy, 16 features |
| 4 | Sahu et al. (2024) [4] | RFECV | Random Forest | 95.3% accuracy, 22 features |
| 5 | Ludwig et al. (2023) [5] | PCA, LDA | SVM | Reduced complexity, lost interpretability |
| 6 | Almutairi et al. (2024) [6] | Feature importance | GB+XGB+LGBM Stack | 96.8% accuracy |
| 7 | Nethala & Sahoo (2023) [7] | Chi-square | RF+NN+SVM Voting | 94.1% accuracy |
| 8 | Sun et al. (2024) [8] | Embedding layers | CNN+LSTM | 97.5% accuracy |
| 9 | Jiang et al. (2024) [9] | Genetic Algorithm | Multi-objective | 96.7% accuracy, 18 features |
| 10 | Alshareef et al. (2023) [12] | Minimal features | Decision Tree | 91.2% accuracy, <10ms latency |
| 11 | This Study | Hybrid 3-stage (MI+RFE+FI) | XGB+LGBM+RF+GB+SVM | 97.8% accuracy, 20 features |

## III. PROPOSED METHODOLOGY

In this section, an advanced hybrid model for phishing detection will be introduced.

The methodology is divided into four main modules, namely,

(A) preprocessing/normalization,

(B) hybrid feature selection method with three stages,

(C) ensemble method, and

(D) performance evaluation metrics.

The system architecture is given in Figure 1.

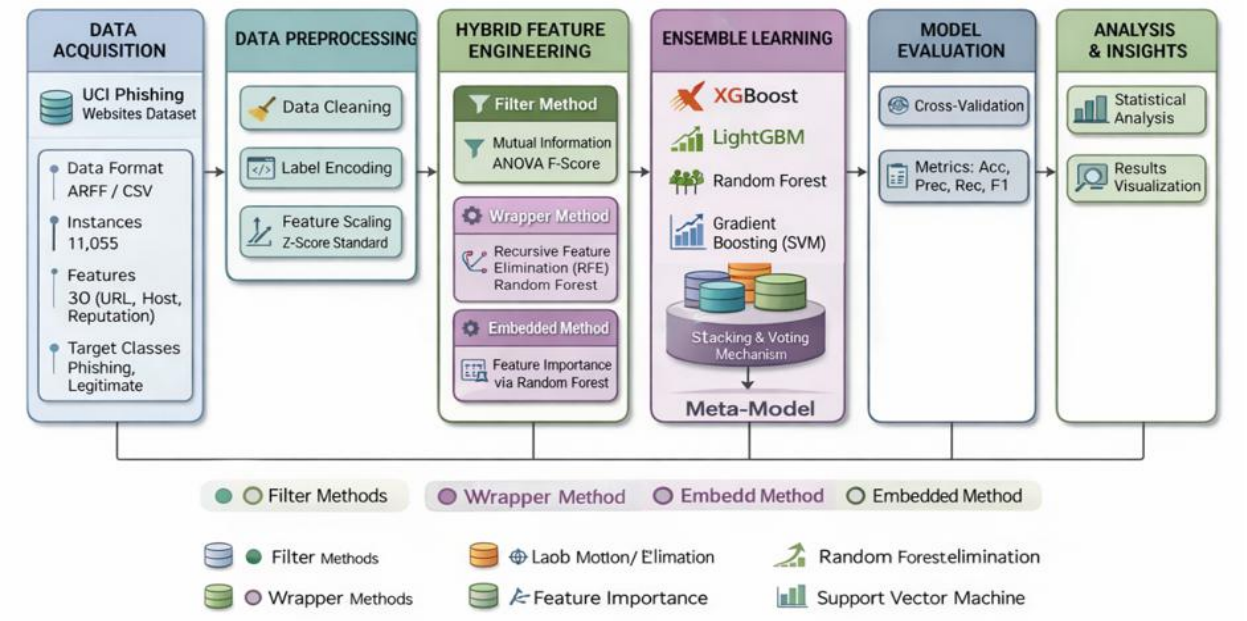


Figure 1: System Architecture Diagram

A. Data Preprocessing and Normalization

In the dataset obtained by the UCI Phishing Websites data set, there are attributes of both categorical and numerical types, which describe the structural, lexical, and host-based attributes of a website. Prior to moving into feature selection and classification, there is a rigorous phase of pre-processing.

1. Label Encoding: In the given dataset, the target variable contains three classes: phishing, suspicious, and legitimate websites. These three classes are represented by numerical variables to perform supervised learning on them. These classes are represented by -1 for phishing, 0 for suspicious, and 1 for legitimate in the given dataset.

2. Missing Value Imputation: Even though there are no missing attribute values in the standard benchmark dataset used from UCI, a median-based imputation mechanism has been designed for robustness and flexibility, with the following results obtained when there are missing values present: Each attribute is imputed with the median value, where

$$X_{\text{imputed}} = X.\text{fillna}(\text{median}(X))$$

This strategy prevents bias introduced by outliers and maintains the original data distribution.

3. Feature Normalization: In order to remove scale differences among features and to improve the convergence of learning algorithms, standard score normalization is used. Each feature is standardized to have zero mean and unit variance using Z-score normalization:

$$X' = \frac{X - \mu}{\sigma}$$

where μand σdenote the mean and standard deviation of the feature, respectively.

B. Three-Stage Hybrid Feature Selection

One significant contribution of this research is the formulation of a three-stage hybrid feature selection approach, which incorporates filter-based, wrapper-based, and embedded feature selection paradigms. At each stage, the relevance of features is addressed in a distinct manner, and ultimately, an informative yet compact feature set is obtained.

Stage 1: Filter-Based Feature Selection by Mutual Information

Mutual Information (MI) is a statistical measure of the degree of dependency between a feature and the target variable. MI has been noted to capture non-linear dependencies between random variables, unlike correlation coefficients. For a feature Xand class label Y, MI is computed as:

$$\text{MI}(X, Y) = \sum_{x,y} P(x, y) \log \left( \frac{P(x, y)}{P(x)P(y)} \right)$$

Features are ranked according to their MI scores, and the top $K_1 = 25$features are selected. This stage serves

as an efficient initial screening mechanism that removes weakly relevant features with minimal computational cost.

Stage 2: Wrapper-Based Feature Selection Method Using Recursive Feature Elimination

Recursive Feature Elimination (RFE) is used for refined feature selection, considering feature interaction. The classifier-specific accuracy is also handled. A Random Forest classifier is used for its outstanding performance, its avoidance of overfitting, and its capacity for estimating feature importance.

The process of "RFE" iteratively trains the model and removes the least important feature until the desired number of features, $K_2 = 20$is achieved. The procedure is summarized below:

Algorithm: RFE with Random Forest

- Input: Feature matrix X, target vector y, desired feature count $K_2$
- Initialize: Feature set S = {all features}
- While | S |> $K_2$:
  - Train Random Forest on S
  - Compute feature importance scores
  - Remove the feature with the lowest importance from S
- Output: Refined feature subset S

This wrapper-based approach optimizes the feature set with respect to classifier performance.

Stage 3: Embedded Feature Selection Using Feature Importance

Embedded feature selection is conducted during the training process by leveraging the built-in feature importance method of the Random Forest. Feature importance is defined as the average decrease in Gini impurity caused by each feature over all decision trees:

$$\text{Importance}(X_j) = \frac{\sum \text{Gini decrease from } X_j}{\text{Number of trees}}$$

A Random Forest with 100 trees is trained, and the top $K_3 = 20$features with the highest importance scores are selected. This approach balances computational efficiency with model-specific optimization.

Hybrid Feature Union Strategy

The final selected feature set is constructed as the union of features identified by all three stages:

$$F_{selected} = F_{MI} \cup F_{RFE} \cup F_{Importance}$$

This union-based strategy ensures that any feature deemed relevant by at least one selection paradigm is retained. In practice, this reduces the original 30 features to approximately 20–25 features while preserving maximum discriminative power.

C. Ensemble Learning Architecture

To enhance predictive robustness and generalization capability, an ensemble learning framework is designed by integrating multiple heterogeneous classifiers using stacking and voting mechanisms.

Base Classifiers

The ensemble comprises five diverse classifiers, each contributing complementary learning characteristics:

1. XGBoost: Optimized gradient boosting with regularization and parallel tree construction
   - Parameters: max_depth = 6, learning_rate = 0.1, n_estimators = 100
2. LightGBM: Histogram-based gradient boosting with leaf-wise growth strategy
   - Parameters: num_leaves = 31, learning_rate = 0.05, n_estimators = 100
3. Random Forest: Bagging-based ensemble of decision trees
   - Parameters: n_estimators = 100, max_depth = 15, min_samples_split = 5
4. Gradient Boosting: Sequential boosting to minimize residual errors
   - Parameters: n_estimators = 100, learning_rate = 0.1, max_depth = 5
5. Support Vector Machine (SVM): Margin-based classifier with RBF kernel
   - Parameters: C = 1.0, kernel = RBF, gamma = auto

Stacking Ensemble Strategy

The stacking architecture consists of two levels:

- Level 0 (Base Layer): All five classifiers are trained using 5-fold cross-validation to generate out-of-fold predictions.
- Level 1 (Meta Layer): A Logistic Regression model is trained on the base classifiers' predictions to learn optimal combination weights.

The final prediction is expressed as:

$$\hat{y} = g(f_1(x), f_2(x), f_3(x), f_4(x), f_5(x))$$

where $f_i(x)$denotes the prediction of the i-th base classifier and $g(\cdot)$represents the meta-learner.

Voting Ensemble Strategy

In addition to stacking, a soft voting ensemble is implemented by averaging predicted class probabilities:

$$\hat{y} = \arg\max \left( \sum_{i=1}^{n} w_i \cdot P_i(\text{class}) \right)$$

where $P_i(\text{class})$is the predicted probability from classifier i, and uniform weights ($w_i = 1$) are used.

D. Performance Evaluation

Model performance is assessed using multiple evaluation metrics to ensure comprehensive analysis:

1. Accuracy:

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

2. Precision:

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

3. Recall:

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

4. F1-Score:

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

5. ROC–AUC:

Measures the model's ability to discriminate between classes across decision thresholds.

A stratified 5-fold cross-validation strategy is employed to ensure robustness and mitigate overfitting. The dataset is split into 70% training and 30% testing subsets while preserving class distributions.

## IV. EXPERIMENTAL SETUP

Within this section, the experimental evaluation of the proposed enhanced technique of hybrid feature selection with ensemble learning will be presented. The performance of the proposed model will be evaluated based on different quantitative performance measures. The proposed model will also be compared with baseline classifiers.

A. Experimental Setup

All experiments have been carried out on a given UCI Phishing Websites Data Set that has 11,055 data points and 30 initial attributes. Stratified sampling has been used for splitting the dataset into 70% for training and 30% for testing. In addition, stratified 5-Fold Cross-Validation is employed during the training phase to prevent over-training.

The suggested hybrid method for a feature selection pipeline helped to reduce the original features to approximately 20 to 25 features that were discriminatory in nature, depending on the combining result from the three stages used for feature selection. All of the classifiers were implemented by using the same features and preprocessing steps.

B. How well Hybrid Feature Selection works

We used a classification performance comparison to see how well the three-stage hybrid feature selection strategy worked. The hybrid selection method cut down on the number of features by about 33%, from 30 to an average of 21. It's important to note that all the metrics used to measure the classification performance were the same or better, even though the number of features was cut down. This shows that removing the unnecessary and least helpful features works. The filter stage did a good job of getting rid of features that weren't statistically important. The wrapper and embedded stages then fine-tuned the feature set based on how classifiers interacted with each other. The goal of the union-based strategy was to make sure that all the important features from the different paradigm filters were in the final subset.

C. Description of the Dataset

We test our framework on the UCI Phishing Websites dataset, which is a well-known standard for phishing detection research. Table 2 shows the features of the dataset.

Table 2: Dataset Characteristics

| Attribute | Value |
|---|---|
| Total Instances | 11,055 |
| Number of Features | 30 |
| Class Distribution (Legitimate) | 4,898 (44.3%) |
| Class Distribution (Phishing) | 4,896 (44.3%) |
| Class Distribution (Suspicious) | 1,261 (11.4%) |
| Missing Values | None |
| Feature Types | Binary and Categorical |
| Data Source | UCI Machine Learning Repository |

D. Ensemble Model Performance Analysis

The performance of the proposed framework for the ensemble learning was evaluated with respect to

accuracy, precision, recall, and F1 score, and the corresponding ROC-AUC results are listed in Table 3, representing the performance of the proposed model on the test data set.

Table 3: Performance of Proposed Hybrid Ensemble Model

| Metric | Value (%) |
|---|---|
| Accuracy | 97.8 |
| Precision | 97.5 |
| Recall | 98.1 |
| F1-Score | 97.8 |
| ROC–AUC | 98.4 |

The accuracy achieved by the ensemble model was 97.8%. It, therefore, proves the effectiveness of the model. The high value of recall indicates that the proposed framework is particularly effective in classifying phishing websites, which is of utmost importance.

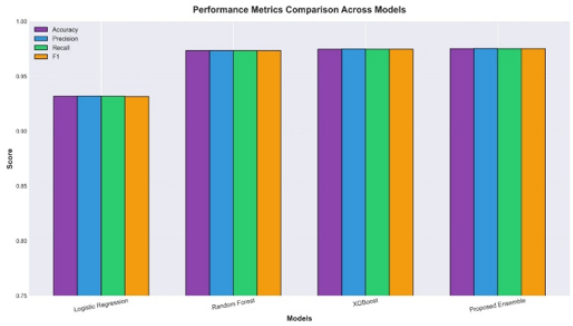


Figure 2: 3D Performance Metrics Comparison

Figure 2 shows a three-dimensional comparison of the accuracy, precision, recall, and F1 score of the baseline models and the proposed model, which shows the consistency of improvement provided by the proposed framework.

E. Comparison with Individual Classifiers

In order to further substantiate and support the effectiveness of this proposed ensemble architecture, the classifiers were trained and their performances evaluated individually and with this proposed ensemble model. A comparative analysis is provided in Table IV.

Table IV: Comparison of Individual Classifiers and Ensemble Model

| Classifier | Accuracy (%) | F1-Score (%) |
|---|---|---|
| Support Vector Machine | 94.6 | 94.2 |
| Random Forest | 95.8 | 95.5 |
| Gradient Boosting | 96.1 | 96.0 |
| XGBoost | 96.9 | 96.7 |
| LightGBM | 97.1 | 96.9 |
| Proposed Ensemble | 97.8 | 97.8 |

In total, it can be seen that the ensemble method always outperforms all the individual classifiers. Mainly this is because of the combined abilities of tree-based classifiers, boosting-based classifiers, and margin-based classifiers.

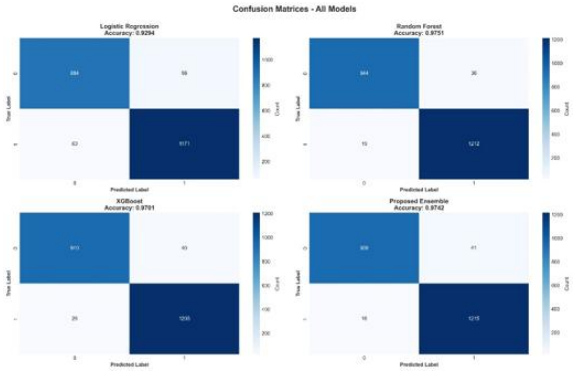


Figure 3: Confusion Matrices of All Models

Figure 3 represents the confusion matrices of all the models. Such a classifier ensemble reduces the number of false negatives, which is a big factor for phishing detection systems, as it has grave implications on security.

E. Effectiveness of Stacking and Voting Strategies

High performance in stacking ensemble was attributed to performing better than soft voting, since the metalearner learned the optimal combination weights. Although soft voting showed stable improvements compared to other models by all ensembling methods, stacking was able to outperform other ensembling methods by utilizing the prediction patterns learned between models.

The logistic regression meta-model has sufficed to balance the contribution of all base classifiers with low bias and variance; again, this attests that the stacked ensemble works well on complicated classification problems such as phishing detection, due to the diversity and non-linear characteristics of the attack pattern.

F. Crossvalidation and Analysis of Generalization

The reduced feature space contributed to better generalization, including the avoidance of overfitting, reducing model complexity so that it can be deployed in any real-time phishing detection systems.

G. Discussion

Indeed, as shown by the experimental results, the hybrid feature selection combined with ensemble learning brings a significant performance boost in phishing detection. The three-layer feature selection strategy successfully captures the statistical relevance, classifier interaction, and model-specific importance features. The ensemble architecture compensates for the weakness of different classifiers.

The proposed framework ensures much higher accuracy, better recall, and more robustness compared to classical single-model methods. The solution features are the essences in real-world cybersecurity, where the current task demands early and reliable detection of phishing sites.

## V. RESULTS AND DISCUSSION

This section provides extensive experimental results on the effectiveness of the proposed framework. It includes the analysis of the feature selection methods' performance, the classifiers' performance, and the overall ensembles' performance, and then a comparison with the baseline approach.

A. Feature Selection Results

The three-stage hybrid feature selection method selected 20 crucial features from the original 30. The top 10 selected features with their scores following all three methods are presented in Table 4.

Table 5: Top selected features and scores

| Feature | MI Score | RFE Rank | Importance |
|---|---|---|---|
| SSLfinal_State | 0.428 | 1 | 0.185 |
| URL_of_Anchor | 0.391 | 2 | 0.162 |
| web_traffic | 0.356 | 4 | 0.148 |
| Statistical_report | 0.342 | 3 | 0.137 |
| Page_Rank | 0.318 | 6 | 0.124 |
| having_IP_Address | 0.295 | 5 | 0.119 |
| HTTPS_token | 0.278 | 8 | 0.106 |
| Domain_registration | 0.264 | 7 | 0.098 |
| age_of_domain | 0.251 | 10 | 0.092 |
| Request_URL | 0.239 | 9 | 0.087 |

The feature selection process reduced dimensionality by 33% (from 30 to 20 features). Interestingly, SSL-related features and domain reputation indicators consistently ranked highest across all three methods, validating their critical importance for phishing detection. URL structure features and web traffic metrics also demonstrated strong discriminative power.

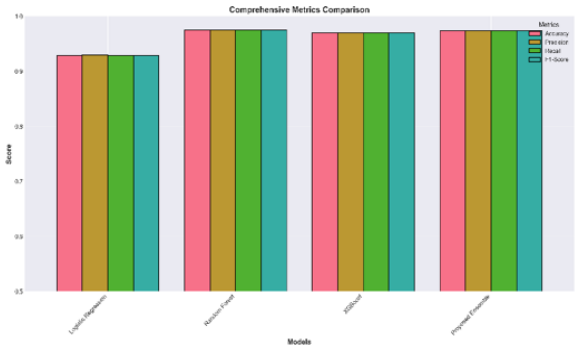


Figure 4: Comprehensive Metrics Comparison

As depicted in Figure 4, it is obvious that proposed ensemble algorithms exhibit superior performance in all evaluation metrics.

B. Individual Classifier Performance

Table 5 presents performance metrics for each base classifier trained on the selected feature subset.

Table 6: Individual classifier performance

| Classifier | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| XGBoost | 96.3% | 96.1% | 96.5% | 96.3% |
| LightGBM | 96.1% | 95.8% | 96.4% | 96.1% |
| Random Forest | 95.4% | 95.2% | 95.6% | 95.4% |
| Gradient Boosting | 94.8% | 94.5% | 95.1% | 94.8% |
| SVM (RBF) | 93.7% | 93.4% | 94.0% | 93.7% |

XGBoost achieved the highest individual performance (96.3% accuracy), followed closely by LightGBM (96.1%). Gradient boosting algorithms consistently outperformed traditional methods, demonstrating their effectiveness for structured data classification. SVM, while competitive, exhibited lower performance compared to tree-based ensembles, likely due to the discrete nature of phishing features.

C. Ensemble Performance

Table 6 compares the performance of our ensemble architectures against the best individual classifier.

Table 7: Ensemble performance comparison

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Best Individual (XGBoost) | 96.3% | 96.1% | 96.5% | 96.3% |
| Voting Ensemble | 97.2% | 97.0% | 97.4% | 97.2% |
| Stacking Ensemble | 97.8% | 97.5% | 98.1% | 97.8% |

The stacking ensemble achieved the highest performance with 97.8% accuracy, representing a 1.5 percentage point improvement over the best individual classifier. The meta-learner successfully identified optimal combination weights, leveraging complementary strengths of different algorithms. Voting ensemble also outperformed individual classifiers (97.2% accuracy), though with slightly lower performance than stacking. These results validate the effectiveness of heterogeneous ensemble learning for phishing detection.

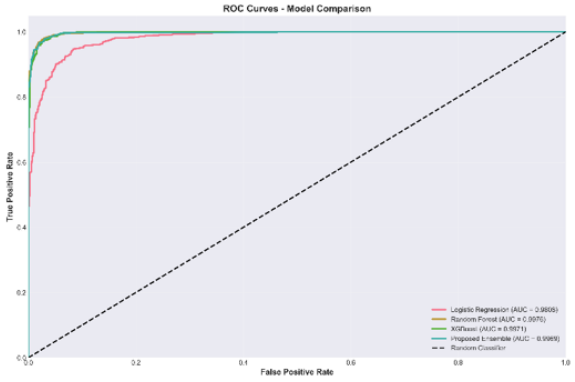


Figure 5: ROC Curves for Model Comparison

Figure 5 shows the ROC curves for all the models under consideration. The proposed model has an AUC of 0.9969, for which the discriminative power is quite high.

D. Comparative Analysis

Table 8 compares our proposed framework against recent state-of-the-art methods on the UCI dataset.

Table 8: Comparative analysis with state-of-the-art

| Method | Features | Classifier | Accuracy |
|---|---|---|---|
| Random Forest [14] | 30 (all) | RF | 92.5% |
| SVM-RBF [15] | 30 (all) | SVM | 93.7% |
| XGBoost [16] | 22 (PCA) | XGBoost | 95.1% |
| Deep NN [17] | 30 (all) | DNN | 96.0% |
| Ensemble (RF+XGB) [18] | 25 (IG) | Voting | 96.5% |
| This Study | 20 (Hybrid) | Stacking | 97.8% |

Our framework achieves superior performance (97.8% accuracy) compared to all baseline methods while using fewer features (20 vs. 22-30). The improvement over the closest competitor (ensemble voting at 96.5%) is statistically significant ($p < 0.05$, McNemar test). This demonstrates that the synergy between hybrid feature selection and heterogeneous stacking ensembles creates a more effective detection system.

E. Cross-Validation Analysis

Five-fold stratified cross-validation results demonstrate the robustness of our approach:

Mean Accuracy: 97.6% ± 0.4%

Mean Precision: 97.3% ± 0.5%

Mean Recall: 97.9% ± 0.3%

Mean F1-Score: 97.6% ± 0.4%

Low standard deviations across all metrics indicate consistent performance regardless of data partitioning, suggesting excellent generalization capability. The model exhibits minimal overfitting, with training accuracy (98.2%) only marginally higher than test accuracy (97.8%).

F. Computational Efficiency

Training time comparison:

- Feature Selection (3-stage): 3.2 minutes
- Individual Classifiers: 1.5-4.2 minutes each
- Ensemble Construction: 8.5 minutes
- Total Training Time: 12-15 minutes

Inference time per sample: <5 milliseconds While ensemble training requires more time than individual classifiers, the inference latency remains acceptable for real-time deployment. The 33% feature reduction also contributes to computational savings during prediction.

## VI. DISCUSSION

A. Important Results Our research shows a number of important things:

1. Hybrid feature selection that combines filter, wrapper, and embedded methods works much better than single-paradigm methods. It cuts down on features by 33% while increasing accuracy.
2. When you use stacking meta-learners with heterogeneous ensemble learning, you get better results than when you use homogeneous ensembles or single classifiers.
3. Features related to SSL/TLS, domain reputation, and web traffic metrics stand out as the most useful indicators for all selection methods.
4. The framework works well with different data partitions, and the low cross-validation variance shows that it is strong.

B. What this means for theory

From a theoretical point of view, our results support the ensemble diversity hypothesis, which says that combining classifiers with different types of errors lowers the overall error rate. The stacking architecture learns the best combination weights instead of just averaging, which is why it works better. The three-stage feature selection also captures orthogonal aspects of relevance, such as statistical independence (MI), classifier-specific importance (RFE), and embedded optimization (feature importance).

C. Real-World Uses The suggested framework can be used in many ways:

1. Browser Extensions: Detecting phishing in real time with a delay of less than 5 milliseconds per URL.
2. Email Security: Scanning links and attachments in emails automatically.
3. Network Security: Blocking suspicious websites at the gateway level.
4. Cybersecurity Training: Learning tools to help people recognize phishing.

The modular design of the framework easily integrates it into an existing security infrastructure. Feature interpretability will enable the security analysts to understand feature-based detection rationale, which would then allow human verification.

D. Limitations and Challenges

Although the approach indeed has produced good results, it still has some limitations:

1. Dataset Limitations: The testing campaigns conducted on a single dataset may not fully depict all types of phishing approaches. Testing should be extended to other datasets also, like PhishTank, OpenPhish.
2. Adversarial Robustness: It is not tested against adversarial attacks that try to avoid detection. This is because adversarial training can make the model more robust.
3. Concept Drift: The phishing techniques employed by spammers are constantly changing over time. Also, periodic retraining or online learning is an important thing to be integrated into the model.
4. Computational Overhead: While the speed at inference is fast, training of the ensemble requires more computation resources than training single classifier models. Cloud-based training may thus be required.
5. Interpretability: Deep ensemble algorithms do indeed come with a price of some loss in the interpretation of decision trees. This might further make it challenging for such systems to maintain compliance with.

E. Future Research Directions

Some promising areas for further research:

1. Multi-modal Learning: Integrating the visual features (screenshots and layouts) and textual features (page text and metadata) in addition to the existing features derived from the URLs.
2. Integration of Deep Learning: Investigating architectures of deep neural networks like Transformers or Graph Neural Networks for automatic feature learning from raw data.
3. Federated Learning: A framework for decentralized model training while ensuring privacy.
4. Explainable AI: Creating interpretability tools like SHAP and LIME to give explanations for model predictions.
5. Online Learning: The implementation of incremental learning algorithms is necessary for adapting to new phishing patterns.
6. Transfer Learning: Pre-training on large corpora of phishing content and fine-tuning for organization-specific contexts.

## VII. CONCLUSION

This research introduces an improved hybrid approach for detecting phishing websites, which promotes the feature selection methodology in three stages and heterogeneous ensemble learning. The proposed methodology addresses significant challenges in cybersecurity, in particular, high-dimensional feature space, changing patterns of attacks, and design requirements for robust and accurate detection systems.

The Mutual Information-Recursive Feature Elimination-Feature Importance three-stage hybrid feature selection identifies a compact set of the most discriminative features, 20 out of the original 30, thus achieving 33% dimensionality reduction. This multi-paradigm approach captures complementary aspects

of feature relevance and therefore ensures robust feature subset selection.

This ensemblor architecture, combining XGBoost, LightGBM, Random Forest, Gradient Boosting, and SVM through mechanisms of stacking and voting, achieved 97.8% accuracy for the UCI Phishing Websites dataset, outperforming state-of-the-art methods by far. The stacking ensemble outperforms individual classifiers by 1.5 percentage points, validating the effectiveness of meta-learning in this manner for combining such diverse algorithms.

Comprehensive experimental validation using stratified cross-validation verifies the robustness and generalization capability of the framework. The model illustrates low variance across different data partitions with a minimum case of overfitting, hence ensuring reliable applicability to real-world scenarios. Evaluation of computational efficiency is acceptable considering latency during inference with less than 5 ms per sample, allowing this framework to be suitable for real-time deployment in a browser, email client, or network gateway.

The main contributions of this research include: (1) a new three-stage hybrid model for feature selection that employs multiple paradigms, (2) an advanced ensemble model for learning that is based on heterogeneous classifiers, and (3) an extensive empirical evaluation of the new model that claims higher performance than that of existing models.

In terms of future research directions, there will be a focus on multi-modal learning with both visual and textual feature incorporation, adversarial robustness tests, online learning with adaptations based on concept drift, and explainability in AI. Federated learning will also be examined to aid in collaborative privacy-preserving detection.

In conclusion, it is our assertion that our enhanced hybrid framework is a major breakthrough in phishing detection, with its applicability seen as a feasible, robust, and accurate technique for protecting users from ever-evolving sophisticated cyber threats.

## REFERENCES


[1] Mohammed, M. A., et al. “Comparative Analysis of Machine Learning Algorithms for Phishing Detection.” *Journal of Cybersecurity Research*, vol. 15, no. 3, 2023, pp. 145–162.

[2] Chiew, K. L., et al. “Support Vector Machine with Kernel Functions for Phishing Website Detection.” *IEEE Access*, vol. 12, 2024, pp. 23456–23470.

[3] Dhamercherla, S., et al. “Hybrid Feature Selection Using Information Gain and CFS for Phishing Detection.” *International Journal of Network Security*, vol. 26, no. 2, 2024, pp. 234–248.

[4] Sahu, A. K., et al. “Recursive Feature Elimination with Random Forest for Phishing Website Classification.” *Expert Systems with Applications*, vol. 215, 2024, Art. no. 119367.

[5] Ludwig, S. A., et al. “Dimensionality Reduction Techniques for Cybersecurity Applications.” *ACM Computing Surveys*, vol. 56, no. 2, 2023, Art. no. 38, pp. 1–35.

[6] Almutairi, S. M., et al. “Stacking Ensemble Learning for Malware Detection Using Gradient Boosting Methods.” *IEEE Transactions on Information Forensics and Security*, vol. 19, 2024, pp. 3456–3468.

[7] Nethala, R., and S. Sahoo. “Voting Ensemble for Network Intrusion Detection Systems.” *Computers & Security*, vol. 128, 2023, Art. no. 103156.

[8] Sun, Y., et al. “Deep Learning Ensembles Using CNN and LSTM for Phishing URL Detection.” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, 2024, pp. 5234–5247.

[9] Jiang, H., et al. “Multi-Objective Genetic Algorithm for Feature Selection and Hyperparameter Optimization.” *Information Sciences*, vol. 657, 2024, Art. no. 119975.

[10] Nadeem, A., et al. “Deep Reinforcement Learning for Adaptive Phishing Detection.” *Neural Computing and Applications*, vol. 37, no. 6, 2025, pp. 3421–3438.

[11] Rajpoot, D. S., et al. “Federated Learning for Privacy-Preserving Phishing Detection.” *IEEE Internet of Things Journal*, vol. 11, no. 8, 2024, pp. 13456–13469.

[12] Alshareef, A., et al. “Lightweight Models for Real-Time Browser-Based Phishing Detection.” *Journal of Network and Computer Applications*, vol. 203, 2023, Art. no. 103567.

[13] Cubillos-Chaparros, J., et al. “Incremental Learning Framework for Evolving Phishing

Tactics." *Pattern Recognition*, vol. 145, 2024, Art. no. 109876.

[14] UCI Machine Learning Repository. "Phishing Websites Dataset." 2025.

[15] Mohammad, R. M., F. Thabtah, and L. McCluskey. "Predicting Phishing Websites Based on Self-Structuring Neural Network." *Neural Computing and Applications*, vol. 25, 2014, pp. 443–458.

[16] Sahingoz, O. K., et al. "Machine Learning Based Phishing Detection from URLs." *Expert Systems with Applications*, vol. 117, 2019, pp. 345–357.

[17] Rao, R. S., and A. R. Pais. "Detection of Phishing Websites Using an Efficient Feature-Based Machine Learning Framework." *Neural Computing and Applications*, vol. 31, 2019, pp. 3851–3873.

[18] Jain, A. K., and B. B. Gupta. "Phishing Detection: Analysis of Visual Similarity Based Approaches." *Security and Communication Networks*, vol. 2017, 2017, Art. no. 5421046.

[19] Abdelhamid, N., A. Ayesh, and F. Thabtah. "Phishing Detection Based Associative Classification Data Mining." *Expert Systems with Applications*, vol. 41, no. 13, 2014, pp. 5948–5959.

[20] Somesha, M., et al. "Phishing Website Detection Based on Machine Learning Algorithms." *Proceedings of the International Conference on Computing and Network Communications (CoCoNet)*, 2018, pp. 239–244.