International Journal of Advances in Applied Sciences (IJAAS)
Vol. 15, No. 3, September 2026, pp. 1131~1146
ISSN: 2252-8814, DOI: 10.11591/ijaas.v15.i3.pp1131-1146

Optimization-enabled machine learning with feature selection for phishing detection system

Lukman Adebayo Ogundele[1], Julius Temitayo Adepoju[2], Femi Emmanuel Ayo[3], Idayat Abike Akano[3], Abidemi Emmanuel Adeniyi[4], Halleluyah Oluwatobi Aworinde[5], Oluwasegun Julius Aroba[6,7], Gbohunmi Ajibesin[8]

[1]Department of Computer Science, Faculty of Science, Olabisi Onabanjo University, Ago-Iwoye, Nigeria
[2]Department of Mathematical Science, Faculty of Physical Science, University of Ilorin, Ilorin, Nigeria
[3]Department of Computer Science, Faculty of Computing, University of Ilesa, Ilesa, Nigeria
[4]Department of Information Systems, Faculty of Computing Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria
[5]Department of Information Technology, Faculty of Accounting and Informatics, Durban University of Technology, Durban, South Africa
[6]Department of Operations and Quality Management, Faculty of Management Sciences, Durban University of Technology, Durban, South Africa
[7]Centre for Ecological Intelligence, Faculty of Engineering and Built Environment, University of Johannesburg, Johannesburg, South Africa
[8]NHS Education for Scotland, Edinburgh, United Kingdom

Article Info

Article history:

Received Jun 14, 2025
Revised Jul 29, 2026
Accepted Aug 13, 2026

Keywords:

Adaptive
Cybercrime
Cybersecurity
Feature selection
Machine learning
Phishing

ABSTRACT

The rapid evolution of phishing methods has long been a threat to cybersecurity that demands adaptive and intelligent detection techniques. This current work envisions an improved phishing detection system that involves the integration of machine learning (ML) with feature selection for improving the performance of real-time classification. A comprehensive dataset was optimized and preprocessed through Chi-square (CHI) feature selection to reduce dimensionality while preserving the most discriminative features. Random forest (RF) algorithm, selected for its robustness, interpretability, and high computational efficiency for binary classification, was employed and achieved an accuracy of 96.25%, effectively identifying sophisticated phishing attacks. The architecture is designed to operate on large datasets with minimal computational overhead but with high precision and scalability. Experimental results also indicate the dependability of the system with an area under the curve (AUC)-receiver operating characteristic (ROC) score of 1.0, indicating perfect separation of phishing from normal cases. In addition, an adaptive classification model was used to accommodate unknown data sets with a 100% accuracy score. The results above identify the potential of adaptive, feature-optimized ML systems in the ability to handle dynamic phishing attacks and enable proactive cybersecurity defense.

Corresponding Author:

Abidemi Emmanuel Adeniyi
Department of Information Systems, Faculty of Computing Science and Engineering
Obafemi Awolowo University
Ile-Ife, Nigeria
Email: aeadeniyi@oauife.edu.ng

1. INTRODUCTION

The cybersecurity threat environment has become more complicated in recent years, and one of the most prevalent and destructive types of cybercrime is phishing. The effort to steal private information, including passwords, usernames, and financial information, by posing as a reliable source in electronic communications is known as phishing. To take advantage of both technological and human shortcomings,

attackers employ a variety of deception techniques to trick people into unintentionally disclosing private information. As a popular entry point for a variety of cyber dangers, phishing was found to be responsible for over 30% of all breaches, according to the Verizon Data Breach Investigations Report (2021) [1]–[3].

Phishing detection methods traditionally use rule-based techniques to identify known indicators of phishing messages. However, these systems are static and ineffective against evolving phishing tactics. Phishing attackers can easily bypass rule-based defenses by creating emails mimicking legitimate communications. This leads to a growing demand for more advanced, intelligent, and adaptive detection systems. One solution is machine learning (ML), which can recognize intricate patterns that human analyst would not immediately recognize. Large datasets can be used to train ML models, which can evaluate a wide range of parameters to differentiate authentic communications from phishing efforts. Research has shown that supervised learning models, such as support vector machine (SVM), decision tree (DT), and neural network (NN), can detect phishing emails with high accuracy [4]–[8].

Phishing remains the most prevalent vector for cyberattacks on individuals, organizations, and critical infrastructure. Its effectiveness is due to the ability of attackers to rapidly adapt their approach, bypass traditional filters, and exploit human and system vulnerabilities. With more organizations depending on cloud computing, remote work, and decentralized digital infrastructure, traditional filter-based methods for phishing attack detection have never been in greater demand.

The proposed optimization-aided ML platform directly addresses this need by providing a deployable framework that achieves an accuracy-efficiency tradeoff. In enterprise environments, the platform can be installed in cloud-based secure email gateways for bulk filtering of incoming and outgoing messages with negligible latency. At the endpoint, it can be embedded as a plugin or browser extension and offer real-time protection against malicious uniform resource locators (URLs) and phishing sites when the user interacts. The lightweight feature selection phase is attained to guarantee that models can also be used in environments where resources are limited, such as on mobile devices or internet of things gateways, to allow decentralized enforcement of security. Threat mitigation perspective, flexibility is most important in the system. Through feature selection and optimization, the model resists overfitting to fixed datasets and dynamically retrains on new samples of phishing, always staying strong against evolving attack vectors. Its integrability with threat intelligence platforms (TIPs) also enhances collective defense: the model can process real-time indicators of compromise (IoCs) while contributing back newly learned phishing signatures to the overall security ecosystem.

The implementation of ML in real-time phishing detection faces challenges such as high accuracy and low latency, particularly when dealing with large-scale data. Researchers are exploring advanced ML techniques like deep learning and ensemble learning to enhance resilience against evasive phishing strategies. The financial and reputational impacts of phishing attacks are significant, as they often serve as entry points for more sophisticated cyber threats. As phishing tactics evolve, there is a need for detection systems that can identify known attacks and detect novel and emerging forms in real time. This study aims to develop an ML-based phishing detection system that integrates cybersecurity vulnerability frameworks to enhance its adaptive capabilities [9]–[12]. The novelty of the proposed work lies in combining optimization-based feature selection with an adaptive ML model, which collectively enhances precision, reduces computational load, and enables real-time phishing attack identification. Unlike current solutions that solve either ensemble classification or feature reduction individually, our approach combines the two together under one framework.

This study develops an adaptive phishing detection system with a strong focus on Chi-square (CHI) feature selection to enhance accuracy and efficiency. By applying the CHI method, the system selects the most relevant features, reducing computational overhead while improving classification performance. This optimized feature selection, combined with ML, ensures real-time phishing detection with higher precision and resilience against evasive tactics. Compared to traditional rule-based or standalone ML approaches, this method enhances detection speed, scalability, and efficiency, contributing to a more robust and adaptive phishing detection framework.

Phishing remains a major cybersecurity threat, requiring continuous advancements in detection techniques. ML models have become essential tools for identifying phishing attempts, but their effectiveness largely depends on the quality of selected features. Optimization techniques, such as genetic algorithms (GA), reinforcement learning, and swarm intelligence, have been integrated into feature selection processes to enhance model accuracy and computational efficiency. This literature review explores recent studies on feature selection and optimization-driven phishing detection, highlighting their strengths, drawbacks, and existing research gaps.

2. RELATED WORKS

Tanimu *et al.* [13] examined various feature elimination techniques, emphasizing the importance of reducing irrelevant features in phishing detection. Their study demonstrated that feature selection enhances

accuracy, but it also introduced concerns regarding the potential removal of critical features, leading to a loss in model generalization ability. Furthermore, the research does not analyze how feature selection impacts modern architectures, such as transformer-based phishing detection models.

Kocyigit *et al.* [14] proposed a GA to optimize the selection of relevant phishing features, improving accuracy while reducing computational complexity. However, GAs often suffers from slow convergence and require careful parameter tuning. Moreover, the study primarily focused on URL-based phishing detection, leaving a gap in evaluating real-time phishing scenarios.

Saeed [15] introduced an optimization-driven method that prioritizes the most impactful features for phishing detection, outperforming conventional baseline models. Despite its effectiveness, the approach relies on high computational resources, making real-world adoption challenging. Furthermore, it does not account for adversarial phishing attacks, where attackers modify website characteristics to bypass detection systems.

Another notable feature selection technique involves fuzzy rough set theory [16]. This method effectively removes redundant features, leading to improved classification accuracy. However, fuzzy rough set models often require extensive computation, which may limit scalability for large datasets. Additionally, the study lacks a comparative analysis against deep learning-based phishing detection models, creating an area for further research.

Ensemble methods, such as the weighted ensemble model, have also been explored. Bidabadi and Wang [17] combined multiple classifiers using a feature selection mechanism, achieving superior accuracy. However, ensemble approaches generally require high computational resources, making real-time deployment difficult. Additionally, their study does not differentiate between different types of phishing attacks, such as spear phishing and URL-based phishing.

Singh *et al.* [18] applied particle swarm optimization (PSO) for feature selection in phishing detection, reporting higher accuracy. Despite its advantages, PSO models are susceptible to slow convergence when handling high-dimensional datasets. Additionally, the study does not compare PSO with other evolutionary algorithms, such as GA or ant colony optimization (ACO).

Zara *et al.* [19] investigated deep learning-based feature selection, showing improvements in phishing detection accuracy. However, deep learning models typically require extensive labelled datasets, which may not always be available. The study also does not assess the effectiveness of deep learning-based feature selection in detecting zero-day phishing attacks, which remain a critical concern in cybersecurity.

While feature selection, optimization techniques, and different advanced ML models have significantly enhanced phishing detection systems, several gaps remain in current research, as phishing attacks continue to be a major cybersecurity threat, exploiting users' trust to steal sensitive information. Various techniques have been developed to detect phishing attempts, ranging from heuristic-based approaches to ML and deep learning models. This literature review discusses key studies that propose different methods for phishing website detection. In this study propose a phishing detection model that leverages the CHI feature selection method combined with a random forest (CHI + RF) classifier to enhance detection accuracy. The CHI method effectively selects the most relevant features, reducing dimensionality while preserving critical information, and the RF algorithm ensures robust classification by aggregating multiple DTs, thereby improving overall performance and resilience against phishing attacks.

Maneriker *et al.* [20] trace the evolution of phishing URL detection from traditional blacklist and rule-based approaches, which were fast but weak in detecting new or slightly modified attacks, to classical ML models that relied on manually crafted lexical and domain features. While RF and SVM algorithms performed better, their feature engineering dependency at the expense of flexibility to obfuscate URLs was a flaw. Subsequent deep learning methods using convolutional neural networks (CNNs) and recurrent neural networks (RNNs) reduced manually engineered feature generation by directly learning from raw URLs, but failed to model long-term dependencies among URL constituents. The authors identify this shortfall and argue that transformer models, as they are capable of modelling local and global sequential patterns with attention mechanisms, offer a promising solution to the lack of hand-crafted features in phishing detection, and thus their contribution (URL transformer (URLTran)) is an improvement over the current state of work.

Joshua *et al.* [21] emphasize the shift from centralized phishing detection models, which are plagued by privacy and scalability problems, to distributed and adaptive approaches. Traditional ML and deep learning-based phishing detection approaches, although effective, rely on centralized data collection and are therefore vulnerable to data leakage and less practical in privacy-sensitive environments. The work addresses the growing applicability of federated learning (FL), enabling joint training across nodes without revealing raw data, and continual learning (CL), enabling incremental adaptation to emerging phishing patterns and avoiding catastrophic forgetting. Prior work on phishing detection focused largely on static datasets and suffered from evolving attack surfaces, emphasizing the need for adapting models in real time. The authors also mention recent advances in attention-based classifiers, which promote robustness by attending to prominent URL or content features, for improved detection accuracy under adversarial or drifting scenarios.

By integrating FL, CL, and attention mechanisms, the study positions itself in a new direction of research that addresses privacy, adaptability, and robustness simultaneously—key demands for successful real-world, large-scale phishing detection systems [22], [23].

3. METHOD

This section presents a methodological framework for developing an adaptive phishing detection system using real-time ML techniques. It aims to mitigate the dynamic nature of phishing threats by combining traditional cybersecurity frameworks with real-time ML algorithms. The system analyzes, predicts, and reacts to new phishing patterns.

3.1. Data collection and description

The phishing detection model is trained and evaluated using two distinct datasets. The phishing detection dataset serves as the primary training dataset, comprising 247,950 entries with 42 distinct features related to URLs and domains. This dataset is structured in a pandas DataFrame format, enabling efficient data manipulation and analysis [24]. On the other hand, the PhiuSiil dataset is used to test the model's adaptive function, ensuring its real-time applicability. This dataset consists of 235,795 entries with 55 distinct features, also focusing on URL and domain-related characteristics. Like the training dataset, it is structured in a pandas DataFrame format for seamless analysis [25].

3.2. Chi-Square test for feature relevance

To assess feature relevance, this study used CHI (χ^2) test, a statistical method for identifying associations between features and the phishing classification label. In this adaptive framework, CHI test helps in selecting features that most effectively capture the behavior of phishing threats, such as unusual URL lengths or domain anomalies. The CHI test statistic for each feature is calculated as (1).

$$X^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (1)$$

Where O_{ij} denotes observed frequency and E_{ij} represents the expected frequency of occurrences under the assumption of independence. This formula identifies features most associated with phishing detection, ensuring that only the most significant cybersecurity-relevant features are retained.

3.3. Proposed model: Chi-square + random forest

This study used the RF classifier, an ML model, to create an adaptive phishing detection system. The approach has distinct advantages when it comes to managing cybersecurity-related duties, especially when it comes to analyzing intricate, real-time data streams that define phishing attempts. A tree-based ensemble learning technique that is particularly good at managing non-linear patterns in phishing data is the RF classifier. The model can identify various patterns of phishing behavior because each DT in the forest is trained on a subset of features. Because of its flexibility, it is ideal for cybersecurity, an area with a wide variety of quickly changing attack vectors. For an input x, each tree $T_k(x)$ in an RF with N trees provide a prediction. The final prediction y is derived by aggregating these individual tree predictions in (2).

$$y = \frac{1}{N} \sum T_k(x) \quad (2)$$

This ensemble method enhances the robustness of phishing detection by averaging predictions across multiple DTs, making the model less susceptible to false positives and more resilient to emerging phishing tactics. RF is a powerful and versatile ensemble learning method used for both classification and regression tasks. It builds upon the concept of DT by creating a "forest" of trees and aggregating their predictions to improve accuracy and control overfitting. The key mathematical components and equations that underpin RF will delve as follows.

3.4. Random forest construction

RF is utilized as the core classifier for detecting phishing websites by analyzing URL and domain-related features. The model is trained on a labelled dataset, distinguishing phishing from non-phishing entries based on extracted characteristics. The classification process begins with initializing and training the RF model.

```
rf_model = RandomForestClassifier(random_state=42) rf_model.fit(X_train, y_train)
rf_pred = rf_model.predict(X_test)
```

The model constructs multiple DTs (T_j) trained on bootstrapped subsets of the dataset. Given a training dataset. Where x_i represents the feature set of a URL, and y_i is the corresponding label (phishing or non-phishing), each tree is trained on a sampled subset $D_b^{(j)}$ as defined in (3).

$$D = \{(x_i, y_i)\}_{i=1}^{Ny} \quad (3)$$

At each node, instead of considering all features p, a random subset of m features (m < p) is selected for splitting. This randomness enhances model generalization and reduces overfitting. The prediction for classifying phishing and non-phishing processes aggregates results from all trees in the forest, as shown in (4).

$$\hat{y} = argmax_{c\in(0,1)} \sum_{j=1}^{M} I(T_{j(x)} = c) \quad (4)$$

Where I is an indicator function that returns 1 if the j^{th} tree classifies x as class c (phishing or non-phishing), otherwise 0; M is the total number of trees; and $\hat{y}$ is the final classification label. By aggregating predictions from multiple DTs, the RF model enhances detection accuracy, reduces variance, and improves robustness against adversarial phishing techniques. This ensemble approach ensures that the system effectively classifies malicious URLs while maintaining low false positive rates, as displayed in Figure 1.

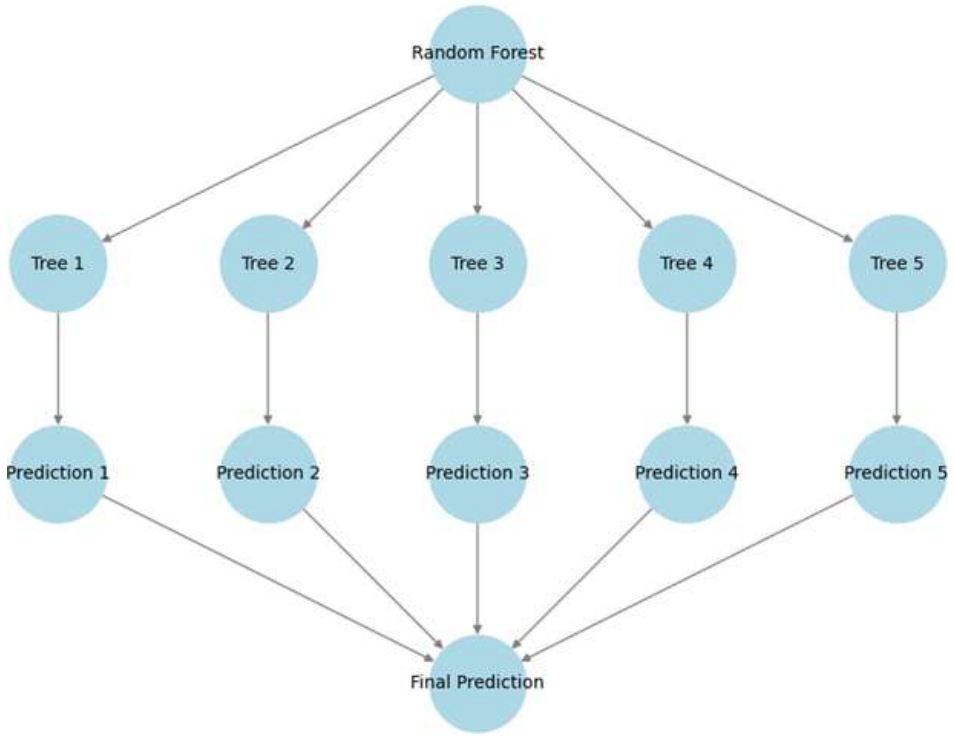

Figure 1. RF model architecture

A two-stage phishing detection pipeline, consisting of univariate feature selection and ensemble classification, is implemented in Algorithm 1. Preliminary data exploration and cleaning operations are done for input dataframe df containing n rows and m features, along with binary target label y (phishing or not). All missing entries are either filled in or zero-filled to ensure the validity of the next steps.

Feature selection is based on the CHI test. A feature selector is trained on X and y, with the retention of k features demonstrating the greatest dependence on the target label. The obtained reduced dataframe $X_{selected}$ will thus have the size of k. Distributions and correlations among selected features are presented in histograms and a heatmap correspondingly, to verify their informativeness and lack of multicollinearity.

Data is split into training and testing samples with proportions of 70:30 ($X_{train}, X_{test}, y_{train}, y_{test}$) with a fixed seed value for reproducibility purposes. Next, an RF classifier is trained on X_{train} with specified number of trees and maximal depth of each tree. After training, the RF outputs the predictions y on the hold-out testing sample. Accuracy, precision, recall, and F1-score metrics are calculated to evaluate its performance.

Distribution pie charts and frequency diagrams are made for the original and partitioned datasets. The algorithm returns the trained RF, a list of names of k selected features, and the full set of metrics. With Algorithm 1, a reduced feature space of size k and a reliable predictor in form of an ensemble model can be obtained in a compact form.

Algorithm 1. CHI + RF: phishing detection model
Require: a dataset *df* with features and target labels, top *k* features to select
Ensure: a trained RF model with selected features and evaluation metrics
1: Data inspection and preprocessing:
2: Load the dataset *df*
3: Check dataset shape, columns, and statistics (e.g., missing values, and unique values)
4: Handle missing values (e.g., fill with zero or impute values)
5: Split the df into features X and target y, where y represents phishing labels
6: Feature selection: CHI method
7: Initialize CHI selector to choose top k features
8: Fit the selector to X and Y
9: Extract and store the indices and names of the selected features
10: Reduce X to include only the selected features, forming Xselected
11: Visualize the reduced feature set using histograms and a correlation heatmap
12: Train-test split:
13: Split Xselected and y into training and testing datasets:
(Xtrain, Xtest, ytrain, ytest) = train test split(Xselected, y, test size = 0.3, random state = 42)(5)
14: RF model training:
15: Initialize the RF classifier with desired hyperparameters (e.g., number of trees, max depth)
16: Train the model using Xtrain and ytrain
17: Model evaluation:
18: Predict phishing labels ypred for Xtest
19: Compute performance metrics such as accuracy, precision, recall, and F1-score
20: Visualization and results:
21: Visualize the class distribution (e.g., legitimate vs phishing) using a pie chart
22: Plot the training and testing label distributions
23: Return trained RF model, selected features, and evaluation metrics

3.5. Evaluation metrics aligned with cybersecurity objectives

Since phishing detection in cybersecurity involves considerable risks, the system's efficacy is evaluated using a variety of metrics, including accuracy, precision, recall, F1-score, and sensitivity. Each signal displays a different aspect of the model's reliability in spotting phishing attempts.

i) Accuracy: this calculates the percentage of authentic and phishing URLs that are successfully categorized out of all URLs. Even if it's helpful, it might not be enough on its own, particularly when there are considerably more valid URLs than phishing ones, as given in (5).

$$Accuracy = \frac{(TP+TN)}{TP+TN+FP+FN} \quad (5)$$

ii) Precision: this shows the percentage of actual phishing detections among all phishing-classified URLs. By minimizing false positives, high precision lowers the possibility of mistakenly reporting valid URLs, as presented in (6).

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

iii) Recall: this indicates what percentage of real phishing URLs the algorithm accurately detects. In cybersecurity situations, high recall is essential since it guarantees that phishing URLs are identified with little error, as given in (7).

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

iv) F1-score: when assessing models for unbalanced datasets, the F1-score, which combines precision and recall, is very instructive. A high F1-score shows that the model successfully strikes a balance between false positives and false negatives, as shown in (8).

$$F1 - score = 2 \times \frac{Precision+Recall}{Precision+Recall} \quad (8)$$

v) Confusion matrix analysis: confusion matrices are generated to provide insight into model performance regarding phishing and legitimate URL classifications. Each matrix offers a detailed breakdown of true positives, false positives, true negatives, and false negatives, allowing for a nuanced understanding of model errors. Analyzing false positives is particularly important in cybersecurity to avoid flagging legitimate URLs incorrectly.

4. RESULTS AND DISCUSSION

This section presents the results of the RF model applied to the phishing detection problem. The performance of the model is evaluated using several metrics, including accuracy, precision, recall, F1-score, and computational efficiency. The experimentation for the developed phishing detection system was implemented with CHI feature selection and an RF model. The phishing dataset was loaded using the Python Jupyter Notebook.

4.1. System specifications

The system is designed to run efficiently on a machine equipped with an Intel Dual-Core processor clocked at 2.20 GHz. To support the smooth execution of processes and enhance computational efficiency, a minimum of 16 GB of RAM is required. Additionally, the system should be installed on a storage device with at least 500 GB of SSD, ensuring faster data retrieval and improved overall performance (Figure 2).

4.2. Statistical distribution of the dataset

Figure 2 shows that there are 119,409 phishing URLs in the dataset, making it 48.2% of the dataset, with 128,541 legitimate URLs, making up 51.8% of the dataset. The pie chart illustrates the division between phishing sites and good sites and displays that good site consist of 128,541 instances (51.8%), while phishing sites account for 119,409 instances (48.2%). The almost balanced division is important during training ML systems because it avoids biasing towards the larger class and allows proper performance evaluation on both classes. While the phishing attacks remain somewhat lower than the original samples, the small margin reflects the continued size of phishing attacks in the web environment and the necessity for robust detection systems capable of effectively distinguishing between the two classes.

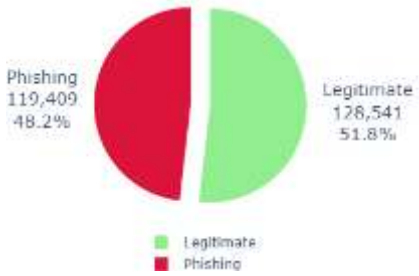

Figure 2. Distribution of the dataset

4.3. Feature selection using Chi-square

The feature selection method was used to extract the most relevant features based on CHI metrics. The CHI feature selection method reduced the number of features from 41 to 30, discarding the rest as shown in Figure 3. The features selected are shown in Table 1. This method is of the filter-based feature subset selection type, where features are ranked independent of the learning algorithm. It is computationally inexpensive, scalable for large datasets, and has good performance for categorical or discrete features (e.g., phishing factors like including "@" in a URL, or usage of an IP address). However, since χ^2 tests features in isolation, it does not take into account interactions between features, and that is why it is often complemented with embedded methods (e.g., least absolute shrinkage and selection operator or LASSO) or optimization methods in robust phishing detection systems.

```
# Apply Chi-Square feature selection
k = 30  # Number of top features to select
chi2_selector = SelectKBest(chi2, k=k)

# Fit the selector to the data
X_kbest = chi2_selector.fit_transform(X, y)

# Get the selected feature indices
selected_indices = chi2_selector.get_support(indices=True)

# Get the names of the selected features
selected_features = X.columns[selected_indices]
print("Selected features:", selected_features)
```

Figure 3. CHI feature selection to select 30 features

Table 1. Features selected using CHI

No.	Feature	No.	Feature
1	URL length	16	Number of hyphens in the domain
2	Number of dots in URL	17	Having special characters in the domain
3	Having repeated digits in URL	18	Number of special characters in the domain
4	Number of digits in URL	19	Having digits in the domain
5	Number of special characters in URL	20	Number of digits in the domain
6	Number of hyphens in URL	21	Having repeated digits in the domain
7	Number of slashes in URL	22	Number of subdomains
8	Number of question marks in URL	23	Average subdomain length
9	Number of equals in URL	24	Having digits in a subdomain
10	Number of @ in URL	25	Number of digits in the subdomain
11	Number of dollars in URL	26	Path length
12	Number of exclamation marks in URL	27	Having query
13	Number of per cent in URL	28	Having anchor
14	Domain length	29	Entropy of URL
15	Number of dots in the domain	30	Entropy of the domain

4.4. Train-test split

To separate the dataset into training and testing sets, a 70/30 split ratio was used. In particular, the model is trained using 70% of the data, which enables it to recognize and adjust to the patterns in the data. The accuracy and efficacy of the model may be assessed on unknown data by using the remaining 30% as a testing set. In order to give the model a significant quantity of training data while maintaining a suitable part for accurate performance evaluation on fresh samples, this split ratio was used. The results of the proposed model are shown in Table 2.

Table 2. CHI + RF classification report

Class	Precision	Recall	F1-score	Support
Legitimate	0.96	0.97	0.96	38569
Phishing	0.97	0.95	0.96	35816
Accuracy			0.96	74385
Macro avg	0.96	0.96	0.96	74385
Weighted avg	0.96	0.96	0.96	74385

4.5. Model performance on phishing detection

The performance of the CHI + RF model was evaluated using a test set containing an equal number of phishing and legitimate samples. The model demonstrated 96.25% accuracy, effectively distinguishing between phishing and legitimate email, minimizing. This high accuracy is essential in minimizing the risk of misclassifying phishing attacks. For precision, the model achieved 96.28%, indicating its ability to correctly identify phishing attempts while keeping false positives to a minimum. The recall was measured at 96.22%, reflecting the model's effectiveness in capturing actual phishing instances. To provide a balanced evaluation, the F1-score, which is the harmonic mean of precision and recall, was 96.22%. This metric ensures that the model maintains both high accuracy in classification and a strong ability to detect phishing attempts without producing excessive false positives.

A confusion matrix was generated to visualize the model's performance in distinguishing phishing from legitimate samples, as shown in Figure 4. This matrix is essential for understanding the distribution of false positives and false negatives, which helps guide improvements in the model. As seen in the confusion matrix, the number of false positives (legitimate emails classified as phishing) was low, indicating the robustness of the model. The false negative rate (phishing emails misclassified as legitimate) was also minimal, which is vital for preventing phishing attacks from bypassing the detection system.

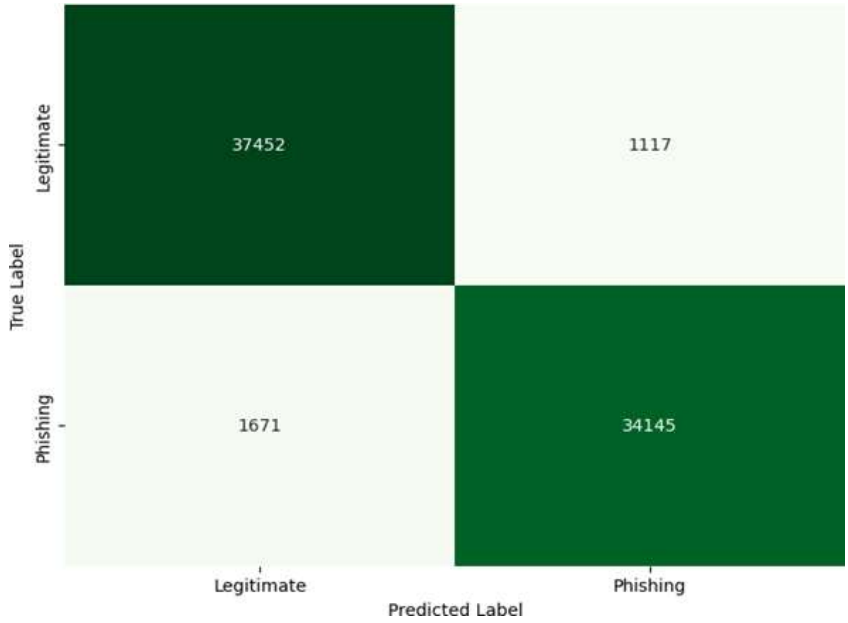

Figure 4. Confusion matrix for CHI with RF model

4.6. Comparison with other models

To evaluate the effectiveness of the proposed model, a comparison was made with other ML models commonly used in phishing detection, such as RF, naïve Bayes, and logistic regression. The results are summarized in Table 3. As seen in Table 3, the CHI + RF model outperforms other models. While bagged tree performed well, it exhibits slightly lower precision and recall than the CHI + RF model. Figure 5 displays the comparison with other models.

The CHI + RF achieves an outstanding area under the curve (AUC) score of 1.00, demonstrating perfect class separation capabilities, which suggests it is exceptionally effective at ranking predictions accurately. Despite this high AUC, its accuracy of 96.25% indicates that a small number of misclassifications occur when a specific decision threshold is applied, though it remains highly reliable overall. Comparatively, the bagged DT model follows closely with an AUC of 0.99, providing robust performance and good generalizability due to reduced variance. The multi-layer perceptron (MLP) and extreme gradient boosting (XGBoost) also perform well with AUC scores of 0.96 and 0.93, respectively, showing their strength in capturing complex patterns, albeit with potentially higher computational demands. Logistic regression, with a lower AUC of 0.87, demonstrates solid but more limited performance, as it may struggle with non-linear relationships that ensemble models handle effectively. In summary, CHI + RF stands out as the best model based on AUC, but the choice between these models could depend on specific needs such as model interpretability, computational efficiency, and the desired balance between AUC and accuracy, as shown in Figure 6.

Table 3. Performance comparison of classifiers

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Specificity (%)
CHI + RF	96.25	96.28	96.22	96.24	97.10
Bagged tree	96.01	96.04	95.98	96.01	96.81
Logistic regression	80.23	80.77	79.91	80.03	87.13
XGBoost	86.14	86.35	86.00	86.07	89.92
MLP	89.31	89.29	89.34	89.30	88.64

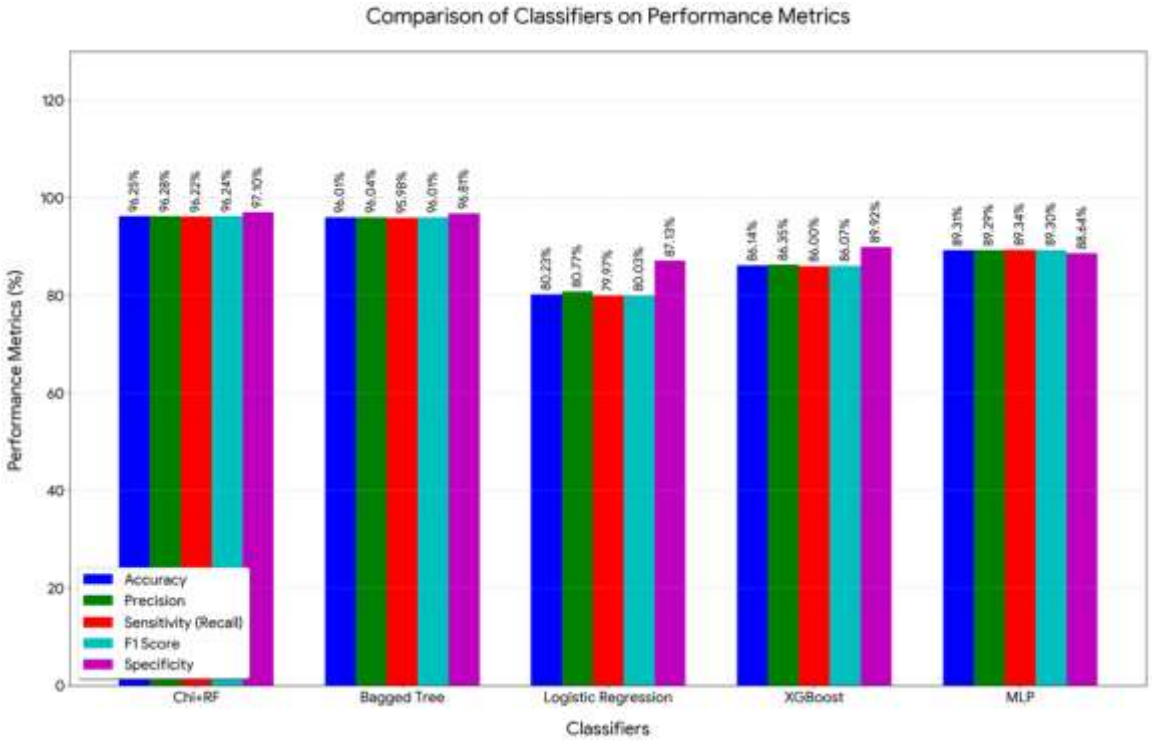

Figure 5. Bar chart comparison with other models

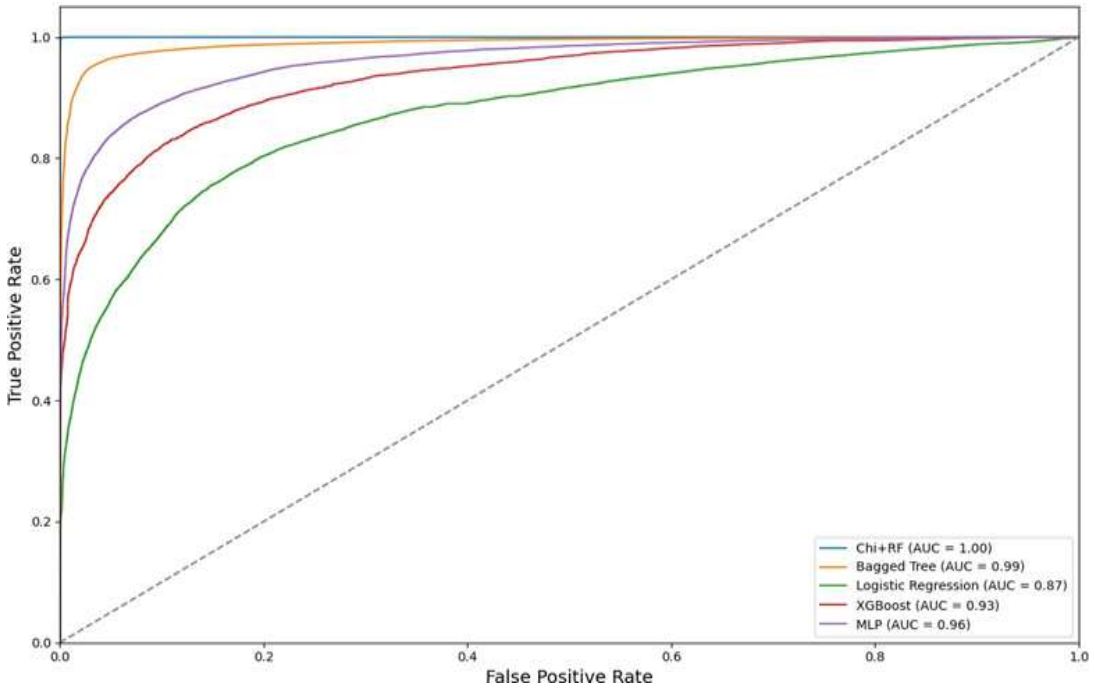

Figure 6. Receiver operating characteristic (ROC) curve of all models

4.7. Adaptive retraining function

The adaptive retraining function is designed to enhance an ML model's resilience against evolving threats, particularly in the context of phishing detection. It accomplishes this by allowing the model to incorporate new data dynamically, ensuring CL and improving its predictive capabilities. Upon receiving fresh data, the function first checks for data validity and applies a pre-existing feature selection method to streamline the input features. It then retrains the RF model using the updated dataset and computes various

performance metrics, including accuracy, precision, recall, and F1-score, to evaluate the model's effectiveness. The function also visualizes the model's performance through a confusion matrix, providing insights into its classification results. Additionally, it saves the updated model for future use, enabling seamless deployment and further adaptation to new phishing tactics. This approach not only bolsters the model's accuracy but also ensures that it remains relevant in the face of rapidly changing cyber threats. Figure 7 displays the adaptive retraining function model, while Figure 8 shows the confusion matrix for the adaptive retraining function. The classification report for the adaptive retraining function is shown in Table 4.

```
def adaptive_retraining(new_data, selector, rf_model):
    try:
        # Split new data for testing the adaptive response
        X_new = new_data.drop(columns=['Type'])
        y_new = new_data['Type']

        # Check if the new data is empty
        if X_new.empty or y_new.empty:
            raise ValueError("New data is empty. Please provide valid data for retraining.")

        target_names = ['Legitimate', 'Phishing']

        # Apply the existing feature selector to the new data
        X_new_selected = selector.transform(X_new)

        # Fit the model with new data (continual learning)
        rf_model.fit(X_new_selected, y_new)

        # Make predictions on the new data
        y_new_pred = rf_model.predict(X_new_selected)
        y_new_proba = rf_model.predict_proba(X_new_selected)[:, 1]  # Probability for 'Phishing' class
```

Figure 7. Adaptive retraining function

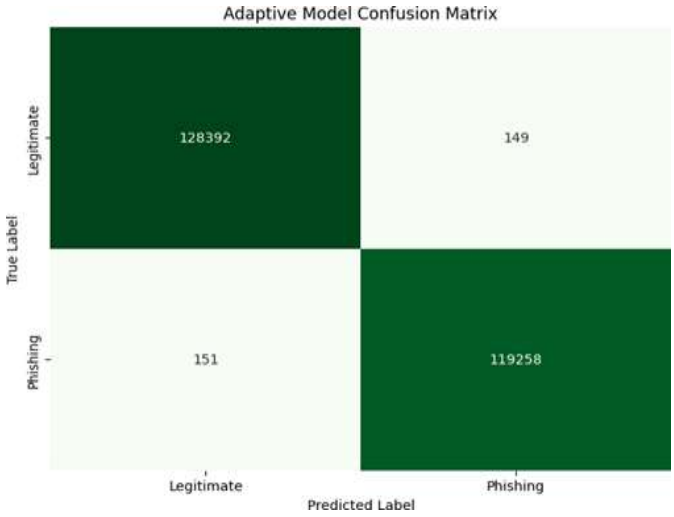

Figure 8. Confusion matrix for the adaptive retraining function used on the same dataset

Table 4. Classification report for the adaptive retraining function used on the same dataset

Class	Precision	Recall	F1-score	Support
Legitimate	1.00	1.00	1.00	128,541
Phishing	1.00	1.00	1.00	119,409
Accuracy			1.00	247,950
Macro avg	1.00	1.00	1.00	247,950
Weighted avg	1.00	1.00	1.00	247,950

The adaptive retraining function demonstrated remarkable performance when tested on the same dataset, yielding exceptionally high evaluation metrics. The adaptive model achieved an accuracy of 99.88%, indicating its effectiveness in accurately classifying both legitimate and phishing instances. Precision and recall metrics were equally impressive, both recorded at 99.88%, signifying that the model successfully identified the vast majority of phishing attempts without misclassifying legitimate emails. The F1-score, which balances precision and recall, also stood at 99.88%, reflecting the model's overall robustness in handling both classes. The classification report further revealed that both classes—legitimate and phishing—

achieved perfect scores across precision, recall, and F1 metrics, with each category receiving full support for their respective instances (128,541 legitimate and 119,409 phishing emails). These results underscore the model's exceptional capability to adapt to new data while maintaining high levels of accuracy and reliability in phishing detection. The confusion matrix of the PhiuSiil dataset is shown in Figure 9. Table 5 shows the adaptive retraining function used on the same dataset.

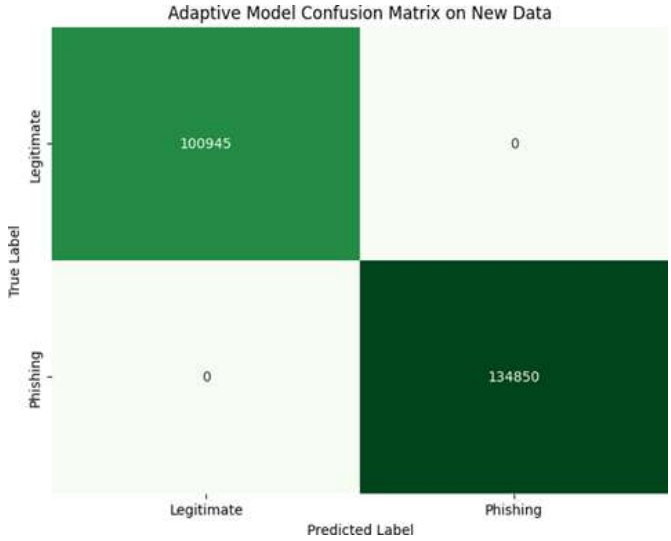

Figure 9. Confusion matrix of the PhiuSiil dataset

Table 5. Classification report for the adaptive retraining function used on the same dataset

Class	Precision	Recall	F1-score	Support
Legitimate	1.00	1.00	1.00	100,945
Phishing	1.00	1.00	1.00	134,850
Accuracy			1.00	235,795
Macro avg	1.00	1.00	1.00	235,795
Weighted avg	1.00	1.00	1.00	235,795

4.8. PhiiuSiil dataset

The adaptive phishing detection model was evaluated using the PhiiuSiil dataset with 55 columns and CHI selecting K = 10, yielding exceptional results across all performance metrics. The model achieved 100% accuracy, 100% precision, 100% recall, and an F1-score of 100%. This performance indicates that the model successfully identified all phishing and legitimate samples in the test dataset without any misclassifications. The confusion matrix shows that the model correctly classified all 100,945 legitimate cases and 134,850 phishing cases, with no false positives or false negatives.

Such outstanding results suggest that the adaptive model effectively captures and responds to characteristics of phishing and legitimate instances, maintaining consistent accuracy even with new datasets. The pipeline, including RF as the core classifier, has been deployed successfully, ensuring that it can be used for real-time applications. This level of precision reflects the model's robustness and high generalization across varying datasets. The comparison with the existing literature is shown in Table 6.

Table 6. Comparison with existing literature

Models	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CHI + RF (1st training)	96.25	96.88	96.22	96.22
CHI + RF (2nd training)	99.88	99.88	99.88	99.88
DT [26]	96.00	96.70	96.10	96.40
RF [26]	96.90	96.70	97.70	97.20
Gradient boosting [26]	98.90	99.00	99.40	98.60

4.9. Comparison with existing methods

Table 6 compares phishing detection models based on accuracy, precision, recall, and F1-score, highlighting the impact of dataset size and feature richness. This study utilized a significantly larger dataset,

comprising 247,950 entries with 42 features, whereas prior research by Sucharitha *et al.* [26] used a smaller dataset with only 11,055 URLs and 32 features. The CHI + RF model in its initial training achieved 96.25% accuracy, which already rivalled existing models, but after further training, it reached an impressive 99.88% across all metrics, demonstrating the advantage of a larger dataset and more extensive training. In contrast, previous models such as DT (96.00%), RF (96.95%), and gradient boosting (98.9%) performed well but did not surpass the refined CHI + RF model. These results suggest that increasing the dataset size and feature set significantly enhances model performance, leading to more reliable phishing detection.

5. CONCLUSION

The CHI + RF demonstrated exceptional performance in phishing detection, achieving high accuracy, precision, recall, and F1-score across multiple datasets. The strength of the approach lies in the effectiveness of CHI as a feature selection method, which ensures that only the most relevant attributes are used for classification, reducing noise and improving efficiency. The model consistently achieved a 96.25% accuracy rate on the initial dataset, outperforming other classifiers such as bagged trees, MLP, and XGBoost in both precision and recall. The adaptive retraining function further enhanced the model's robustness by allowing it to learn from new data dynamically, leading to a remarkable 99.88% accuracy on the same dataset and an unprecedented 100% accuracy on the PhiuSiil dataset. The near-perfect classification rates, as reflected in the confusion matrices, suggest that the model effectively minimizes false positives and false negatives, making it highly reliable for real-world phishing detection. Moreover, its superior AUC score of 1.00 indicates excellent class separation capability, reaffirming its dominance over other competing models. By leveraging the power of CHI for optimal feature selection and RF for classification, this model provides a scalable, adaptive, and highly effective solution for combating phishing threats in dynamic cybersecurity environments. Additional work can extend this effort in several promising directions. First, the incorporation of explainable artificial intelligence (XAI) techniques such as Shapley additive explanations (SHAP), local interpretable model-agnostic explanations (LIME), or attention visualization would promote transparency through explainable phishing alerts, building greater user trust and enabling compliance in high-risk domains such as finance and health care. Second, applying the framework to multilingual phishing detection using models such as multilingual-bidirectional encoder representations from transformers (mBERT) or cross-lingual language model with RoBERTa (XLM-R) can further enhance its use across diverse linguistic settings, responding to the global nature of phishing attacks.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Lukman Adebayo Ogundele	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Julius Temitayo Adepoju		✓				✓		✓	✓	✓	✓	✓		
Femi Emmanuel Ayo	✓		✓	✓			✓			✓	✓		✓	✓
Idayat Abike Akano		✓				✓		✓	✓	✓	✓	✓		
Abidemi Emmanuel Adeniyi	✓		✓	✓			✓			✓	✓		✓	✓
Halleluyah Oluwatobi Aworinde					✓		✓			✓		✓		✓
Oluwasegun Julius Aroba					✓		✓			✓		✓		✓
Gbohunmi Ajibesin					✓		✓			✓		✓		✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The author declares that there are no known conflicts of interest associated with this publication. There are no financial or personal relationships that could inappropriately influence or bias the content of this work.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The study did not make use of any human or animal parts to carry out the work.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study. The dataset was obtained from a publicly available repository at the PhiuSiil dataset, reference number [25].

REFERENCES

[1] I. Kumar, "Emerging threats in cybersecurity: a review article," *International Journal of Applied and Natural Sciences Journal*, vol. 1, no. 1, pp. 1–9, 2023.

[2] A. Goni, M. U. F. Jahangir, and R. R. Chowdhury, "A study on cyber security: analyzing current threats, navigating complexities, and implementing prevention strategies," *International Journal of Research and Scientific Innovation*, vol. 10, no. 12, pp. 507–522, 2024, doi: 10.51244/ijrsi.2023.1012039.

[3] M. Thakur, "Cyber security threats and countermeasures in digital age," *Journal of Applied Science and Education*, vol. 4, no. 1, pp. 1–20, 2024, doi: 10.54060/a2zjournals.jase.42.

[4] M. Nadeem, S. W. Zahra, M. N. Abbasi, A. Arshad, S. Riaz, and W. Ahmed, "Phishing attack, its detections and prevention techniques," *International Journal of Wireless Information Networks*, vol. 1, no. 2, pp. 13–25, 2023.

[5] A. A. Akinyelu, "Machine learning and nature inspired based phishing detection: a literature survey," *International Journal on Artificial Intelligence Tools*, vol. 28, no. 5, 2019, doi: 10.1142/S0218213019300023.

[6] K. Gowri and S. Brindha, "Advancements in smishing detection: a novel approach utilizing graph neural networks," *Computer Research and Development*, vol. 24, no. 7, pp. 197–212, 2024, doi: 20.2040/crd.v24.2024.4217.

[7] M. Moghimi and A. Y. Varjani, "New rule-based phishing detection method," *Expert Systems with Applications*, vol. 53, pp. 231–242, 2016, doi: 10.1016/j.eswa.2016.01.028.

[8] P. Singh, Y. P. S. Maravi, and S. Sharma, "Phishing websites detection through supervised learning networks," in *2015 International Conference on Computing and Communications Technologies*, Chennai, India, 2015, pp. 61-65, doi: 10.1109/ICCCT2.2015.7292720.

[9] A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti, and M. Alazab, "A comprehensive survey for intelligent spam email detection," *IEEE Access*, vol. 7, pp. 168261–168295, 2019, doi: 10.1109/ACCESS.2019.2954791.

[10] Y. Chen, J. Jiang, S. Sun, B. He, and M. Chen, "RUSH: real-time burst subgraph detection in dynamic graphs," *Proceedings of the VLDB Endowment*, vol. 17, no. 11, pp. 3657–3665, 2024, doi: 10.14778/3681954.3682028.

[11] A. Ashiru, "Identifying phishing as a form of cybercrime in Nigeria," *Nnamdi Azikiwe University Journal of International Law and Jurisprudence*, vol. 12, no. 2, pp. 176–186, 2021.

[12] M. O. Ibiyemi and D. O. Olutimehin, "Cybersecurity in supply chains: addressing emerging threats with strategic measures," *International Journal of Management & Entrepreneurship Research*, vol. 6, no. 6, pp. 2024–2047, 2024, doi: 10.51594/ijmer.v6i6.1241.

[13] J. Tanimu, S. Shiaeles, and M. Adda, "A comparative analysis of feature eliminator methods to improve machine learning phishing detection," *Journal of Data Science and Intelligent Systems*, vol. 2, no. 2, pp. 87–99, 2024, doi: 10.47852/bonviewJDSIS32021736.

[14] E. Kocyigit, M. Korkmaz, O. K. Sahingoz, and B. Diri, "Enhanced feature selection using genetic algorithm for machine-learning-based phishing URL detection," *Applied Sciences*, vol. 14, no. 14, 2024, doi: 10.3390/app14146081.

[15] M. M. Saeed, "Novel optimization-driven feature selection approach for enhancing phishing website detection accuracy," *Journal of Electrical Systems*, vol. 20, no. 6s, pp. 1408–1417, 2024, doi: 10.52783/jes.2923.

[16] M. Zabihimayvan and D. Doran, "Fuzzy rough set feature selection to enhance phishing attack detection," *IEEE International Conference on Fuzzy Systems*, 2019, pp. 1–6, doi: 10.1109/FUZZ-IEEE.2019.8858884.

[17] F. S. Bidabadi and S. Wang, "A new weighted ensemble model for phishing detection based on feature selection." 2022, *arXiv:2212.11125*.

[18] T. Singh, M. Kumar, and S. Kumar, "Enhancing phishing website detection using particle swarm optimization and feature selection techniques," in *2023 IEEE World Conference on Applied Intelligence and Computing, AIC 2023*, 2023, pp. 977–982, doi: 10.1109/AIC57670.2023.10263814.

[19] U. Zara, K. Ayyub, H. U. Khan, A. Daud, T. Alsahfi, and S. G. Ahmad, "Phishing website detection using deep learning models," *IEEE Access*, vol. 12, pp. 167072–167087, 2024, doi: 10.1109/ACCESS.2024.3486462.

[20] P. Maneriker, J. W. Stokes, E. G. Lazo, D. Carutasu, F. Tajaddodianfar, and A. Gururajan, "URLTran: improving phishing URL detection using transformers," in *Proceedings - IEEE Military Communications Conference MILCOM*, 2021, pp. 197–204, doi: 10.1109/MILCOM52596.2021.9653028.

[21] M. J. Joshua, R. Adhithya, S. S. Dananjay, and M. Revathi, "Exploring the efficacy of federated-continual learning nodes with attention-based classifier for robust web phishing detection: an empirical investigation," in *International Conference on Advancements in Power, Communication and Intelligent Systems, APCI 2024*, 2024, pp. 1–7, doi: 10.1109/APCI61480.2024.10617245.
[22] S. R. Atla, "Phishing URL detection using DistilBERT and capsule neural networks," M.Sc. Thesis, School of Computing, National College of Ireland, Dublin, Ireland, 2024.
[23] A. B. Sakpere, H. O. Aworinde, O. F. Afe, S. Adebayo, A. E. Adeniyi, and O. J. Aroba, "Exploring user adoption and experience of automated machine learning platforms with a focus on learning curves, usability, and design considerations," *The Open Biomedical Engineering Journal*, vol. 19, no. 1, 2025, doi: 10.2174/0118741207395767250818130213.
[24] M. Tamal, "Phishing detection dataset," *Mendeley Data*, 2023, doi: 10.17632/6tm2d6sz7p.1.
[25] A. Prasad and S. Chandra, "PhiUSIIL: a diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning," *Computers and Security*, vol. 136, Jan. 2024, doi: 10.1016/j.cose.2023.103545.
[26] B. Sucharitha, B. Chandini, D. S. Kumar, M. Surendra, and D. G. K. Kumar, "Detecting phishing websites using machine learning," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 13, no. 4, Apr. 2024, doi: 10.17148/IJARCCE.2024.134145.

BIOGRAPHIES OF AUTHORS

Lukman Adebayo Ogundele is a lecturer at Olabisi Onabanjo University, Ago-Iwoye, Ogun State. A graduate of Federal University of Technology, Minna, Niger State; Obafemi Awolowo University, Ile-Ife; and Federal University of Technology, Akure, where he bagged Ph.D. in Computer Science. He has worked in many places, among them Osun State College of Education, Ilesa. He has attended many national and international conferences and has many local and international journals in the area of intelligence, information technology, networking, and security to his credit. He can be contacted at email: ogundele_lukman@ooa.edu.ng.

Julius Temitayo Adepoju is a researcher in the Department of Mathematics with a focus on numerical analysis. He is skilled in data analysis, machine learning, and artificial intelligence. He holds a master of science degree in Mathematics (distinction) from the University of Ilorin, and a bachelor's degree in Mathematics (second class upper) from Olabisi Onabanjo University, Ogun State. His research interests center on applying advanced computational and analytical methods in data-driven problem-solving. He can be contacted at email: adepoju.jt@unilorin.edu.ng.

Femi Emmanuel Ayo is a lecturer in the Department of Computer Science, University of Ilesa, Ilesa, Nigeria. He received his bachelor of Computer Science from Olabisi Onabanjo University, Ago-Iwoye, in 2012 and his master of Computer Science from Federal University of Agriculture, Abeokuta, in 2016. He obtained his Ph.D. degree at the Federal University of Agriculture, Abeokuta, Nigeria, in 2021. He is a co-primary investigator in the Google ExploreCSR 2023, an artificial intelligence Google grant. He is a member of the International Association of Engineers (IAENG). He has published in many international and local journals. He has won many awards and grants. His primary research interests include artificial intelligence, information systems, expert systems, and information security. He can be contacted at email: femi.emmanuel@unilesa.edu.ng.

Idayat Abike Akano is an academic and researcher at the University of Ilesa, with expertise in computing and a strong interest in intelligent systems, cybersecurity, and forensic linguistics. She is actively engaged in teaching, curriculum development, and institutional quality assurance. Her work focuses on leveraging emerging technologies to enhance security, innovation, and digital inclusion. She can be contacted at email: idayat.abike@unilesa.edu.ng.

Abidemi Emmanuel Adeniyi is a lecturer/researcher in the Department of Information Systems, Obafemi Awolowo University, Ile-Ife, Nigeria and a postdoctoral research fellow at Wits Machine Intelligence and Neural Discovery (MiND) Institute, University of the Witwatersrand, Johannesburg, South Africa, Wits Games and Artificial Intelligence and Culture Lab, School of Arts, University of the Witwatersrand, Johannesburg, South Africa. He has authored or coauthored more than 80 journal and conference papers, and 80 book chapters with Elsevier and Springer. His research interests include the internet of things security, information security, applications of artificial intelligence, application of machine learning for intrusion detection, internet of medical things, and cryptographic algorithms. He can be contacted at email: aeadeniyi@oauife.edu.ng.

Halleluyah Oluwatobi Aworinde is a postdoctoral research fellow at Durban University of Technology, South Africa. He holds a Ph.D. in Computer Science and previously served as Director of Digital Services (ICT) at Bowen University. His research focuses on cognitive and persuasive computing with applications in healthcare, agriculture, education, and security. He has published widely in high-impact journals, contributed to international book chapters, and received fellowships and awards, including the Heidelberg Laureate Forum and ICTP Visiting Fellowship. He can be contacted at email: halleluyahA@dut.ac.za.

Oluwasegun Julius Aroba is an artificial intelligence and data science expert at the University of Johannesburg, was recently selected as one of the top 10 South African AI specialists in 2025, an NRF-rated researcher, an associate professor, senior lecturer, and an honorary research associate at Durban University of Technology. He has a Ph.D. in Information Technology. A graduate of Information Technology from the prestigious Coventry University, United Kingdom. B.Sc. Computer Science and Technology (upper-class division) Crawford University whose research inclination focus area are into wireless sensor networks, project management, data science, machine learning, SAP specialist, robotics, and artificial intelligence with over a decade year of experience in the higher education sector, healthcare industries, government parastatals, a consultant across the globe, a graduate member of IEEE, IITPSA South Africa, Member of IET UK. He can be contacted at email: oluwasegunA@dut.ac.za.

Gbohunmi Ajibesin is a test analyst at NHS Education for Scotland with a master's in Cybersecurity from the University of Derby and a bachelor's in Computer Science from Bowen University. His expertise spans software quality assurance and cybersecurity, with a focus on defect prevention, automation, and system integration testing. He has contributed to major projects across fintech and healthcare, including DDoS detection using machine learning. He previously worked with Inspired Gaming Group, First City Monument Bank, Ourpass, and Assurdly, and remains committed to advancing software reliability while exploring world through travel. He can be contacted at email: ajibesingbohunmi@gmail.com.