IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.15563

PHISHING WEBSITE DETECTION SYSTEM USING ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Mr. D.S. Jaybhay[1], Ms. Chaitrali S. Shinde[2], Ms. Bhakti D. Nannaware[3], Ms. Sakshi A. Harnawal[4] Ms. Priyanka S. Gadhe[5]

Guide,Dept. of Computer Science & Engineering, Dattakala College of Engineering, Swami Chincholi[1]

Dept. of Computer Science & Engineering, Dattakala College of Engineering, Swami Chincholi[2-5]

Abstract: Phishing websites are one of the most common cybersecurity threats used by attackers to steal sensitive information such as usernames, passwords, banking details, and personal data. These fake websites are designed to look similar to legitimate websites, making it difficult for users to identify them manually. Traditional phishing detection methods such as blacklist-based and rule-based systems are not effective for newly generated or unknown phishing websites.

This research presents a Phishing Website Detection System using Artificial Intelligence and Machine Learning. The proposed system analyzes different URL and website-based features such as URL length, number of special characters, suspicious keywords, domain-related attributes, login form presence, iframe usage, redirection behavior, and external links. Machine learning models such as Random Forest, XGBoost, and Multi-Layer Perceptron are used for classification. A hybrid ensemble voting model is applied to improve prediction accuracy and reliability.

The system is implemented as a Flask-based web application where users can enter a website URL and receive an instant prediction result. The output includes the classification result, phishing probability, risk level, important feature values, scan history, and downloadable PDF report. Experimental results show that the ensemble model performs better than individual classifiers and provides an effective solution for phishing website detection.

Index Terms: Phishing Detection, Artificial Intelligence, Machine Learning, Cybersecurity, URL Analysis, Flask, Ensemble Learning, Website Security.

I. INTRODUCTION

The rapid growth of internet services and online transactions has significantly increased the risk of cyber-attacks. Among these attacks, phishing has emerged as one of the most common and dangerous threats in the field of cybersecurity. Phishing is a type of cyber attack in which attackers create fraudulent websites or send malicious links that appear to be legitimate in order to trick users into revealing sensitive information such as usernames, passwords, credit card details, or banking credentials. These attacks are commonly distributed through emails, social media messages, advertisements, and malicious websites. As more individuals and organizations rely on online platforms for communication and financial transactions, phishing attacks continue to grow in both frequency and sophistication.

Traditional phishing detection techniques mainly rely on blacklist databases and rule-based filtering systems. Blacklist-based methods work by storing previously identified phishing URLs and blocking them when users attempt to access them. Although this approach is simple and effective for known phishing websites, it fails to detect newly generated or modified phishing URLs. Attackers frequently create new domains and change URL structures, making blacklist systems insufficient for detecting zero-day phishing attacks. Similarly, rule-based detection methods depend on predefined patterns and heuristics, which are often unable to adapt to the evolving strategies used by attackers.

To overcome these limitations, researchers have started applying Artificial Intelligence (AI) and Machine Learning (ML) techniques for phishing detection. Machine learning algorithms are capable of learning patterns from large datasets and can automatically classify URLs as phishing or legitimate based on extracted features. These techniques analyze different characteristics of URLs such as length of the URL, number of special characters, presence of suspicious domain names, redirection patterns, and other behavioral indicators. By learning from historical data, machine learning models can detect previously unseen phishing URLs more effectively than traditional methods.

DOI: 10.17148/IJARCCE.2026.15563

In recent years, hybrid AI approaches that combine multiple machine learning models and deep learning techniques have shown promising results in improving phishing detection accuracy. Ensemble learning methods integrate the predictions of multiple classifiers in order to reduce errors and improve overall performance. Additionally, deep learning models such as neural networks are capable of capturing complex relationships among features, allowing them to detect hidden phishing patterns that may not be easily identified by traditional algorithms.

This research proposes a Phishing Website Detection System using Artificial Intelligence and Machine Learning that analyzes URL structure, website content, and domain-related features to classify websites as legitimate or phishing. The system analyzes multiple types of features extracted from URLs, including structural features, lexical features, content behavior features, and domain intelligence information such as domain age, DNS records, and page ranking. Machine learning algorithms such as Random Forest and XGBoost are used along with a neural network model to build a robust ensemble classification system. Furthermore, an Isolation Forest-based anomaly detection module is incorporated to identify zero-day phishing attacks that may not follow known patterns.
Another important challenge in AI-based cybersecurity systems is the lack of transparency in model decisions. To address this issue, the proposed system integrates an Explainable Artificial Intelligence (XAI) approach using SHAP (SHapley Additive Explanations) to highlight the most influential features responsible for the prediction results. This improves user trust and helps security analysts understand the reasoning behind classification decisions. Additionally, the system includes a risk scoring mechanism that categorizes URLs into Safe, Suspicious, or High-Risk levels based on prediction probabilities.

The main objective of this research is to develop an intelligent and scalable phishing detection framework capable of identifying both known and unknown phishing URLs with high accuracy. The proposed system is designed to be implemented as a real-time web application where users can enter a URL and receive an instant security assessment. By integrating machine learning, anomaly detection, and explainable AI techniques, this research aims to contribute to the development of advanced cybersecurity solutions for protecting users against evolving phishing threats.

II. ADVANCED LITERATURE REVIEW

A. Machine Learning Approaches for Phishing Detection

Machine learning has significantly improved phishing detection by enabling systems to automatically learn patterns from large datasets of malicious and legitimate URLs. Early research in phishing detection applied traditional classification algorithms such as Decision Trees, Logistic Regression, and Support Vector Machines (SVM) to identify phishing websites based on structural and lexical URL features. Studies have shown that ensemble models such as Random Forest provide improved accuracy and robustness because they combine multiple decision trees to reduce overfitting and handle complex feature relationships. In addition, gradient boosting techniques such as XGBoost have demonstrated high performance in cybersecurity classification tasks due to their ability to optimize model predictions using sequential learning and regularization mechanisms. Recent research reports detection accuracy above 95% when tree-based ensemble models are trained on large phishing datasets containing URL, domain, and webpage behavior features. In this work, ensemble machine learning models including Random Forest and XGBoost are adopted as core components of the phishing detection framework to classify URLs based on extracted structural, lexical, and domain-related attributes.

B. Deep Learning Models for URL Classification

Deep learning techniques have recently been explored to improve phishing detection performance by learning complex feature relationships automatically. Neural networks such as Multi-Layer Perceptrons (MLP), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) have been used to analyze URL patterns and webpage characteristics. Deep neural networks are capable of capturing hidden patterns and nonlinear relationships between features that may not be detected by traditional machine learning models. For example, CNN-based models have been applied to analyze character-level patterns in URLs, while recurrent models such as Long Short-Term Memory (LSTM) networks can capture sequential dependencies in textual URL structures. However, deep learning models typically require large training datasets and computational resources. To balance performance and efficiency, this research utilizes a Multi-Layer Perceptron neural network combined with traditional machine learning models within a hybrid architecture, enabling improved classification accuracy while maintaining computational efficiency.

C. Ensemble Learning for Robust Phishing Detection

Ensemble learning has emerged as an effective approach to improve classification accuracy by combining predictions from multiple machine learning models. Instead of relying on a single classifier, ensemble techniques integrate the outputs of different models using strategies such as bagging, boosting, and voting mechanisms. Soft voting classifiers aggregate prediction probabilities from multiple models to produce a more reliable final decision. In phishing detection

DOI: 10.17148/IJARCCE.2026.15563

systems, ensemble learning helps reduce false positives and false negatives by leveraging the strengths of different algorithms. Several studies have shown that hybrid models combining Random Forest, Gradient Boosting, and neural networks outperform individual classifiers in phishing detection tasks. Motivated by these findings, the proposed system implements a hybrid ensemble model that integrates Random Forest, XGBoost, and a neural network using a soft voting strategy to enhance detection performance and system reliability.

D. Anomaly Detection for Zero-Day Phishing Attacks

A major challenge in phishing detection is identifying zero-day attacks, which involve newly created phishing URLs that do not follow known patterns present in training datasets. Traditional supervised learning models may fail to detect such attacks because they rely on previously labeled data. To address this limitation, anomaly detection techniques have been introduced to identify unusual patterns that deviate from normal behavior. Isolation Forest is one of the most widely used unsupervised anomaly detection algorithms in cybersecurity applications. It works by isolating abnormal observations in the dataset using randomly generated decision trees. Since phishing URLs often exhibit unusual structural patterns compared to legitimate URLs, Isolation Forest can effectively identify suspicious samples even when labeled data is unavailable. In this research, an Isolation Forest module is incorporated into the phishing detection framework to detect potential zero-day phishing attacks and improve the robustness of the overall system.

E. Explainable Artificial Intelligence for Cybersecurity

Although machine learning models provide high accuracy in phishing detection, many of these models operate as black boxes, making it difficult to understand the reasoning behind their predictions. Lack of transparency can reduce trust in AI-based cybersecurity systems, especially in environments where analysts need to verify detection results. Explainable Artificial Intelligence (XAI) techniques have been introduced to address this challenge by providing interpretable insights into model decisions. SHAP (SHapley Additive Explanations) is one of the most widely used explainability methods that measures the contribution of each feature to a prediction outcome. By analyzing SHAP values, researchers can identify the most influential features responsible for classifying a URL as phishing or legitimate. In this research, SHAP is integrated into the phishing detection system to improve transparency and provide clear explanations of classification decisions, enabling users and security analysts to better understand model behavior.

F. Real-Time Phishing Detection Systems

Modern cybersecurity solutions require real-time detection capabilities to protect users from malicious websites during browsing activities. Several studies have proposed web-based phishing detection frameworks that integrate machine learning models with real-time analysis tools. These systems typically extract URL features dynamically and perform classification using trained models deployed on web servers or cloud platforms. Lightweight web frameworks such as Flask are commonly used to implement real-time prediction interfaces that allow users to analyze URLs instantly. In addition, scalable architectures involving cloud deployment and API-based integration enable phishing detection services to be used across multiple platforms and applications. Building upon these approaches, the proposed research implements a real-time phishing detection system using a Flask-based web application that allows users to submit URLs and receive instant risk assessment results along with model explanations.

III. ENHANCED SYSTEM ARCHITECTURE

The proposed system architecture consists of five main layers: User Interface Layer, Feature Extraction Layer, Machine Learning Detection Layer, Result Analysis Layer, and Report Generation Layer. The user enters a website URL through the Flask web interface. The feature extraction module extracts URL-based and content-based features. These features are passed to the trained ensemble machine learning model for prediction. The system then calculates the phishing probability and risk level. Finally, the result is displayed to the user along with scan history and downloadable PDF report.

A. Data Acquisition Layer

The Data Acquisition Layer is responsible for collecting and preparing URL data required for phishing analysis. This layer gathers information from multiple sources including user-submitted URLs, publicly available phishing datasets, and domain intelligence services. When a user submits a URL through the web interface, the system automatically extracts the webpage content and associated metadata.

Several types of information are collected at this stage, including URL structural characteristics, lexical properties, domain registration information, and webpage behavioral attributes. Structural features include parameters such as URL length, number of dots, slashes, and presence of prefix-suffix patterns. Lexical features analyze suspicious keywords and character patterns often used in phishing URLs. In addition, domain-related information such as domain age, domain

DOI: 10.17148/IJARCCE.2026.15563

registration length, DNS records, and search engine indexing status are retrieved through WHOIS and DNS queries. By collecting these diverse data sources, the system ensures a comprehensive dataset for accurate phishing detection.

B. Feature Engineering Layer

The Feature Engineering Layer performs preprocessing and transformation of raw data into meaningful features suitable for machine learning models. This stage includes data cleaning, feature extraction, feature scaling, and feature selection to improve model performance and reduce computational complexity.

Feature extraction algorithms compute multiple attributes from the input URL and webpage content. These include URL structural features, lexical word features, content behavior indicators, and domain intelligence metrics. After extraction, feature normalization and scaling are applied to ensure consistent value ranges, especially for neural network models that require normalized input data.

In addition, feature selection techniques such as correlation analysis and importance ranking are applied to remove redundant or irrelevant features. Highly correlated attributes are filtered out to prevent model overfitting and improve generalization. If required, dimensionality reduction techniques such as Principal Component Analysis (PCA) can also be applied to reduce feature space while preserving essential information.

C. Intelligent Detection Layer

The Intelligent Detection Layer represents the core artificial intelligence component of the proposed system. This layer implements a hybrid AI architecture that combines multiple machine learning and deep learning models to classify URLs as phishing or legitimate.

Two tree-based machine learning models, Random Forest and XGBoost, are used to capture complex decision boundaries in structured feature data. Random Forest provides high robustness against overfitting by aggregating predictions from multiple decision trees, while XGBoost improves performance through gradient boosting optimization.

In addition to traditional machine learning models, a Multi-Layer Perceptron (MLP) neural network is implemented to capture non-linear relationships among features. The neural network consists of multiple hidden layers with activation functions that enable the model to learn complex phishing patterns that may not be detected by rule-based methods.

The predictions generated by these models are combined using an ensemble voting mechanism, where probability outputs from individual classifiers are aggregated to produce the final classification result. This ensemble approach improves prediction accuracy and reduces classification errors.

D. Intelligence and Analytics Layer

The Intelligence and Analytics Layer enhances the system with advanced analytical capabilities including anomaly detection, explainable AI, and risk assessment. One of the key components in this layer is an Isolation Forest–based anomaly detection module, which identifies abnormal patterns in URL features that may indicate zero-day phishing attacks. Since such attacks may not follow previously known patterns, anomaly detection helps improve system robustness against emerging threats.

Another important component is the Explainable Artificial Intelligence (XAI) module, implemented using SHAP (SHapley Additive Explanations). This module analyzes the contribution of individual features to the model's prediction and highlights the most influential attributes responsible for classifying a URL as phishing. By providing interpretable insights, the system improves transparency and helps cybersecurity analysts understand model decisions.

The layer also includes an Intelligent Risk Scoring Engine that categorizes URLs into different security levels based on prediction probability. URLs with high phishing probability are labeled as High Risk, while intermediate values are categorized as Suspicious, and lower probabilities are classified as Safe. This risk-based classification enables users to quickly understand the potential threat level associated with a URL.

E. Presentation Layer

The Presentation Layer provides user-facing interfaces that allow interaction with the phishing detection system. A web-based application developed using the Flask framework serves as the primary interface where users can submit URLs for analysis. Once a URL is entered, the system automatically processes the input, extracts features, performs classification using the trained models, and returns the prediction results.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.15563

The output interface displays several key elements including the predicted classification (phishing or legitimate), probability score, risk level, and explanation of influential features identified by the SHAP analysis. This information allows users to understand both the detection result and the reasons behind the decision.

Additionally, the presentation layer includes administrative dashboards for monitoring system performance and analyzing detection statistics. These dashboards display metrics such as prediction accuracy, phishing detection rate, and historical analysis of detected malicious URLs. By combining real-time detection with interpretable insights, the presentation layer enhances user awareness and supports effective cybersecurity decision-making.

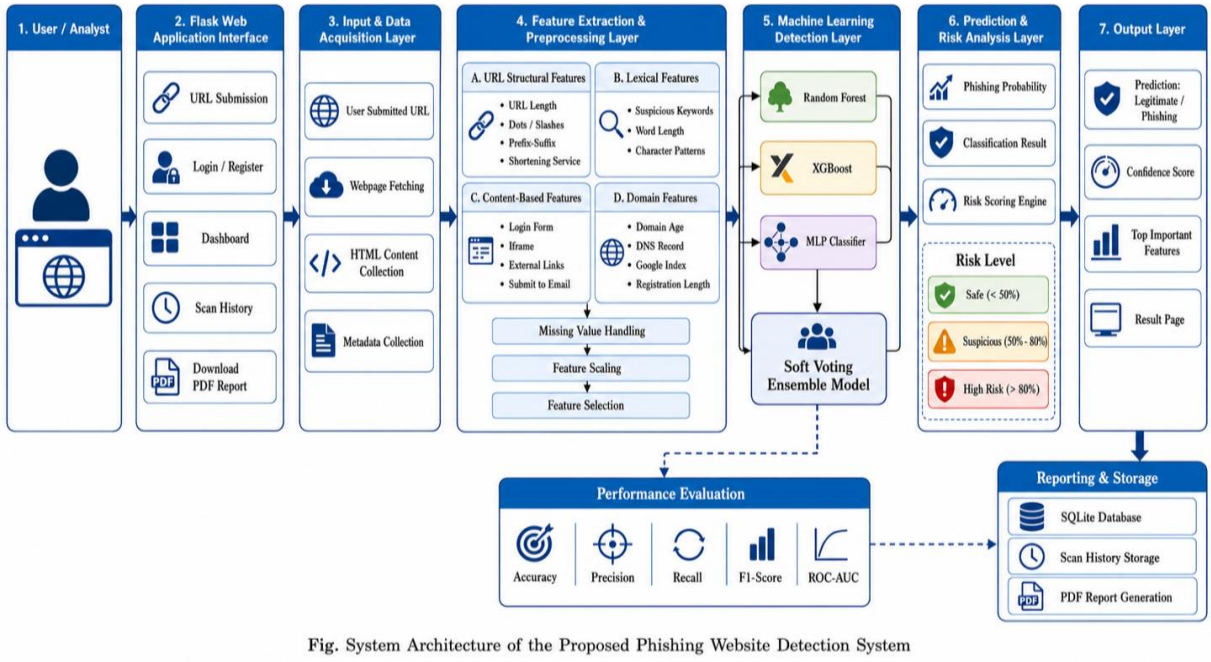

Fig. System Architecture of the Proposed Phishing Website Detection System

IV. DATASET DESCRIPTION AND FEATURE ENGINEERING

The effectiveness of any machine learning–based phishing detection system largely depends on the quality of the dataset and the feature engineering process used to represent malicious patterns. In this research, a large real-world phishing dataset is utilized to train and evaluate the proposed hybrid AI detection framework. The dataset contains both phishing and legitimate URLs along with multiple attributes that describe structural characteristics, lexical patterns, webpage behavior, and domain intelligence information. These attributes help machine learning models identify hidden patterns that distinguish malicious URLs from legitimate ones.

The dataset used in this research contains 11,430 URL samples, including 5,715 legitimate websites and 5,715 phishing websites. Each website record contains multiple extracted features related to URL structure, lexical patterns, webpage behavior, and domain-related information. The dataset is balanced, which helps the machine learning models learn both phishing and legitimate patterns effectively.

Before training the machine learning models, the dataset undergoes several preprocessing steps to ensure data quality and consistency. These steps include data cleaning, handling missing values, feature normalization, and data balancing. Duplicate records are removed to avoid bias in the training process, and missing values in domain-related attributes are handled using appropriate imputation techniques. In addition, the dataset is divided into training and testing subsets, typically using an 80:20 ratio to evaluate model performance on unseen data.

Feature engineering plays a critical role in improving phishing detection accuracy. The extracted features are categorized into four main groups: URL structural features, lexical features, content behavior features, and domain intelligence features.

A. URL Structural Features

URL structural features describe the format and structure of the URL string. Phishing URLs often contain unusual patterns such as excessive length, multiple special characters, or abnormal domain structures designed to mimic legitimate websites. Examples of structural features used in this research include:

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026
DOI: 10.17148/IJARCCE.2026.15563

- URL Length: Phishing URLs are often longer than legitimate URLs to hide malicious components.
- Number of Dots (nb_dots): Multiple subdomains are often used to imitate trusted websites.
- Number of Slashes (nb_slash): Indicates the depth of directory structure in the URL.
- Prefix-Suffix Pattern: Use of hyphens in domain names is common in phishing URLs.
- URL Shortening Services: Attackers frequently use URL shortening platforms to conceal malicious links.

These features help identify suspicious patterns commonly used in phishing attacks.

B. Lexical and Word-Based Features

Lexical features analyze the textual components of the URL and domain name. Phishing URLs often contain misleading keywords or abnormal character patterns designed to trick users into believing the website is legitimate. Examples of lexical features include:

- Character Repetition (char_repeat): Repeated characters or random strings are frequently used in malicious URLs.
- Average Word Length in Host (avg_word_host): Abnormal word lengths may indicate automated domain generation.
- Phishing Hint Words (phish_hints): Presence of suspicious keywords such as “login”, “verify”, “account”, or “secure”.
- Suspicious Top-Level Domains (suspicious_tld): Some top-level domains are more frequently used in phishing campaigns.

By analyzing these textual characteristics, the system can detect deceptive naming strategies used by attackers.

C. Content Behavior Features

Content behavior features analyze the behavior and structure of the webpage associated with the URL. Phishing websites often include elements that attempt to capture sensitive user information or redirect users to malicious destinations. Important content-based features include:

- Presence of Login Forms (login_form): Phishing websites frequently mimic authentication pages.
- Iframe Usage (iframe): Hidden iframes may be used to load malicious content.
- Popup Windows (popup_window): Suspicious popups requesting personal information.
- Links in HTML Tags (links_in_tags): Abnormal link distribution inside webpage tags.
- Email Submission (submit_email): Forms that attempt to collect user credentials through email.

These behavioral indicators help detect malicious webpages that attempt to capture sensitive data.

D. Domain Intelligence Features

Domain intelligence features provide information about domain registration and reputation. Attackers often register domains specifically for phishing campaigns, which results in identifiable domain-level patterns. Important domain features used in this research include:

- Domain Age: Newly registered domains are more likely to be used for phishing attacks.
- Domain Registration Length: Short registration durations may indicate malicious intent.
- Page Rank: Legitimate websites usually have higher search engine rankings.
- Google Index Status: Legitimate websites are typically indexed by search engines.
- DNS Record Availability: Missing or suspicious DNS records may indicate fraudulent domains.

These domain-related attributes provide additional context that improves the reliability of phishing detection models.

E. Feature Scaling and Dimensionality Reduction

After feature extraction, numerical features are normalized using feature scaling techniques such as StandardScaler to ensure that all features contribute equally during model training. This step is particularly important for neural network models that are sensitive to feature magnitude differences.

In addition, correlation analysis is performed to identify redundant or highly correlated features that may negatively affect model performance. If necessary, dimensionality reduction techniques such as Principal Component Analysis (PCA) can be applied to reduce the number of features while preserving the majority of dataset variance.

Through these feature engineering processes, the dataset is transformed into a structured format suitable for machine learning and deep learning models. This structured feature representation enables the proposed hybrid AI framework to effectively learn patterns that distinguish phishing URLs from legitimate websites, thereby improving detection accuracy and system robustness.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026

DOI: 10.17148/IJARCCE.2026.15563

V. HYBRID AI DETECTION MODEL AND METHODOLOGY

The proposed system follows a complete machine learning pipeline consisting of dataset preprocessing, feature extraction, model training, model evaluation, and web-based deployment. The input URL is analyzed using a feature extraction module that generates numerical features from the URL and webpage content. These features are then passed to trained machine learning models for prediction.

Random Forest, XGBoost, and Multi-Layer Perceptron classifiers are trained and evaluated. A soft voting ensemble model combines the prediction probabilities of these models to generate the final result. The final output is displayed through a Flask web application with risk level classification and report generation.

A. Random Forest Classification Model

Random Forest is a widely used ensemble machine learning algorithm that constructs multiple decision trees during training and combines their outputs to produce the final classification result. Each decision tree is trained on a randomly selected subset of the dataset and features, which improves generalization and reduces the risk of overfitting.

In the context of phishing detection, Random Forest is particularly effective because it can handle large numbers of input features and capture complex relationships between URL attributes. During the training phase, each decision tree evaluates features such as URL length, number of dots, suspicious keywords, and domain age to determine whether a URL is likely to be phishing or legitimate.

The final prediction of the Random Forest model is obtained through majority voting among all decision trees, which improves classification stability. Due to its robustness and ability to handle high-dimensional data, Random Forest serves as one of the primary classification models in the proposed system.

B. XGBoost Gradient Boosting Model

XGBoost (Extreme Gradient Boosting) is an advanced boosting algorithm that builds decision trees sequentially, where each new tree attempts to correct the errors made by the previous ones. This iterative learning process allows XGBoost to produce highly accurate models while maintaining computational efficiency.

XGBoost incorporates several optimization techniques such as regularization, gradient-based learning, and parallel processing, which make it highly suitable for large-scale machine learning tasks. In phishing detection, XGBoost analyzes multiple URL and domain features simultaneously to identify subtle patterns that may indicate malicious activity.

Hyperparameters such as learning rate, tree depth, and number of estimators are optimized using cross-validation techniques to achieve the best performance. The ability of XGBoost to model complex relationships between features makes it a powerful component of the hybrid detection framework.

C. Multi-Layer Perceptron Neural Network

To capture complex nonlinear relationships among features, the system incorporates a Multi-Layer Perceptron (MLP) neural network. The MLP is a feedforward deep learning model consisting of an input layer, multiple hidden layers, and an output layer.

The input layer receives the engineered feature vector extracted from the URL and domain attributes. The hidden layers apply nonlinear activation functions such as ReLU (Rectified Linear Unit) to learn complex feature interactions. The final output layer uses a sigmoid activation function to produce a probability score indicating the likelihood that the URL is phishing.

The neural network model is trained using backpropagation and gradient descent optimization techniques. Dropout layers are included during training to prevent overfitting and improve generalization. By learning hidden feature relationships, the neural network complements traditional machine learning models in detecting sophisticated phishing patterns.

D. Ensemble Voting Classifier

To improve detection accuracy and reliability, the predictions from Random Forest, XGBoost, and the neural network model are combined using an ensemble voting classifier. Ensemble learning integrates the strengths of multiple models to produce a more accurate and stable prediction.

DOI: 10.17148/IJARCCE.2026.15563

In this research, a soft voting strategy is used, where the predicted probabilities from each model are averaged to generate the final classification result. The ensemble prediction can be expressed as the weighted average of probabilities generated by individual models.
This approach reduces the likelihood of incorrect predictions caused by weaknesses in a single model. As a result, the ensemble classifier achieves better performance in terms of accuracy, precision, and recall compared to individual classifiers.

E. Isolation Forest for Anomaly Detection
A significant challenge in phishing detection is identifying zero-day attacks, which are newly created phishing URLs that may not follow patterns present in the training dataset. To address this issue, the proposed system integrates an Isolation Forest anomaly detection model.

Isolation Forest is an unsupervised learning algorithm that detects anomalies by isolating observations using randomly generated decision trees. Since phishing URLs often exhibit unusual feature patterns compared to legitimate URLs, the algorithm can effectively identify suspicious samples that deviate from normal behavior.

The anomaly score generated by the Isolation Forest is combined with the ensemble classifier's probability score to improve the detection of previously unseen phishing attacks. This additional layer of analysis enhances the robustness of the overall phishing detection system.

F. Model Training and Evaluation
The dataset is divided into training and testing sets to evaluate the performance of the proposed hybrid model. Stratified sampling is used to maintain the same class distribution in both subsets. The training process involves optimizing model parameters using Stratified K-Fold Cross-Validation, which improves generalization by evaluating model performance across multiple data partitions.

Several evaluation metrics are used to measure system performance, including accuracy, precision, recall, F1-score, and Receiver Operating Characteristic (ROC) curve analysis. Among these metrics, recall is particularly important in phishing detection because it measures the ability of the system to correctly identify malicious URLs.

Through the integration of machine learning, deep learning, and anomaly detection techniques, the proposed hybrid AI framework provides a robust and scalable approach for identifying phishing URLs in real-time environments.

VI. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

This section evaluates the performance of the proposed AI-based Phishing URL Detection System using multiple machine learning and deep learning models. The experiments are conducted to measure the effectiveness of the hybrid detection framework in accurately classifying URLs as phishing or legitimate. The performance of individual models such as Random Forest, XGBoost, and the Multi-Layer Perceptron (MLP) neural network is compared with the proposed ensemble model. In addition, the effectiveness of the anomaly detection module using Isolation Forest is analyzed to determine its ability to detect previously unseen phishing patterns.

A. Experimental Setup
The experiments were performed using a phishing dataset containing both legitimate and malicious URL samples. The dataset includes various structural, lexical, behavioral, and domain-related features extracted from each URL. Prior to model training, the dataset was preprocessed through data cleaning, feature scaling, and feature selection techniques to improve data quality.

The dataset was divided into training and testing sets using an 80:20 ratio, where 80% of the data was used for training the models and the remaining 20% was used for evaluating their performance. Stratified sampling was applied to ensure that both phishing and legitimate classes were equally represented in the training and testing subsets.

Model training and evaluation were conducted using the Python programming language with machine learning libraries such as Scikit-learn, TensorFlow, and XGBoost. Hyperparameter tuning was performed using Grid Search with Stratified K-Fold Cross-Validation to identify optimal model configurations and improve prediction accuracy

B. Evaluation Metrics
To measure the effectiveness of the proposed system, several standard performance evaluation metrics were used. These metrics are widely applied in cybersecurity and machine learning research to assess classification models.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 5, May 2026
DOI: 10.17148/IJARCCE.2026.15563

- Accuracy: Measures the overall percentage of correctly classified URLs.
- Precision: Indicates the proportion of URLs predicted as phishing that are actually phishing.
- Recall (Detection Rate): Measures the ability of the model to correctly identify phishing URLs.
- F1-Score: Harmonic mean of precision and recall, providing a balanced performance measure.
- ROC-AUC Score: Evaluates the ability of the model to distinguish between phishing and legitimate URLs.

Among these metrics, recall is particularly important for phishing detection, as failing to detect a phishing website may result in significant security risks.

*C. Performance Comparison of Machine Learning Models*Several machine learning models were evaluated to determine their effectiveness in phishing detection. The performance comparison of different models is summarized in Table 1.

Table 1: Performance Comparison of Classification Models

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Random Forest	96.24%	96.00%	96.50%	96.25%	99.30%
XGBoost	96.63%	96.03%	97.29%	96.65%	99.43%
MLP Classifier	95.67%	95.55%	95.80%	95.67%	98.89%
Hybrid Ensemble Model	96.81%	96.68%	96.94%	96.81%	99.39%

The results show that the hybrid ensemble model achieves the highest accuracy compared to individual classifiers. The combination of machine learning and deep learning models improves the system's ability to detect phishing URLs while reducing classification errors.

D. Effectiveness of Anomaly Detection
To enhance the system's capability to detect unknown phishing attacks, an Isolation Forest–based anomaly detection module was integrated into the framework. The anomaly detection model identifies URLs with unusual feature patterns that differ significantly from normal web behavior.

Experimental results indicate that the anomaly detection layer successfully identifies suspicious URLs that may not be recognized by traditional supervised learning models. This capability significantly improves the system's resilience against zero-day phishing attacks, where attackers generate new URL patterns not present in the training dataset.

E. Explainability Analysis Using SHAP
To improve transparency in model predictions, SHAP (SHapley Additive Explanations) was used to analyze the importance of individual features contributing to phishing detection. SHAP values provide insights into how each feature influences the model's prediction outcome.

Analysis of feature importance revealed that attributes such as domain age, suspicious top-level domains, presence of login forms, URL length, and phishing hint keywords have strong influence on phishing classification. By providing these explanations, the system allows users and security analysts to understand the reasoning behind prediction decisions, improving trust in the AI-based detection system.

F. Discussion of Results
The experimental results demonstrate that the proposed hybrid AI framework significantly improves phishing detection performance compared to single-model approaches. The ensemble learning mechanism increases classification accuracy, while the anomaly detection module enhances the system's ability to detect new phishing patterns. Furthermore, the integration of explainable AI techniques ensures transparency and interpretability of model decisions.

Overall, the proposed system provides an effective and scalable approach for phishing detection, capable of identifying malicious URLs with high accuracy while maintaining interpretability and real-time applicability.

VII. CHALLENGES

Although the proposed AI-based Phishing URL Detection System demonstrates promising performance in detecting malicious URLs, several challenges and limitations remain in the practical implementation and deployment of such

DOI: 10.17148/IJARCCE.2026.15563

systems. Understanding these limitations is important for improving the robustness, scalability, and reliability of AI-driven cybersecurity solutions.

A. Evolving Nature of Phishing Attacks

One of the major challenges in phishing detection is the constantly evolving nature of phishing attacks. Cyber attackers continuously develop new techniques to bypass detection systems by modifying URL structures, using domain generation algorithms, or employing URL shortening services. These rapidly changing attack patterns make it difficult for machine learning models trained on historical datasets to detect newly generated phishing websites effectively. Although the proposed system incorporates anomaly detection to address zero-day attacks, completely eliminating this challenge remains difficult.

B. Dataset Quality and Availability

The performance of machine learning models largely depends on the quality and diversity of the training dataset. In many cases, publicly available phishing datasets may contain outdated information or may not represent the latest phishing strategies used by attackers. In addition, some datasets may suffer from class imbalance, where legitimate URLs significantly outnumber phishing samples. Such imbalances can affect model training and lead to biased predictions. Ensuring access to large, updated, and well-balanced datasets remains a challenge for researchers working in phishing detection.

C. Feature Extraction Complexity

Another limitation is related to the extraction of certain features required for accurate phishing detection. Features such as domain age, WHOIS information, DNS records, and webpage content analysis may require additional network requests and external data sources. These operations may increase system latency, especially in real-time detection scenarios. In some cases, domain information may also be hidden or protected due to privacy policies, making it difficult to retrieve complete domain intelligence data.

D. Model Interpretability and Trust

While machine learning models provide high detection accuracy, some models—particularly deep learning architectures—operate as complex black-box systems. This lack of transparency can make it difficult for users and security analysts to fully understand the reasoning behind classification decisions. Although the integration of Explainable AI techniques such as SHAP improves interpretability, explaining complex model behavior in an intuitive way for non-technical users remains a challenge.

E. Real-Time Deployment Constraints

Deploying phishing detection models in real-time environments introduces additional challenges related to system performance and scalability. Real-time detection requires rapid feature extraction, model inference, and result presentation without causing delays in user browsing activities. Handling large volumes of incoming URL requests in production environments may require high computational resources and optimized infrastructure. Ensuring low-latency prediction while maintaining high accuracy is therefore an important design consideration.

F. False Positives and False Negatives

Despite high accuracy, no phishing detection system can achieve perfect classification. False positives occur when legitimate websites are incorrectly classified as phishing, which may affect user trust and accessibility. Conversely, false negatives occur when phishing websites are incorrectly identified as legitimate, potentially exposing users to security risks. Reducing these classification errors remains a key challenge in improving the reliability of AI-based phishing detection systems.

VIII. FUTURE DIRECTIONS

Although the proposed AI-based Phishing URL Detection System demonstrates strong performance in identifying phishing websites using machine learning and deep learning techniques, there are several opportunities for further improvement and research. Future work can focus on enhancing system intelligence, expanding detection capabilities, and improving real-time deployment in large-scale cybersecurity environments.

A. Integration of Advanced Deep Learning Models

Future research can explore more advanced deep learning architectures such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) for analyzing complex URL patterns and webpage content. These models can automatically learn hierarchical feature representations from raw URL strings or HTML content, potentially improving

DOI: 10.17148/IJARCCE.2026.15563

the detection of sophisticated phishing attacks. Additionally, transformer-based architectures could be investigated for better contextual analysis of textual and structural data.

B. Real-Time Browser Extension Implementation

An important future direction is the development of a browser-based phishing detection extension that integrates directly with web browsers. Such an extension can analyze URLs in real time when users attempt to access a website and immediately provide warnings if the link is suspected to be phishing. This would significantly enhance user protection by preventing access to malicious websites before sensitive information is entered.

C. Continuous Learning and Model Updating

Phishing attacks evolve rapidly, which requires detection systems to adapt to new patterns continuously. Future versions of the system can incorporate automated model retraining mechanisms that periodically update the machine learning models using newly collected phishing data. Continuous learning frameworks can help the system remain effective against emerging phishing strategies and newly generated malicious domains.

D. Integration with Threat Intelligence Platforms

Another potential enhancement is the integration of the phishing detection system with cyber threat intelligence platforms. By combining machine learning predictions with external threat intelligence feeds, the system can access additional contextual information about domains, IP addresses, and malicious campaigns. This integration would improve detection accuracy and allow the system to respond more effectively to coordinated cyber threats.

E. Mobile and Cloud-Based Security Applications

Future work may also focus on extending the system to mobile security applications and cloud-based cybersecurity services. Implementing the phishing detection model within mobile security applications can protect smartphone users from malicious links received through messaging platforms or emails. Cloud-based deployment would enable scalable security services capable of analyzing large volumes of URLs in enterprise environments.

F. Advanced Explainable AI for Security Analysis

Further research can improve the explainability of phishing detection models by integrating more advanced Explainable Artificial Intelligence (XAI) techniques. Improved visualization tools and interactive dashboards could help cybersecurity analysts better understand phishing patterns, feature importance, and evolving attack strategies. Such transparency would increase trust in AI-based security systems and support more informed decision-making.

IX. CONCLUSION

Phishing websites are a major cybersecurity threat because they trick users into entering sensitive information on fake websites. Traditional blacklist-based detection methods are not effective against newly created phishing websites. Therefore, this research presents a **Phishing Website Detection System using Artificial Intelligence and Machine Learning** to detect phishing websites more accurately.

The proposed system extracts URL-based, lexical, content-based, and d omain-related features from websites. Machine learning models such as Random Forest, XGBoost, and Multi-Layer Perceptron are trained and evaluated. A soft voting ensemble model is used to combine the strengths of these classifiers and improve prediction performance.

The system is implemented as a Flask-based web application where users can enter a website URL and receive an instant result. The result includes prediction status, phishing probability, risk level, important feature values, scan history, and downloadable PDF report. Experimental results show that the hybrid ensemble model achieves an accuracy of 96.81% and ROC-AUC score of 99.39%, making it effective for phishing website detection.

In the future, the system can be improved by adding real-time WHOIS and DNS lookup, SHAP-based explainability, Isolation Forest for zero-day phishing detection, browser extension support, and cloud deployment.

.

ACKNOWLEDGMENT

The authors express sincere gratitude to Mr. D.S. Jaybhay for his continued mentorship and technical guidance throughout Stage 2 development. The authors also acknowledge the Department of Computer Science and Engineering, Dattakala Group of Institution Faculty of Engineering, Swami Chincholi, for providing the computational resources and

DOI: 10.17148/IJARCCE.2026.15563

research environment that made this work possible. Special thanks are extended to the reviewers whose constructive feedback on the Stage 1 paper directly informed the design decisions presented in this paper.

REFERENCES

[1]. A. K. Jain and B. B. Gupta, "Phishing detection: Analysis of visual similarity based approaches," *Security and Communication Networks*, vol. 2017, pp. 1–20, 2017.

[2]. M. Aburrous, M. A. Hossain, F. Thabatah, and K. Dahal, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7913–7921, 2010.

[3]. S. Marchal, J. Francois, R. State, and T. Engel, "PhishStorm: Detecting phishing with streaming analytics," *IEEE Transactions on Network and Service Management*, vol. 11, no. 4, pp. 458–471, 2014.

[4]. I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the International World Wide Web Conference*, 2007, pp. 649–656.

[5]. R. Verma and N. Hossain, "Semantic feature selection for text with application to phishing email detection," in *Proceedings of the IEEE International Conference on Data Mining Workshops*, 2017, pp. 455–462.

[6]. J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious web sites from suspicious URLs," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 1245–1254.

[7]. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, New York, NY, USA: Springer, 2009.

[8]. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9]. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[10]. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Cambridge, MA, USA: MIT Press, 2016.

[11]. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12]. M. Zahid, M. Masood, and A. Shibli, "Machine learning based phishing detection using URL features," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 7, pp. 1–8, 2019.

[13]. F. T. Alsharnouby, F. Alaca, and S. Chiasson, "Why phishing still works: User strategies for combating phishing attacks," *International Journal of Human-Computer Studies*, vol. 82, pp. 69–82, 2015.

[14]. F. Chollet, *Deep Learning with Python*, 2nd ed. Shelter Island, NY, USA: Manning Publications, 2021.

[15]. L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1–2, pp. 1–39, 2010.

[16]. F. T. Liu, K. M. Ting, and Z. Zhou, "Isolation forest," in *Proceedings of the IEEE International Conference on Data Mining*, 2008, pp. 413–422.

[17]. S. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[18]. B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, "Defending against phishing attacks: Taxonomy of methods, current issues and future directions," *Telecommunication Systems*, vol. 67, pp. 247–267, 2018.

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

[20]. A. Aggarwal, A. Rajadesingan, and P. Kumaraguru, "PhishAri: Automatic realtime phishing detection on Twitter," in *Proceedings of the IEEE International Conference on eCrime Researchers Summit*, 2012.

ABOUT THE AUTHORS

Ms. Chaitrali S. Shinde is a final-year undergraduate student in the Department of Computer Science and Engineering at Dattakala Group of Institution Faculty of Engineering, Swami Chincholi. Her research interests include artificial intelligence, machine learning, cybersecurity, and phishing detection systems. She is currently working on the research project titled *"Phishing Website Detection System Using AI and Machine Learning,"* which focuses on developing intelligent models for identifying malicious websites and improving online security.

Ms. Bhakti D. Nannaware is a final-year undergraduate student in the Department of Computer Science and Engineering at Dattakala Group of Institution Faculty of Engineering, Swami Chincholi. Her research interests include machine

DOI: 10.17148/IJARCCE.2026.15563

learning, data analytics, and cybersecurity applications. She is involved in the research project *"Phishing Website Detection System Using AI and Machine Learning,"* contributing to the development of feature engineering techniques and machine learning models for phishing detection.

Ms. Sakshi A. Harnawal is a final-year undergraduate student in the Department of Computer Science and Engineering at Dattakala Group of Institution Faculty of Engineering, Swami Chincholi. Her research interests include distributed systems, data security, and web application development. She is currently contributing to the project *"Phishing Website Detection System Using AI and Machine Learning,"* focusing on system implementation and real-time web-based phishing detection.

Ms. Priyanka S. Gadhe is a final-year undergraduate student in the Department of Computer Science and Engineering at Dattakala Group of Institution Faculty of Engineering, Swami Chincholi. Her research interests include artificial intelligence, data mining, and cybersecurity. She is working on the research project *"Phishing Website Detection System Using AI and Machine Learning,"* where she contributes to model evaluation, performance analysis, and system optimization for detecting phishing URLs.

ABOUT THE PROJECT GUIDE

Mr. D. S. Jaybhay is the project guide for the research work titled "Phishing Website Detection System Using Artificial Intelligence and Machine Learning." He is associated with the Department of Computer Engineering, Dattakala Group of Institution Faculty of Engineering, Swami Chincholi. His guidance and technical support helped the project team in system design, research methodology, implementation planning, and documentation.
For research paper, this is better than writing the guide as a normal author. Keep students under About the Authors and guide under About the Project Guide.

