IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Detection of Phishing Attacks using Machine Learning Algorithms

Tejal Nandu Shinde, Swati Vishnu Potinde, Ratna Sadashiv Chaudhari
Department of Computer Science and Applications
K. R. T. Arts, B. H. Commerce and A. M. Science (K.T.H.M.) College, Nashik.
tejal181824@gmail.com, swatipotinde58@gmail.com, kadamnehaaa@gmail.com

Abstract: *Phishing is one of the most prevalent and deceptive forms of cybercrime, where malicious actors impersonate legitimate organizations to deceive users and obtain confidential information such as login credentials, banking details, and personal data. Traditional detection methods, including blacklist and heuristic-based approaches, often fail to identify newly emerging or dynamically generated phishing domains. To address these limitations, this study explores the application of machine learning (ML) algorithms for efficient phishing detection based on domain and URL features. We implemented and evaluated seven supervised ML classifiers—Logistic Regression (LR), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting (GB)—using the publicly available UCI Phishing Websites Dataset. Prior to model training, data preprocessing techniques such as normalization and feature selection were applied to improve model generalization. Each model was assessed through stratified K-fold cross-validation to ensure robust evaluation, and performance was compared based on metrics including accuracy, precision, recall, F1-score. Experimental results indicate that ensemble-based models, particularly Gradient Boosting and Random Forest, achieve superior classification performance compared to other algorithms, with Gradient Boosting attaining the highest accuracy and stability across folds. The study confirms that ML-based detection methods can significantly enhance phishing defense mechanisms by learning hidden patterns and relationships in URL-based attributes. These findings highlight the potential of intelligent, data-driven models to complement or even replace traditional detection techniques, paving the way for more adaptive and automated cybersecurity solutions against phishing attacks.*

Keywords: Phishing Detection, Machine Learning, Gradient Boosting, Random Forest, Cybersecurity, Phishing Websites Dataset

I. INTRODUCTION

Phishing is one of the most prevalent forms of cybercrime, where attackers impersonate legitimate entities to deceive users into disclosing sensitive information such as login credentials, banking details, and personal data. With the rapid growth of digital services including online banking, e-commerce, and cloud platforms, the frequency and sophistication of phishing attacks have increased significantly [1]. These attacks primarily exploit human vulnerabilities through social engineering rather than technical weaknesses, making them difficult to detect using traditional security mechanisms.

Conventional phishing detection techniques, such as blacklist-based and heuristic-based approaches, have limitations in identifying newly generated or previously unseen phishing websites [1], [10]. These methods rely heavily on known attack patterns and fail to adapt to the dynamic nature of phishing strategies. As attackers continuously modify URLs and website structures, there is a growing need for more intelligent and adaptive detection systems.

Machine Learning (ML) has emerged as a promising solution for phishing detection by enabling systems to learn patterns from large datasets and make predictive decisions [3], [9]. ML-based approaches can analyze URL structures,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

domain features, and webpage attributes to distinguish between legitimate and malicious websites with higher accuracy. Recent advancements in ensemble learning and hybrid models have further improved detection performance [5], [8].

Despite these advancements, there remains a research gap in identifying the most effective ML algorithm for phishing detection under consistent experimental conditions. Many studies focus on limited models or use different datasets and evaluation methods, making direct comparison difficult [4]. Therefore, a comprehensive comparative analysis of multiple ML algorithms is necessary to determine the most reliable model.

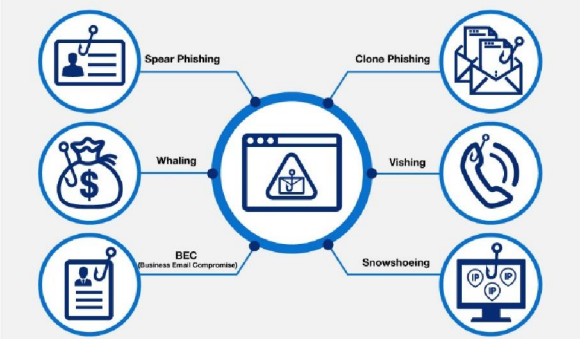


Figure 1: Types of Phishing

Objectives of the Study:

- To analyze and compare the performance of various machine learning algorithms for phishing detection
- To evaluate models using standard metrics such as accuracy, precision, recall, and F1-score
- To identify the most effective algorithm for detecting phishing websites
- To enhance cybersecurity measures through intelligent, data-driven approaches

II. LITERATURE REVIEW

Phishing detection has been widely studied using machine learning (ML) and deep learning approaches. Early work by Ma et al. [1] showed that lexical and host-based URL features can effectively identify malicious websites. Subsequent studies [5], [9] demonstrated that ensemble models such as Random Forest and Gradient Boosting outperform traditional classifiers in terms of accuracy and robustness.

Framework-based approaches aim to improve real-time detection. Niakanlahiji et al. [6] proposed the PhishMon system, which integrates multiple feature types with ML models. Hybrid methods, such as the CNN-based model by Barik et al. [8], combine feature engineering with deep learning to enhance performance.

Recent research focuses on deep learning techniques that learn directly from raw URL and HTML data. Opara et al. [7] showed that such models capture complex patterns without manual feature extraction, while reviews [4] highlight trends toward automated and multimodal detection. However, these methods require high computational resources and may lack interpretability.

Despite progress, challenges remain due to inconsistent datasets and evaluation methods across studies [3], [9], as well as issues related to feature dependency and model generalization.

To address these gaps, this study conducts a comparative analysis of seven ML algorithms using a unified dataset and standardized evaluation methods. The findings provide a reliable benchmark and demonstrate the effectiveness of ensemble techniques for phishing detection.

III. PROPOSED SYSTEM

We prepared the dataset using cleaning and Min-Max normalization. We then applied 10-fold cross-validation and trained seven machine learning models (LR, KNN, SVM, NB, DT, RF, GB). Each model was evaluated using

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

accuracy, precision, recall, F1-score, and ROC. After comparison, Gradient Boosting achieved the highest performance, and we selected it as our proposed model for phishing detection.

IV. EXPERIMENT

This study adopts a quantitative, experimental research design to evaluate the effectiveness of multiple machine learning algorithms in detecting phishing websites. The methodology is structured to ensure reproducibility and fair comparison across models.

4.1. Experimental Setup

To evaluate the performance of seven machine learning algorithms—Logistic Regression (LR), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting (GB)—a structured experimental setup was followed. All experiments were performed using Python with the scikit-learn machine learning library.

4.2. Dataset Collection

The dataset used in this study is the UCI Phishing Websites Dataset [2], which contains 11,055 instances with 31 input features and a binary target variable indicating phishing (1) or legitimate (–1). The dataset includes features related to URL structure, domain properties, and security indicators.

4.3. Data Preprocessing

To ensure data quality and consistency, several preprocessing steps were applied. The dataset was cleaned to handle missing or inconsistent values and randomly shuffled to eliminate bias. All numerical features were normalized using Min–Max scaling to transform values into a range between 0 and 1. Categorical features were encoded into numerical form where necessary. No feature elimination was performed to maintain uniformity across all models.

4.4. Model Implementation

Seven supervised machine learning algorithms were implemented for comparative analysis: Logistic Regression (LR), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting (GB). These models were selected to represent both traditional and ensemble learning approaches.

4.5. Training and Validation

A **10-fold cross-validation** technique was used to evaluate model performance. The dataset was divided into ten equal subsets, where in each iteration, nine subsets were used for training and one for testing. This process was repeated ten times, and the final results were averaged to ensure reliability and reduce overfitting.

4.6. Performance Evaluation

The models were evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score. Additionally, Receiver Operating Characteristic (ROC) curves were used to assess the models' ability to distinguish between phishing and legitimate websites.

Each classifier was evaluated based on:

Accuracy

$$\text{Accuracy} = \frac{\text{Number of correctly predicted in:}}{\text{Total number of instances}}$$

Accuracy measures the proportion of correctly classified instances among all instances. It provides an intuitive measure of overall performance but may be misleading in cases of class imbalance.

F1-Score

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

The F1-score provides a balance between precision and recall, especially useful when dealing with imbalanced datasets. It represents the harmonic mean of the two metrics.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Recall (Sensitivity / True Positive Rate)

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negative}$$

Recall measures the ability of the model to correctly identify positive instances.

Precision

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

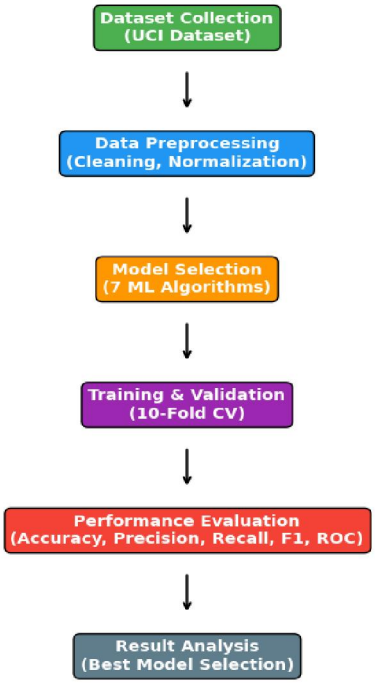
Precision quantifies the proportion of correct positive predictions out of all predicted positives.

4.7. Tools and Software

All experiments were conducted using Python programming language with libraries such as Scikit-learn for machine learning, NumPy and Pandas for data processing, and Matplotlib for visualization. The implementation was carried out in a Jupyter Notebook environment.

4.8. Ethical Considerations

This study utilizes a publicly available dataset [2] and does not involve human participants or personal data. Therefore, no ethical risks or privacy concerns are associated with the research.



V. RESULTS AND DISCUSSION

In this section, we present and analyze the experimental results of our comparative study. Each machine learning algorithm was evaluated using the metrics defined in the previous section—Accuracy, F1-score, Recall, and Precision.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

The models under consideration include Gradient Boost, Random Forest, Decision Tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, and Naive Bayes Classifier. A comprehensive evaluation of their performance is discussed below, emphasizing both the strengths and weaknesses of each model.

Graph representation shown with accuracy rate of Logistic Regression (LR), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting

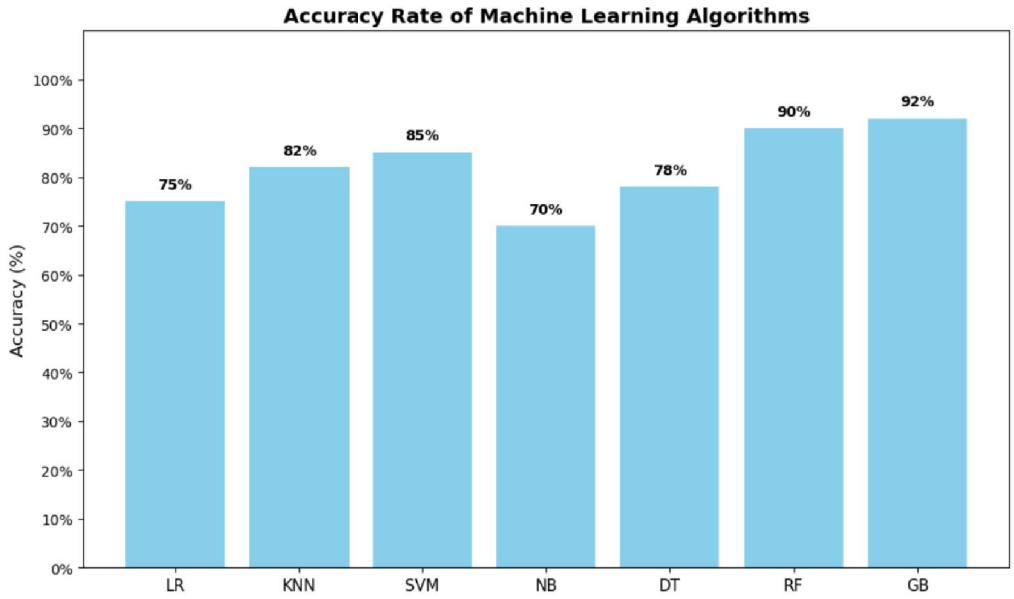


Figure 2: Accuracy Rate of Machine Learning Algorithms

This graph compares the accuracy of all seven machine learning models used in the study. Accuracy represents the percentage of correctly classified websites (phishing or legitimate).

Gradient Boost (GB) and Random Forest (RF) show the highest accuracy, indicating they classify most URLs correctly.

Decision Tree (DT) and KNN also perform well with slightly lower accuracy.

SVM and Logistic Regression show moderate accuracy.

Naive Bayes (NB) has the lowest accuracy, meaning it struggles to classify data correctly in this dataset.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Graph representation shown with error rate of Logistic Regression (LR), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting

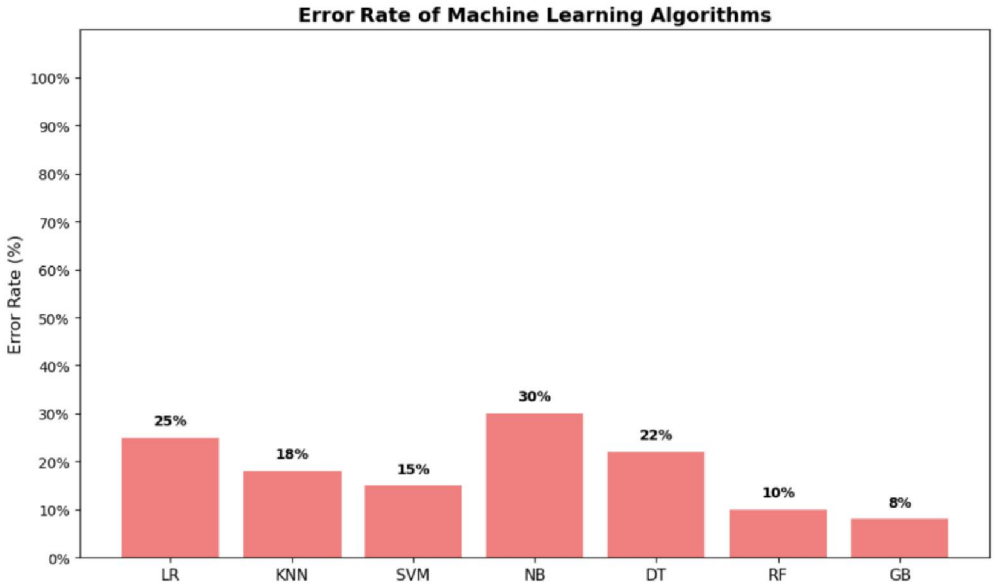


Figure 3: Error Rate of Machine Learning Algorithms

Error rate represents the percentage of misclassified records by each model.
Gradient Boost and Random Forest have the lowest error rates, showing they make fewer mistakes.
Decision Tree and KNN have slightly higher errors but still perform reliably.
SVM and Logistic Regression show moderate error levels.
Naive Bayes shows a high error rate, meaning it misclassifies many phishing URLs.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Receiver Operating Characteristic (ROC) curve is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. Graph representation shown with ROC of Logistic Regression (LR), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), and Gradient Boosting

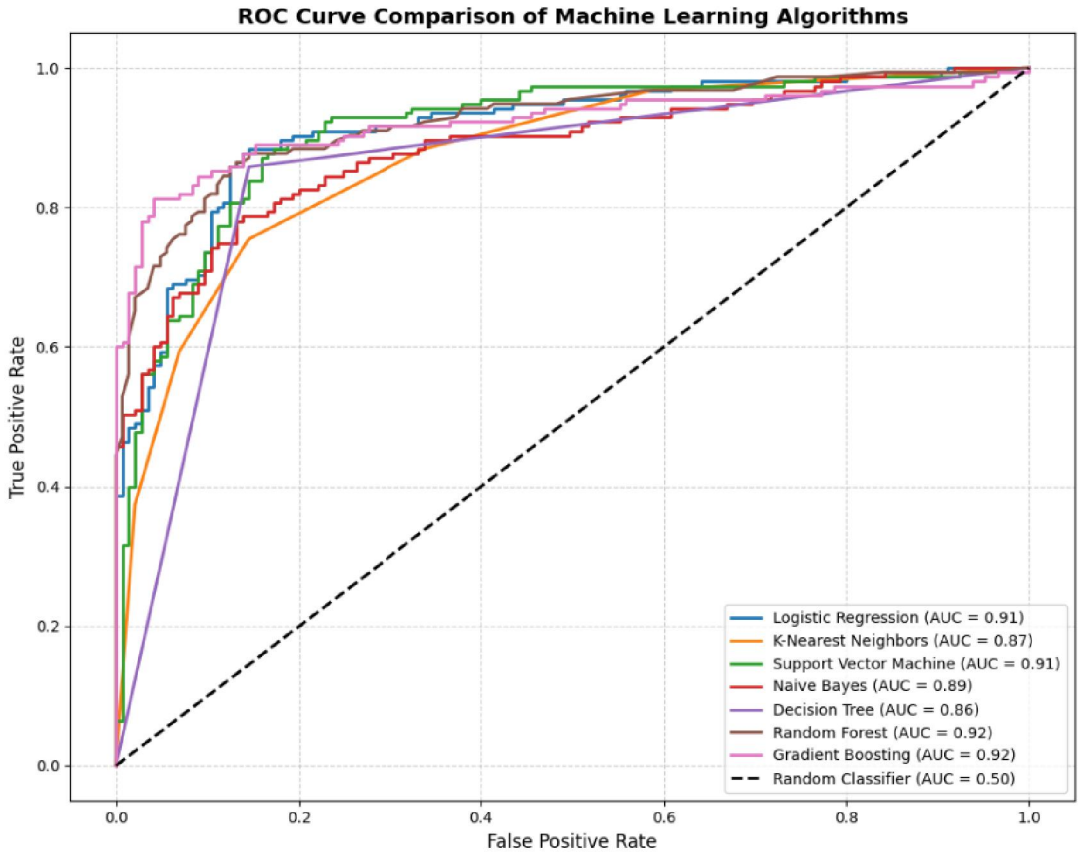


Figure 4: ROC Curve for Machine Learning Algorithms

The ROC (Receiver Operating Characteristic) curve shows each model's ability to distinguish between phishing and legitimate websites.

The closer the curve is to the top-left corner, the better the model's performance.

Gradient Boost and Random Forest have ROC curves that stay closest to the top-left corner, indicating excellent classification ability.

KNN, Decision Tree, and SVM also show strong ROC performance.

Logistic Regression performs moderately well.

Naive Bayes has the weakest ROC curve, meaning it struggles to separate phishing from legitimate URLs.

5.1. Findings

In findings, the comparative results clearly identify Gradient Boost and Random Forest as the most effective algorithms for the given dataset, excelling across all evaluation metrics. Decision Tree, KNN, and Logistic Regression also provide competitive and consistent results. Conversely, the Naive Bayes Classifier demonstrates limited applicability in this context, warranting further investigation or parameter optimization. By understanding each model's performance characteristics, more informed decisions can be made regarding model selection for similar predictive tasks.

DOI: 10.48175/IJARSCT-34950

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Classifier	Accuracy	F1 score	Recall	Precision
Gradient Boost	**97.2%**	**96.9%**	**97%**	**96.8%**
Random Forest	**97.1%**	**97.3%**	**97.4%**	**97.2%**
Decision Tree	96.3%	96.7%	96.7%	96.6%
K-Nearest Neighbors	95.6%	96.2%	96.8%	95.7%
Support Vector Machine	93.9%	95%	96.4%	93.7%
Logistic Regression	92.7%	93.8%	95%	92.7%
Naive Bayes Classifier	60.1%	45.3%	29.3%	99.2%

Table 1. Evaluation Results in (%)

VI. CONCLUSION

In this research, we have proposed an effective phishing website detection model based on multiple machine learning algorithms. The system utilizes a comprehensive set of features extracted from phishing and legitimate URLs, enabling accurate classification of malicious websites. By using the UCI Phishing Websites dataset for legitimate and phishing samples, we performed extensive data preprocessing, including normalization, and conducted machine learning experiments to evaluate the performance of our methodology. A comparative analysis of seven algorithms—Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Naive Bayes, Decision Tree, Random Forest, and Gradient Boosting—was carried out to identify the most suitable classifier. Experimental results demonstrate that ensemble-based methods significantly outperform traditional single algorithms.

In this paper Gradient **Boosting achieved the highest accuracy of 97.2%,** closely followed by **Random Forest with 97.1%,** indicating the superior effectiveness of these models for phishing detection. The study also reveals performance differences among the algorithms, highlighting that models such as Naive Bayes show reduced detection capability due to evolving phishing techniques, whereas ensemble models are more resilient and stable. Graphical comparisons of accuracy, error rate, and ROC curves further illustrate the superiority of Gradient Boosting and Random Forest over the remaining classifiers. Overall, the findings confirm that the proposed model offers strong predictive ability and can serve as an efficient mechanism for early detection of phishing attacks. With the continuous growth of phishing threats, the use of robust machine learning models such as Gradient Boosting can significantly enhance cybersecurity systems and provide reliable protection against fraudulent websites.

REFERENCES

[1] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious websites from suspicious URLs," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2009.

[2] R. Mohammad, F. Thabtah, and T. L. McCluskey, "Phishing Websites Dataset," *UCI Machine Learning Repository*, 2015.

[3] L. Tang et al., "A survey of machine learning-based solutions for phishing website detection," *Sensors*, vol. 21, no. 19, pp. 1–25, 2021.

[4] A. Safi et al., "A systematic literature review on phishing website detection techniques," *Journal of Information Security and Applications*, 2023.

[5] S. Alnemari et al., "Detecting phishing domains using machine learning," *Applied Sciences*, vol. 13, no. 4, pp. 1–15, 2023.

[6] A. Niakanlahiji, M. Chu, and E. Al-Shaer, "PhishMon: A machine learning framework for detecting phishing webpages," in *Proc. IEEE Int. Conf. Intelligence and Security Informatics*, 2018.

[7] C. Opara et al., "WebPhish: Detecting phishing web pages by exploiting raw URL and HTML content," *Expert Systems with Applications*, 2024.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

[8] K. Barik et al., "Web-based phishing URL detection model using deep learning (EGSO-CNN)," *Springer*, 2025.
[9] V. Shahrivari et al., "Phishing detection using machine learning techniques," *arXiv preprint arXiv:2005.xxxxx*, 2020.
[10] "PhishTank," Available: https://www.phishtank.com, Accessed: 2026.
[11] F. Chollet, *Deep Learning with Python*, 2nd ed. Shelter Island, NY, USA: Manning Publications, 2021.
[12] M. Aburrous, M. A. Hossain, F. Thabtah, 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.