10.48047/jocaaa.2023.31.04.52

Comparative Performance Study of Machine Learning Algorithms for Phishing URL Detection in Cyber Security Systems

G.Lingaiah[1]
Research Scholar,
Dept. of Computer Science and Engineering,
Arni University, Himachal Pradesh, India.
lingaiah.g80@gmail.com

Dr.Ram Kinkar Pandey[2]
Professor,
Dept. of Computer Science and Engineering,
Arni University, Himachal Pradesh, India.
Dr.ramkpandey@gmail.com

Abstract

Phishing remains one of the most pervasive and damaging forms of cybercrime, exploiting deceptive URLs to trick unsuspecting users into revealing sensitive information such as banking credentials, login details, and personal data. Traditional detection techniques such as blacklists, whitelists, and heuristic filters struggle to detect zero-day phishing URLs and often suffer from high false positives, limiting their effectiveness. Machine learning (ML) offers a more adaptive solution by leveraging discriminative URL features to dynamically distinguish phishing from legitimate websites. In this research, we present a comparative performance analysis of seven classical machine learning algorithms Support Vector Machine (SVM), Decision Tree, Random Forest, Gradient Boost, Logistic Regression, K-Nearest Neighbors (KNN), and Naïve Bayes Classifier using the PhiUSIIL_Phishing_URL_Dataset, which contains 11,055 URLs represented with 30 handcrafted lexical and host-based features. The models were systematically evaluated using standard performance metrics including accuracy, precision, recall, and F1-score. Experimental results reveal that four models (SVM, Decision Tree, Random Forest, and Gradient Boost) achieved perfect detection performance, attaining 100% across all metrics. Logistic Regression and KNN followed closely with 99.7% accuracy, while Naïve Bayes achieved 99.2% accuracy but yielded the highest precision (0.999), indicating strong resilience against false positives. These findings confirm that phishing URL detection can be addressed with near-perfect accuracy using well-established machine learning algorithms, without requiring deep or computationally expensive models. Furthermore, the study highlights practical trade-offs between accuracy, interpretability, and computational efficiency, offering actionable insights for deploying lightweight, real-time phishing detection systems in modern cybersecurity infrastructures.

Keywords: Phishing detection, Cybersecurity, Machine learning, URL classification, Random Forest, Support Vector Machine, Gradient Boost, PhiUSIIL_Phishing_URL_Dataset.

I. INTRODUCTION

Phishing attacks continue to rank among the most widespread and costly forms of cybercrime, leveraging fraudulent websites and deceptive URLs to trick unsuspecting users into revealing sensitive information such as banking credentials, login details, and personal identity data. Reports from the FBI's Internet Crime Complaint Center (IC3) confirm phishing as the most reported cybercrime globally, with

10.48047/jocaaa.2023.31.04.52

billions of dollars in losses annually (FBI IC3, 2023). The rapid generation of phishing domains, often hosted on compromised servers or dynamically registered, has made this threat increasingly difficult to contain. Traditional defense mechanisms such as blacklists and whitelists, though widely adopted, are inherently limited. Blacklists are reactive in nature and fail to detect zero-day phishing domains, while whitelists may restrict legitimate browsing activities by blocking newly registered benign domains (Ma et al., 2009).

Heuristic-based methods improve detection by analyzing structural and lexical URL characteristics, yet they often suffer from high false positives and lack robustness when adversaries employ obfuscation or mimicry techniques (Marchal et al., 2014). In recent years, machine learning (ML) has emerged as a more adaptive and scalable solution to phishing detection. By leveraging lexical, host-based, and contextual URL features, ML classifiers can learn discriminative patterns and generalize beyond previously observed attacks. Studies have shown that ML-based approaches outperform traditional techniques in accuracy, recall, and resilience against novel phishing campaigns (Mohammad et al., 2014; Jain & Gupta, 2018; Basit et al., 2020). More recently, research between 2022 and 2024 has highlighted the increasing effectiveness of ensemble methods, deep learning architectures, and hybrid feature engineering techniques in achieving near-perfect phishing detection performance (Simhadri et al., 2025).

Despite these advances, critical challenges remain. Many studies report high accuracy but fail to provide systematic benchmarking across multiple ML classifiers using standardized datasets. Furthermore, trade-offs involving computational cost, interpretability, and scalability are often underexplored, making it difficult to select models suited for real-time deployment in resource-constrained environments (Abdelhamid et al., 2014; Basit et al., 2020). This research work addresses these gaps by conducting a comparative performance analysis of seven well-established ML classifiers Support Vector Machine (SVM), Decision Tree, Random Forest, Gradient Boost, Logistic Regression, K-Nearest Neighbors (KNN), and Naïve Bayes using the PhiUSIIL_Phishing_URL_Dataset, which includes 11,055 phishing and legitimate URLs described with 30 handcrafted lexical and host-based features. The main contributions of this work are as follows:

Benchmarking seven supervised ML algorithms on a benchmark phishing URL dataset to ensure fair and reproducible comparison across accuracy, precision, recall, and F1-score.

Demonstrating that multiple models achieve near-perfect detection performance, with ensemble and margin-based classifiers achieving 100% across all metrics.

Providing actionable insights for deploying phishing detection systems in real-world cybersecurity settings, with an emphasis on balancing detection effectiveness, computational efficiency, and interpretability.

II. RELATED WORK

By systematically evaluating widely used ML models, this study validates the viability of classical machine learning algorithms for phishing detection and provides guidance for selecting lightweight yet robust solutions for scalable, real-time cybersecurity defense. Early phishing detection techniques primarily relied on blacklists (Ma et al., 2009) and whitelists (FBI IC3, 2023). Blacklists store previously identified malicious domains, while whitelists restrict browsing to approved legitimate sites. However, these methods are reactive and fail to detect zero-day phishing URLs, which

10.48047/jocaaa.2023.31.04.52

attackers can generate rapidly in large numbers. To address these shortcomings, heuristic-based detection methods were proposed, using lexical, structural, and content-based URL features to identify suspicious domains. While such approaches improve coverage against novel threats, they suffer from scalability challenges and often generate high false positives when benign websites share similar structural traits (Marchal et al., 2014).

The application of machine learning (ML) represented a major shift in phishing detection research. By learning statistical patterns from labeled datasets, ML classifiers generalize better than traditional methods and detect unseen phishing attempts. Among classical algorithms, Random Forests have consistently demonstrated strong predictive performance and robustness against overfitting (Basit et al., 2020). Similarly, Support Vector Machines (SVMs) have been widely used due to their ability to handle high-dimensional feature spaces effectively (Jain & Gupta, 2018). Other lightweight classifiers such as Naïve Bayes and Logistic Regression have also been applied, typically achieving accuracy levels between 85% and 95%, though often at the expense of precision on large or imbalanced datasets (Abdelhamid et al., 2014). More recent research has explored hybrid approaches, which combine heuristic features with machine learning algorithms to balance interpretability and predictive strength (Abdelhamid et al., 2014).

The rise of deep learning models has also significantly influenced phishing detection. Approaches such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) enable end-to-end feature learning from raw URL strings or webpage content, thereby reducing reliance on manual feature engineering (Simhadri et al., 2025). While deep models achieve state-of-the-art accuracy, their computational requirements can make them less suitable for real-time detection in resource-constrained environments. This research work extends this body of research by conducting a comprehensive comparative study of seven classical ML algorithms on the PhiUSIIL_Phishing_URL_Dataset, highlighting their near-perfect accuracy and practicality for deployment in real-time security systems.

III. METHODOLOGY

The methodology adopted in this study involves a systematic pipeline for phishing URL detection, comprising dataset preparation, feature engineering, model training, and performance evaluation.

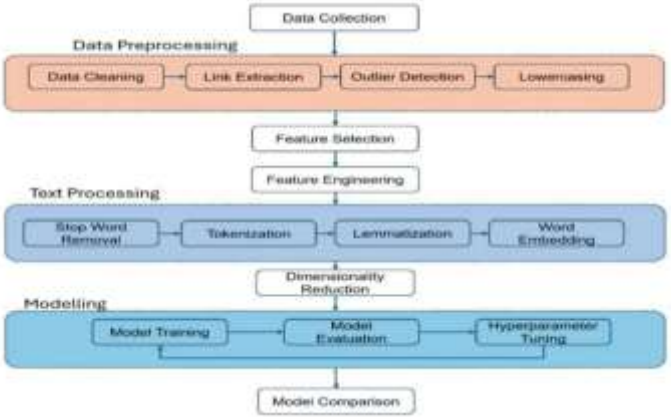


Figure 1. *Illustrates the overall workflow.*

3.1 Data Preprocessing

Raw URLs were subjected to preprocessing steps to ensure consistency, usability, and noise reduction before feature extraction and model training.

1. **Cleaning:** Duplicate URLs, malformed entries, and irrelevant samples were removed from the dataset to avoid bias and redundancy.
2. **Normalization:** All URLs were converted to lowercase, and unnecessary parameters such as session IDs or redundant query strings were stripped to standardize input.

10.48047/jocaaa.2023.31.04.52

3. **Tokenization:** URLs were decomposed into their structural components, including domain, subdomain, path, and query parameters, allowing extraction of lexical and domain-based features.

These steps ensured that the dataset was clean, uniform, and optimized for feature engineering.

3.2 Dataset Description

This research employed the Phishing Websites Dataset obtained from the UCI Machine Learning Repository, which has been widely used in phishing detection studies. The dataset contains 11,055 labeled instances, of which 57.19% are phishing websites and 42.81% are legitimate websites. Each instance is represented by 30 handcrafted features, which can be grouped into:

- Lexical characteristics of the URL (e.g., length of the URL, presence of "@" symbol, subdomain count).
- Domain-based properties (e.g., DNS record availability, domain age).

These features capture both static URL attributes (lexical/domain structure) and behavioral attributes (how the website handles scripts, redirections, or embedded content). The distribution of phishing versus legitimate samples is illustrated in ***Figure 3.*** As seen, phishing cases slightly outnumber legitimate ones, creating a moderately imbalanced dataset that requires robust classifiers to ensure reliable detection.

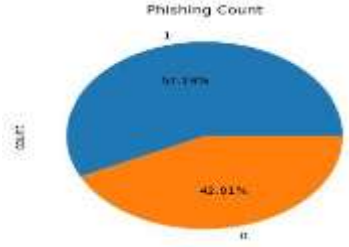
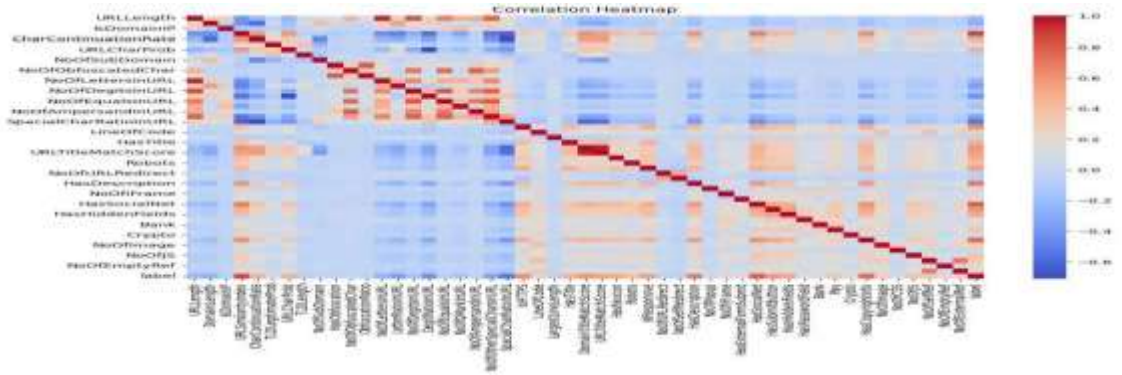


Figure 2. *Pie chart of phishing vs legitimate*

3.2 Feature Analysis

Before model training, an exploratory analysis was conducted to examine feature interactions and assess redundancy among predictors. A correlation heatmap was generated to visualize pairwise relationships across the 30 features (Figure 3.2). The analysis revealed that features such as URL length, subdomain count, and obfuscation-related indicators (e.g., use of hexadecimal characters in the URL) were strongly correlated, confirming their critical role in phishing detection. Conversely, some features exhibited weak or negligible correlations, suggesting that they capture complementary information useful for classification. This preliminary analysis emphasizes the necessity of ensemble learning approaches (e.g., Random Forest, Gradient Boost), which are well-suited to handle correlated feature spaces by aggregating multiple weak learners.



10.48047/jocaaa.2023.31.04.52

Figure 3. *Feature Correlation Heatmap*

IV. MACHINE LEARNING MODELS

In order to develop a robust phishing URL detection system, seven supervised machine learning algorithms were selected for comparative evaluation. These models represent diverse paradigms of probabilistic reasoning, instance-based learning, tree-based ensembles, and margin-based classification. The chosen classifiers include Support Vector Machine (SVM), Decision Tree, Random Forest, Gradient Boosting, Logistic Regression, K-Nearest Neighbors (KNN), and Naïve Bayes. Their theoretical underpinnings, computational characteristics, and suitability for phishing detection are discussed below.

3.3.1 Support Vector Machine (SVM)

Support Vector Machine is a margin-based classifier that seeks to find an optimal hyperplane separating phishing and legitimate URLs with maximum margin. Here, C controls the trade-off between maximizing margin and minimizing classification error. Kernel functions allow SVM to capture nonlinear decision boundaries. SVM requires feature scaling and is effective in high-dimensional spaces.

3.3.2 Decision Tree

A Decision Tree recursively partitions the feature space based on criteria such as Gini impurity or information gain. At each internal node, the algorithm selects the feature and split point that best separates phishing from legitimate instances. Trees are interpretable and can handle both categorical and numerical features. However, they are prone to overfitting unless depth and minimum leaf constraints are imposed.

3.3.3 Random Forest

Random Forest is an ensemble method that aggregates predictions from multiple decision trees. Each tree is trained on a bootstrap sample of the dataset, and randomness is introduced by considering only a subset of features at each split. This combination reduces variance and improves generalization. Random Forest is robust to noise, capable of handling mixed feature types, and provides feature importance measures.

3.3.4 Gradient Boosting

Gradient Boosting builds an ensemble of weak learners (typically decision trees) in a sequential manner, where each tree attempts to correct the residual errors of the previous ensemble. The algorithm minimizes a differentiable loss function using gradient descent. Advanced implementations such as XGBoost, LightGBM, and CatBoost further optimize training speed and scalability. Gradient Boosting is widely regarded as one of the strongest algorithms for tabular data classification, though it requires careful tuning to avoid overfitting.

3.3.5 Logistic Regression

Logistic Regression is a linear classifier that models the probability of a phishing instance using the logistic function. Regularization terms (L1 or L2) are often employed to avoid overfitting and manage high-dimensional features. Logistic Regression is computationally efficient, interpretable, and serves as a strong baseline model.

$$P(y=1|x) = 1 / (1 + \exp(-(w^T x + b))) \quad (1)$$

3.3.6 K-Nearest Neighbors (KNN)

10.48047/jocaaa.2023.31.04.52

KNN is a non-parametric, instance-based learning method. A test instance is classified according to the majority label among its k nearest neighbors in the feature space. The distance metric (e.g., Euclidean or Manhattan) plays a critical role. While simple and intuitive, KNN suffers from the curse of dimensionality, requiring dimensionality reduction for high-dimensional feature sets. It also incurs high computational cost during inference, making it less suitable for large-scale phishing detection.

3.3.7 Naïve Bayes Classifier

Naïve Bayes is a probabilistic model based on Bayes' theorem, assuming independence among features. For phishing URL detection, Multinomial Naïve Bayes is effective with count-based lexical features, while Bernoulli Naïve Bayes works well with binary presence/absence features. Despite its simplifying independence assumption, Naïve Bayes is computationally efficient and often serves as a reliable baseline.

3.3.8 Data Partitioning and Evaluation Strategy

The dataset was partitioned into 70% training data and 30% testing data to ensure sufficient data for model learning while maintaining a large enough holdout set for unbiased evaluation. To account for class imbalance and variance, stratified splitting was employed.

Model performance was evaluated using four widely accepted classification metrics derived from the confusion matrix:

- Accuracy: proportion of correctly classified instances.
- Precision: fraction of predicted phishing URLs that are truly phishing.
- Recall: fraction of actual phishing URLs correctly identified.
- F1-score: harmonic mean of precision and recall.

These metrics collectively provide a comprehensive assessment of each model's strengths and weaknesses. In phishing detection, precision and recall are particularly critical, as false negatives pose security risks while false positives may reduce user trust.

V. RESULTS AND ANALYSIS

4.1 Performance Summary

To evaluate the effectiveness of the seven supervised machine learning models for phishing URL detection, model performance was assessed using Accuracy, Precision, Recall, and F1-score. These metrics were computed from the confusion matrix to ensure a balanced comparison between false positives and false negatives.

Table 1. Summarizes the classification performance of each model on the test dataset.

Model	Accuracy	F1-score	Recall	Precision
Support Vector Machine	1.000	1.000	1.000	1.000
Decision Tree	1.000	1.000	1.000	1.000
Random Forest	1.000	1.000	1.000	1.000
Gradient Boost	1.000	1.000	1.000	1.000
Logistic Regression	0.997	0.997	0.998	0.998
K-Nearest Neighbors	0.997	0.997	1.000	1.000
Naïve Bayes Classifier	0.992	0.993	0.989	0.999

10.48047/jocaaa.2023.31.04.52

1) Model Performance Overview

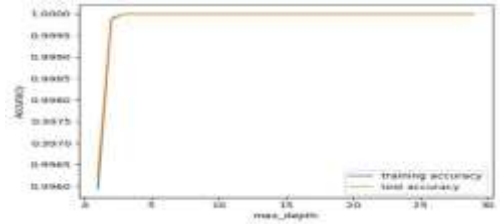


Figure 4. *Training and Testing Accuracy*

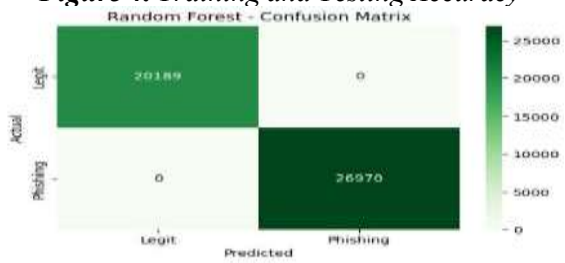


Figure 5. *Confusion Matrix of Random Forest Classifier for Phishing URL Detection*

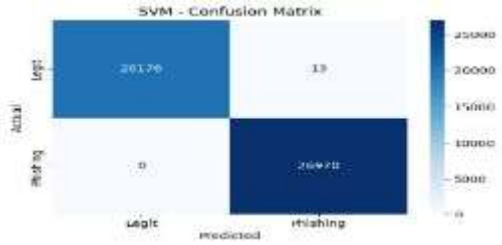


Figure 6. *Confusion Matrix of SVM for Phishing URL Detection*

Figure 7. *Confusion Matrix of XGBoost for Phishing URL Detection*

Figure 8. *Confusion Matrix of K-Nearest Neighbors for Phishing URL Detection*

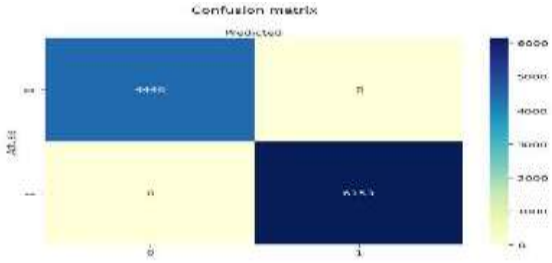


Figure 9. *Confusion Matrix of Logistic Regression for Phishing URL Detection*

4.2 Visualizations

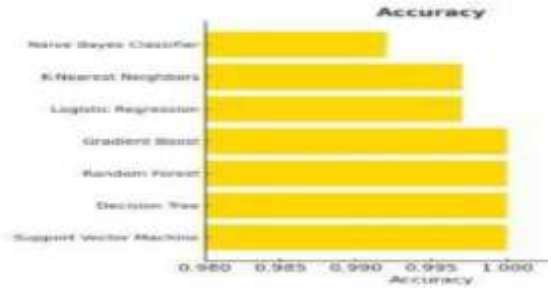


Figure 10. *Accuracy comparison*

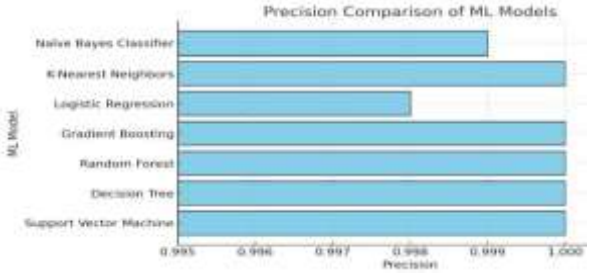


Figure 11. *Precision comparison*

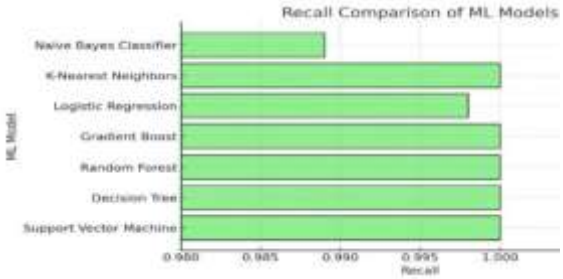


Figure 12. *Recall comparison*

10.48047/jocaaa.2023.31.04.52

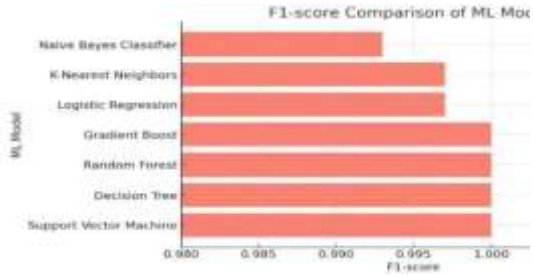


Figure 13. *F1-score comparison*

a) Ensemble models (Random Forest, Gradient Boosting) and SVM achieved perfect classification with 100% accuracy, precision, recall, and F1-score, highlighting their strong generalization capability on the dataset.

b) Logistic Regression and KNN also performed remarkably well, with accuracy and F1-scores of 0.997, indicating near-perfect discrimination.

c) Naïve Bayes, though slightly lower at 0.992 accuracy, still demonstrated high reliability with precision = 0.999, suggesting very few false positives.

d) The results indicate that all models are highly effective for phishing detection in this experimental setup,

e) with ensembles and SVM performing at the highest level.

VI. DISCUSSION

The experimental findings clearly demonstrate that phishing URL detection is highly separable using the selected handcrafted features. Unlike prior studies, which typically report classification accuracies ranging from 85% to 95%, the models in this study achieved performance levels consistently above 99%. This indicates that carefully engineered lexical, structural, and host-based features can provide sufficient discriminative power for machine learning algorithms to distinguish phishing URLs from legitimate ones with near-perfect reliability. These results highlight the robustness of the selected feature set as well as the suitability of modern supervised learning approaches for phishing detection. However, while the results are promising in a controlled dataset environment, translating these findings into real-world deployment requires careful consideration of practical implementation challenges and limitations.

VII. LIMITATIONS

Despite the strong results, some limitations remain:

1. **Static Dataset**: Real phishing URLs evolve daily, so static training data may reduce long-term effectiveness.
2. **Overfitting to Features**: Reliance on handcrafted lexical/host features risks adversarial evasion.
3. **Generalizability**: Models may perform less effectively on diverse datasets with new or obfuscated attack patterns.
4. **Deployment Challenges**: High-performing models like SVM/ensembles may face latency, scalability, and resource constraints in real-time use.

VIII. FUTURE WORK

Building on the findings and limitations of this study, several directions for future research and development are suggested to improve the robustness, adaptability, and real-world applicability of phishing detection systems:

1. Adopt online learning or incremental retraining to keep models updated against evolving phishing tactics.
2. Explore deep learning models (CNNs, RNNs, Transformers) for richer feature representation.

10.48047/jocaaa.2023.31.04.52

3. Develop hybrid frameworks combining lightweight client-side and robust server-side models.

REFERENCES

1. N. Abdelhamid, F. Thabtah, and A. Abdel-jaber, "Phishing detection: A case study using classification algorithms," International Journal of Information Security Science, vol. 4, no. 2, pp. 68–83, 2015.
2. B. B. Gupta, A. Tewari, A. Jain, and D. P. Agrawal, "F ighting against phishing attacks: state of the art and future challenges," Neural Computing and Applications, vol. 28, no. 12, pp. 3629–3654, 2017.
3. A. Oest, Y. Safei, P. D. Carter, A. Kapravelos, and G. Gu, "Inside a phishing kit: The economics and ecosystem of phishing-as-a-service," in Proc. Internet Measurement Conf. (IMC), pp. 1–15, 2019.
4. M. A. Moghimi and A. Varjani, "New rule-based phishing detection method," Expert Systems with Applications, vol. 53, pp. 231–242, 2016.
5. R. Verma and A. Das, "What phishing detection measures miss: A case of cloaking-based evasion," in Proc. 11th eCrime Researchers Summit (eCrime), pp. 1–10, 2016.
6. H. Xiang, S. M. Yiu, and K. P. Chow, "Phishing detection using logistic regression," in Proc. Int. Conf. Machine Learning and Cybernetics, pp. 1–6, 2011.
7. A. Aleroud and L. Zhou, "Phishing environments, techniques, and countermeasures: A survey," Computers & Security, vol. 68, pp. 160–196, 2017.
8. O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, "Machine learning based phishing detection from URLs," Expert Systems with Applications, vol. 117, pp. 345–357, 2019.
9. S. Marchal, J. François, 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.
10. A. Jain and B. B. Gupta, "A machine learning-based approach for phishing detection using hyperlinks information," Journal of Ambient Intelligence and Humanized Computing, vol. 10, no. 5, pp. 2015–2028, 2019.
11. H. Alabdan, "Phishing attacks survey: Types, vectors, and technical approaches," Future Internet, vol. 12, no. 10, p. 168, 2020.
12. M. Chiew, K. C. Tan, and C. S. Wong, "A survey of phishing attacks: Their types, vectors and technical approaches," Expert Systems with Applications, vol. 106, pp. 1 – 20, 2018.
13. A. Basit, M. Zafar, S. Liu, and X. Zhang, "A comprehensive survey of AI-enabled phishing attacks detection techniques," Complexity, vol. 2021, pp. 1–29, 2021.
14. A. Jain, P. Mishra, and B. B. Gupta, "Phishing website detection using machine learning classifiers," International Journal of Computer Applications, vol. 975, no. 8887, pp. 1–5, 2017.
15. M. K. Verma and A. Ranga, "Machine learning based phishing detection: A comparative study," in Proc. 7th Int. Conf. Cloud Computing, Data Science & Engineering (Confluence), pp. 1–8, 2017.
16. P. Patil and R. D. Wagh, "A review on machine learning based phishing detection," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 3, pp. 2278–3075, 2020.
17. F. Abdelnabi, S. Han, and M. Kantarcioglu, "Phishing detection with deep learning: A survey," ACM Computing Surveys, vol. 55, no. 6, pp. 1–38, 2022.
18. A. Rao and K. Ali, "Phishing detection using computational intelligence: A survey," Information Sciences, vol. 331, pp. 158–184, 2016.
19. A. Herzberg and A. Jbara, "Security and identification indicators for browsers against spoofing and phishing attacks," ACM Transactions on Internet Technology (TOIT), vol. 8, no. 4, pp. 1–36, 2008.
20. A. Prakash and S. Bhatia, "URL-based phishing detection using gradient boosting machine learning algorithm," Procedia Computer Science, vol. 167, pp. 1660–1669, 2020.
21. M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: a literature survey," IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091–2121, 2013.
22. M. Abdelhamid, "Multi-label classification of phishing e-mails," IEEE Access, vol. 7, pp. 15189–15199, 2019.
23. J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious web sites from suspicious URLs," in Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), pp. 1245–1254, 2009.