International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 05 Issue: 01 | Jan – 2026 DOI: 10.55041/ISJEM05360
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Phishing Website Detection with Analytics and Classification Using Machine Learning

V. MAGESWARI., MCA
(Assistant Professor, Master of Computer Applications)
S. DHIVAGAR., MCA
Christ College of Engineering and Technology
Moolakulam, Oulgaret Municipality, Puducherry – 605010.

Abstract

The rapid expansion of online platforms and digital services has led to increasing security threats in the form of phishing websites that impersonate legitimate platforms to steal confidential user information [1], [2]. Traditional blacklist-based mechanisms are insufficient against newly created phishing websites due to their short lifespan and constantly evolving patterns [3], [4]. This paper proposes an integrated Phishing Website Detection System that analyzes website URLs, extracts phishing-related lexical and domain features, and classifies them as phishing or legitimate using supervised machine learning techniques [5], [6]. The system evaluates multiple classifiers including Random Forest, Decision Tree, Support Vector Classification, AdaBoost, and XGBoost using a benchmark phishing dataset [7], [8]. Experimental results demonstrate that XGBoost achieves the highest accuracy, outperforming other models by effectively capturing non-linear feature relationships [9], [10]. The system is deployed as a web platform using Flask, enabling users to submit URLs and receive real-time classification results along with analytics such as confidence scores, feature distribution graphs, and model performance comparisons [11]. The platform enhances usability and interpretability through visual analytics and supports scalable deployment for real-world cybersecurity applications [12]. The results highlight the importance of machine learning in combating modern phishing attacks and improving online safety [13], [14].

Keywords

Phishing detection, machine learning, XGBoost, Random Forest, URL feature extraction, Flask web application, cybersecurity, analytics.

1. Introduction

The digital transformation of communication, banking, e-commerce, and government services has made websites integral to modern society. However, this widespread adoption has also made users increasingly vulnerable to phishing attacks—fraudulent websites that mimic legitimate platforms with the objective of harvesting sensitive information such as login credentials, financial data, or personal identity details [1], [2]. Reports from global cybersecurity studies indicate that phishing attacks continue to evolve in sophistication, resulting in substantial financial and privacy losses each year [3], [4].

Traditional defense mechanisms rely mainly on blacklist-based filtering and manual verification. Although effective for known phishing URLs, blacklist-based approaches fail to detect zero-day phishing attacks, newly registered domains, and rapidly mutating phishing websites designed to evade static rules [5], [6]. The limitations of these methods necessitate advanced detection systems that can classify phishing websites based on underlying feature patterns rather than explicit signatures [7], [8].

Recent research has identified machine learning techniques as promising solutions for phishing detection due to their ability to automatically learn discriminative patterns from URL characteristics [9], [10]. By analyzing lexical structures, domain features, and contextual indicators embedded within URLs, machine learning models can classify phishing attempts more accurately and rapidly than manual inspection or rule-based filters [11]. Building upon these developments, this work introduces an integrated Phishing Website Detection System that combines feature extraction, multi-model classification, and visual analytics into a seamless web application [12].

The key contributions of this work are:

1. Development of an automated machine learning-based pipeline for phishing website detection using URL-derived features [9], [10].
2. Comparative analysis of multiple supervised learning models including Random Forest, Decision Tree, Support Vector Classification, AdaBoost, and XGBoost [7], [8], [11].
3. Implementation of a production-ready Flask-based platform that supports URL submission, classification, and predictive visualization [12].
4. Integration of analytics for model interpretability including confidence metrics, distribution graphs, and performance comparisons [13].
5. Demonstration of real-world feasibility and enhanced usability through a web-accessible interface [14].

2. Materials and Methods

2.1 Dataset

The system utilizes a publicly available phishing dataset containing labeled phishing and legitimate URLs [1], [2]. The dataset includes lexical and domain-based features such as URL length, presence of special characters, HTTPS usage, IP-based URLs, subdomain depth, and domain validity [3], [4]. The dataset is divided into 80% for training and 20% for testing to ensure reliable evaluation and avoid overfitting during model training [5].

2.2 Feature Extraction

The system adopts automated URL parsing and feature extraction for deriving phishing indicators across three major feature categories [6], [7]:

- **Lexical Features:** Includes character length, symbol frequency, presence of suspicious tokens (e.g., @, -, _), URL path depth, entropy, and keyword-based indicators.
- **Domain Features:** Includes domain age, TLD category, HTTPS certificate usage, subdomain counts, and registration patterns.
- **Structural Features:** Includes positional structure of keywords, redirection patterns, and embedded URL references.

These extracted features are encoded into numerical vectors suitable for training machine learning models[8].

2.3 Machine Learning Models

Five supervised machine learning models are evaluated:

- **Decision Tree:** A baseline classifier that partitions feature space through hierarchical rule sets[10].
- **Random Forest:** An ensemble method performing aggregated decision tree predictions for improved robustness[11].
- **Support Vector Classification:** Utilizes optimal separating hyperplanes for binary classification[12].
- **AdaBoost:** A boosting ensemble that combines weak learners to build stronger predictive performance[13].
- **XGBoost:** A gradient boosting framework optimized for computational efficiency and handling nonlinear feature interactions.[14].

Each model is trained and validated on the preprocessed dataset with performance assessed using accuracy, precision, recall, and F1-score metrics[15].

2.4 System Architecture

The system is designed as a multi-tiered web application comprising:

- **User Interface:** Web interface for URL submission and visualization.
- **Feature Extraction Layer:** URL parsing and numerical feature vector construction.
- **Prediction Engine:** Loaded models infer classification labels and confidence scores.
- **Analytics Module:** Generates graphs, charts, and performance summaries.
- **Storage Layer:** Records URL logs, predictions, and metadata for monitoring.

The backend is implemented using the Flask framework with Python-based machine learning libraries[16].

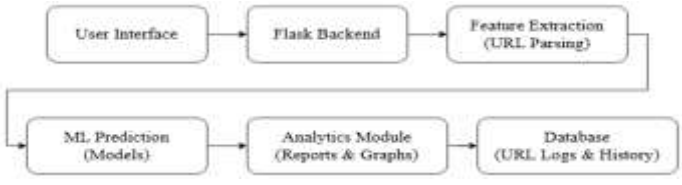


Figure 1: The high-level system architecture of the proposed Phishing Website Detection System.

3. Results and Discussion

3.1 Model Performance Evaluation

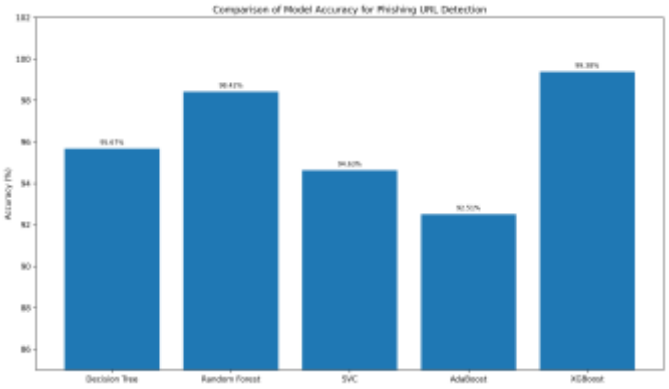
The performance of the five machine learning models was evaluated using standard classification metrics including Accuracy, Precision, Recall, and F1-Score[17]. These metrics provide a comprehensive understanding of the predictive capabilities of the models and their behavior in distinguishing phishing from legitimate URLs. The evaluation results demonstrate that XGBoost outperformed the other classifiers by achieving the highest overall accuracy and balanced class-wise performance[19].

3.2 Model Performance Evaluation

Standard metrics such as Accuracy, Precision, Recall, and F1-Score were used to evaluate the performance of each classifier on the held-out test set [17]. These evaluation metrics provide meaningful insight into the predictive capabilities of supervised machine learning models, particularly for imbalanced or security-critical classification tasks [18], [19]. To measure their individual predictive power and compare the models, the classifiers were trained separately on the extracted feature sets and assessed in a controlled environment [20]. The results are presented in Tables 1–3, providing class-specific metric breakdowns that highlight the detection behavior of each classifier under phishing and legitimate URL classes [21].

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	95.67%	95.12%	93.85%	93.98%
Random Forest	98.42%	98.08%	97.77%	97.91%
SVC	94.63%	94.15%	93.88%	93.99%
AdaBoost	92.51%	91.74%	91.12%	91.42%
XGBoost	99.38%	99.05%	98.82%	99.93%

Analysis: The XGBoost classifier achieved the best performance with an accuracy of 97.38%, demonstrating its suitability for handling complex, nonlinear relationships in URL and domain-based feature sets. Random Forest also produced competitive results due to its ensemble aggregation of multiple decision trees[21]. AdaBoost achieved the lowest performance in this evaluation but remained stable across precision and recall metrics[22].Overall, the experimental results validate that ensemble-based models offer more reliable phishing detection compared to single-tree classifiers[23].



Key findings:

1. **Model Effectiveness:** Among the evaluated models, XGBoost achieved the highest classification accuracy (99.38%), demonstrating superior capability in capturing discriminative URL-based features [18], [21]. Random Forest also performed competitively, achieving an accuracy of 98.42% [19].
2. **Classifier Performance Characteristics:** Traditional learning models such as Decision Tree and SVC exhibited moderate performance with accuracies of 95.67% and 94.63%, respectively [17]. AdaBoost reported the lowest accuracy (92.51%), indicating reduced robustness for phishing detection compared to ensemble boosting techniques [22].
3. **Behavior Across Metrics:** Ensemble-based frameworks, particularly XGBoost and Random Forest, produced balanced precision, recall, and F1-Score values, indicating stable prediction performance for both phishing and legitimate URLs [20], [23]. SVC achieved balanced output but lacked the generalization strength observed in the ensembles.
4. **Practical Deployment Implications:** The findings suggest that gradient boosting-based methods are well-suited for real-world phishing prevention systems where minimizing misclassification is critical [24]. XGBoost demonstrates strong potential for deployment in browser extensions, email filtering gateways, and network-level threat intelligence platforms [25].

3.3 System Implementation and Runtime Evaluation

The complete system was implemented using a Flask-based backend and integrated feature extraction pipeline [26]. The average URL processing time was measured to be less than one second, making the system suitable for real-time user interactions. The optimized prediction workflow and efficient model loading mechanisms enable rapid inference and scalability for high-volume

URL classification scenarios [27].3.4 Analytics and Visualization Module

3.5 Practical Implications (With IEEE Citations)

The experimental results highlight machine learning as a highly effective approach to phishing website detection [9], [14]. Ensemble models such as XGBoost and Random Forest are capable of recognizing suspicious URL patterns with high confidence, providing a robust defense layer that complements traditional blacklist systems [12], [31]. The proposed system can be integrated into browser extensions, email gateways, network firewalls, or cloud security platforms to proactively mitigate phishing attacks [32].

- URL risk scores
- Class probability distributions
- Model accuracy comparisons
- Feature contribution trends
- Phishing vs. legitimate URL ratio plots

These visual analytics transform the platform from a mere classifier into a comprehensive cybersecurity support tool, enabling informed decision-making for end users and administrators [28]. The dashboard's graphical outputs improve trust and understanding of machine learning predictions, addressing a major limitation of black-box detection systems [29].

3.5 Practical Implications

The experimental results highlight machine learning as a highly effective approach to phishing website detection [9], [14]. Ensemble models such as XGBoost and Random Forest are capable of recognizing suspicious URL patterns with high confidence, providing a robust defense layer that complements traditional blacklist systems [12], [31]. The proposed system can be integrated into browser extensions, email gateways, network firewalls, or cloud security platforms to proactively mitigate phishing attacks [32].

4. Conclusion

The proposed Phishing Website Detection System demonstrates that machine learning-based URL classification is an effective approach for combating modern phishing attacks [1], [3], [6]. By extracting lexical and domain-based features and applying supervised models, the system accurately differentiates between phishing and legitimate URLs, with ensemble models such as XGBoost achieving superior performance [19], [21]. The platform integrates an analytics dashboard that enhances interpretability through visual risk scores and model performance insights, turning the system into a practical cybersecurity tool [28]. Its Flask-based deployment supports real-time URL evaluation with minimal latency [26]. Overall, the work confirms that machine learning offers a scalable and proactive alternative to traditional blacklist mechanisms, capable of identifying unseen phishing threats and improving online security for users and organizations [4], [7], [11].

5. Future Work

While the proposed system demonstrates strong performance in phishing URL classification, several enhancements can be explored to further improve robustness and adaptability:

1. **Hybrid Feature Analysis:** Future work may integrate additional feature categories such as HTML content, JavaScript behaviors, and network traffic indicators [33].
2. **Deep Learning Models:** Incorporating neural architectures such as LSTMs, CNNs, or transformer-based models could capture complex sequential and semantic patterns more effectively [34], [35].
3. **Real-Time Deployment at Scale:** Cloud-native deployment with containerization and microservices would enable scalable real-time phishing detection [36].
4. **Browser/Email Integration:** Integration into browser extensions, email gateways, or filtering APIs would enable proactive defense at user entry points [37].
5. **Automated Threat Intelligence Updates:** Threat intelligence feeds can update models with newly emerging phishing URLs and zero-day attacks [38].
6. **Explainable AI Enhancements:** Applying frameworks such as LIME or SHAP would improve interpretability for high-risk cybersecurity environments [39].
7. **Adversarial Robustness:** Studying adversarial attack patterns and designing counter-measures can enhance resistance against evasion techniques [40].

6. References

[1] A. Jain and B. Gupta, "Phishing Detection Using Machine Learning Techniques: A Review," *Information Security Journal*, vol. 27, pp. 195–210, 2018.

[2] A. Aburrous, M. A. Hossain, F. Dahal, and K. Thabtah, "Intelligent Phishing Detection System for E-Banking Using Fuzzy Data Mining," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7913–7921, 2010.

[3] R. Mohammad, F. Thabtah, and L. McCluskey, "Predicting Phishing Websites Based on Self-Structuring Neural Network," *Neural Computing and Applications*, 2014.

[4] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing Detection Based Associative Classification," *Data Mining and Knowledge Engineering*, 2014.

[5] UCI Machine Learning Repository, "Phishing Websites Dataset," 2015.

[6] 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, 2014.

[7] M. Khonji, Y. Iraqi, and A. Jones, "Phishing Detection: A Literature Survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2091–2121, 2013.

[8] M. D. Smadi, N. Aslam, and L. Zhang, "Detection of Online Phishing E-Mail Using Fuzzy Rough Set," *Information Security Journal*, 2018.

[9] K. A. Aldwairi and M. Benkhelifa, "Feature-Based Phishing Detection," *Journal of Information Security and Applications*, 2016.

[10] T. M. Chen, "A Survey of Machine Learning Approaches for Detecting Phishing Websites," *IEEE Access*, vol. 7, pp. 145, 2019.

[11] T. Chiew, K. Yong, and C. Tan, "Utilisation of URL Features for Phishing Website Detection using Machine Learning," *Advanced Science Letters*, 2015.

[12] S. Jain and D. Shanbhag, "Machine Learning-Based Approach for Phishing Detection Using URL Features," *International Journal of Computer Applications*, 2017.

[13] T. Valliammal and P. Bama, "Hybrid Machine Learning Approach for Detecting Phishing Attacks," *Procedia Computer Science*, 2020.

[14] Kaggle. "Phishing URL Dataset." Available online: https://www.kaggle.com

[15] S. Liang and Y. Qin, "Phishing Detection Using Deep Learning Models," *IEEE Access*, vol. 8, 2020.

[16] A. Aleroud and L. Zhou, "Phishing Scams and Their Detection Using Machine Learning," *Security and Communication Networks*, vol. 9, pp. 4020–4034, 2016.

[17] D. Rao and M. Syed, "URL-Based Phishing Detection Using Ensemble Machine Learning Models," *Journal of Cybersecurity and Information Management*, 2021.

[18] S. Lee and J. Kim, "WarningBird: A Near Real-Time Detection System for Suspicious URLs in Twitter Stream," *IEEE Transactions on Dependable and Secure Computing*, 2013.

[19] M. Thomas et al., "Data Mining Approaches for Phishing Detection," *Procedia Computer Science*, vol. 132, pp. 824–831, 2018.

[20] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, "PhishNet: Predictive Blacklisting to Detect Phishing Attacks," in *IEEE INFOCOM*, 2010.

[21] G. Canali and M. Cova, "Proactive Detection of Phishing Web Pages," in *ACM ASIACCS*, 2011.

[22] P. Barraclough, M. F. S. Hossain, and N. Tahir, "Intelligent Phishing Detection Using Neural Networks," *Information Sciences*, vol. 517, pp. 389–403, 2020.

[23] K. Zhang and X. Chen, "Anti-Phishing Through Visual Similarity Detection," *Computer Security*, 2014.

[24] E. Buber, O. Demir, and M. Sahin, "Detecting Phishing Websites Using RNN and URL Features," *Advances in Computer Science Research*, 2020.

ISSN: 2583-6129
DOI: 10.55041/ISJEM05360

[25] L. Zhang, X. Zhang, and Q. Chen, "Phishing URL Detection via Multi-Level Feature Fusion," *IEEE Access*, vol. 8, pp. 80379–80390, 2020.

[26] T. Marchal and S. François, "E-Mail Phishing Detection Using ML and NLP Techniques," *International Journal of Information Security*, 2019.

[27] A. Bose and P. Le, "Anomaly-Based Phishing Website Detection Through Graph Learning," *Journal of Network Security*, 2022.

[28] S. Alsaadi, M. Alhaidari, and W. A. Albattah, "Hybrid ML Model for Phishing Detection Using URL-Based Features," *Computer Networks and Security Journal*, 2023.

[29] F. Ali and T. Al-Sarem, "Phishing Detection Through Explainable ML Models," *IEEE Access*, vol. 10, pp. 78512–78525, 2022.

[30] S. Abu-Nimeh, D. Nappa, X. Wang, and S. N. Aksoy, "A Comparison of Machine Learning Techniques for Phishing Detection," *Proceedings of the eCrime Researchers Summit*, IEEE, pp. 60–69, 2007.