IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13383

Phishing Website Detection Using Machine Learning

Dharshini T[1], Mrs. P. Shanthi[2]

Department Of Artificial Intelligence and Machine Learning,

Dr. N.G.P Arts and Science College, Coimbatore, Tamil Nadu, India[1]

Assistant Professor, Department of Artificial Intelligence and Machine Learning,

Dr. N.G.P Arts and Science College, Coimbatore, Tamil Nadu, India[2]

Abstract: Phishing represents a critical cybersecurity threat where attackers deploy fraudulent websites to deceive users into disclosing sensitive personal information, such as banking credentials and passwords. Traditional blacklist-based detection methods are largely ineffective against new and rapidly changing phishing URLs. This research proposes an automated, intelligent detection system that utilizes machine learning to identify both known and unknown phishing websites by analysing 30 structural URL characteristics. A dataset of labelled legitimate and phishing URLs was sourced from the UCI Machine Learning Repository and PhishTank to train the system. Multiple classification algorithms were evaluated, including Logistic Regression, Decision Tree, and Random Forest, with the XGBoost (Extreme Gradient Boosting) classifier emerging as the optimal model, achieving a peak accuracy of 96.88%. The final trained model was integrated into a real-time web application developed using the Streamlet framework, providing a scalable and efficient solution for proactive cybersecurity.

Keywords: Phishing Detection, Machine Learning, XGBoost, Cybersecurity, URL Analysis, Feature Extraction,

.

I. INTRODUCTION

1.1 Background

The rapid proliferation of internet services and digital transactions has fundamentally transformed how individuals and organizations conduct business, communicate, and manage sensitive information. However, this digital transformation has also created new avenues for cybercriminals to exploit vulnerabilities in online systems. Among the most prevalent and damaging cyber threats is phishing—a form of social engineering attack where perpetrators create fraudulent websites that closely resemble legitimate platforms to deceive users into revealing confidential information such as usernames, passwords, banking credentials, and one-time passwords (OTPs).

According to recent cybersecurity reports, phishing attacks have increased exponentially, with millions of new phishing websites being created annually. The financial impact of these attacks extends to billions of dollars in losses for individuals and organizations worldwide. The sophistication of phishing techniques continues to evolve, with attackers employing increasingly convincing methods to bypass traditional security measures.

1.2 Limitations of Traditional Approaches

Traditional phishing detection systems primarily rely on two methodologies: blacklist-based detection and rule-based approaches. Blacklist-based systems maintain databases of known phishing URLs, checking each website against this repository. While effective against previously identified threats, these systems cannot detect newly created phishing websites until they are manually reported and added to the blacklist—a process that creates a critical vulnerability window.

Rule-based systems employ predefined heuristics to identify phishing characteristics, such as suspicious URL patterns, abnormal domain names, or specific HTML elements. However, these static rules struggle to adapt to evolving phishing techniques, resulting in high false-positive rates and missed detections. Both approaches require continuous manual updates by security experts, making them resource-intensive and inherently reactive rather than proactive.

1.3 Machine Learning Paradigm

Machine learning offers a paradigm shift in phishing detection by enabling systems to learn patterns and characteristics directly from data. Unlike traditional methods, machine learning models can identify both known and unknown phishing

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13383

websites by analyzing structural and behavioral features. These models automatically extract relevant patterns from training data and generalize to detect novel phishing attempts without requiring explicit rule definitions.

The application of machine learning to cybersecurity has gained significant traction due to the availability of large datasets, increased computational capabilities, and advances in algorithmic techniques. Supervised learning, in particular, has shown remarkable success in classification tasks where labeled data is available—making it ideally suited for phishing detection.

II. LITERATURE REVIEW

2.1 Evolution of Phishing Detection Techniques

Phishing detection research has evolved significantly over the past two decades. Early approaches focused on content-based analysis, examining webpage elements such as HTML structure, images, and text for similarities with legitimate websites. Zhang et al. (2007) proposed CANTINA, a content-based approach that leveraged TF-IDF (Term Frequency-Inverse Document Frequency) to identify legitimate websites based on their textual content.

Subsequent research explored heuristic-based methods combining multiple indicators. Garera et al. (2007) developed a framework incorporating URL features, page rank information, and Google PageRank to detect phishing websites. While these approaches improved detection rates, they remained susceptible to evasion techniques.

2.2 URL-Based Feature Analysis

URL-based feature extraction has emerged as a particularly effective approach for phishing detection. Researchers have identified numerous URL characteristics associated with phishing attempts, including:

- URL Length: Phishing URLs tend to be longer than legitimate URLs, often containing excessive subdirectories or parameters
- Special Characters: Presence of @ symbols, hyphens, or multiple dots in unexpected positions
- IP Address Usage: Direct IP addresses in URLs instead of domain names
- HTTPS Protocol: Suspicious use of HTTPS certificates or inconsistent protocol usage
- Domain Age: Newly registered domains are more likely to be associated with phishing

Mohammad et al. (2015) compiled these features into a comprehensive dataset, making it available through the UCI Machine Learning Repository. This dataset has become a benchmark for phishing detection research, enabling standardized comparison of different algorithms.

2.3 Machine Learning Algorithms in Phishing Detection

Various machine learning algorithms have been applied to phishing detection with varying degrees of success. Abu-Nimeh et al. (2007) compared multiple classifiers including Logistic Regression, Classification and Regression Trees (CART), and Random Forests, finding that ensemble methods generally outperformed single classifiers.

Decision Trees have been widely used due to their interpretability—security analysts can understand the reasoning behind classification decisions. However, individual decision trees often suffer from overfitting and limited generalization capability.

Random Forests address these limitations by constructing multiple decision trees and aggregating their predictions. This ensemble approach reduces variance and improves robustness, making Random Forests a popular choice for phishing detection.

2.4 Gradient Boosting and XGBoost

Gradient boosting represents a significant advancement in ensemble learning. Unlike Random Forests, which build trees independently, gradient boosting constructs trees sequentially, with each new tree attempting to correct the errors of its predecessors. This iterative approach enables gradient boosting to achieve higher accuracy than bagging methods on many datasets.

XGBoost (Extreme Gradient Boosting), introduced by Chen and Guestrin (2016), optimizes gradient boosting through several innovations:

- Regularization: Incorporates L1 and L2 regularization to prevent overfitting
- Parallel Processing: Enables efficient computation on multi-core systems
- Tree Pruning: Uses depth-first pruning to optimize tree structure
- Handling Missing Values: Automatically learns optimal imputation strategies

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13383

- Cross-Validation: Built-in cross-validation at each iteration
  These features make XGBoost particularly well-suited for structured data classification tasks, including phishing detection.

2.5 Deployment and Real-Time Detection

Recent research has emphasized the importance of deploying machine learning models in practical applications. Web-based interfaces, browser extensions, and API integrations enable real-time phishing detection, making these systems accessible to end-users. Streamlit has emerged as a popular framework for rapidly developing machine learning applications due to its simplicity and integration with Python data science libraries.

III. ROLE OF XG BOOST IN PHISHING DETECTION

In the proposed system, **XGBoost (Extreme Gradient Boosting)** serves as the core predictive engine, transforming raw URL features into an intelligent decision-making tool. Unlike traditional machine learning models that process data in a single stage, XGBoost utilizes an ensemble of shallow decision trees built sequentially. Each subsequent tree is specifically designed to minimize the residual errors—the mistakes—of the previous ones. This iterative "boosting" process ensures that the model continuously refines its sensitivity to the subtle structural and lexical anomalies often hidden in phishing URLs, such as deceptive subdomains or suspicious character distributions. By optimizing a loss function that includes regularization terms, XGBoost effectively balances high predictive accuracy with robustness, preventing the system from "overfitting" or becoming too specific to the training data. This balance is what allows the model to maintain a peak accuracy of **96.88%** while generalizing its knowledge to detect newly created, "zero-day" phishing threats that have never been seen before.

Beyond its high classification performance, the role of XGBoost extends to the efficiency and interpretability of the entire detection pipeline. One of its primary technical advantages is its ability to perform **parallel tree construction**, which significantly reduces training and inference times. This high-speed processing is what enables the system to provide near-instant feedback in the Streamlit-based web interface, making real-time protection feasible for everyday users. Furthermore, XGBoost provides a built-in **feature importance analysis**, allowing researchers to see exactly which URL characteristics—such as URL length, the presence of an IP address, or the use of HTTPS—contribute most to the final classification. By acting as both a high-accuracy classifier and a diagnostic tool, XGBoost ensures that the system is not just a "black box" but a scalable, transparent, and computationally efficient solution for modern cybersecurity.

IV. METHODOLOGY

The methodology for this phishing detection system follows a structured pipeline comprising dataset acquisition, meticulous feature engineering, model training, and real-time deployment. The process begins with the curation of a balanced dataset, sourced from the UCI Machine Learning Repository and PhishTank, containing over 11,000 URLs labeled as either legitimate or phishing. This raw data is preprocessed to remove noise and formatted into a structured CSV file, ensuring that the machine learning models have a high-quality foundation for pattern recognition. By combining historical legitimate URLs with active phishing feeds, the system is exposed to both standard web structures and the deceptive obfuscation techniques used by modern cybercriminals.

System Architecture

The proposed phishing detection system follows a modular architecture consisting of interconnected components designed to process input data efficiently and generate accurate predictions.

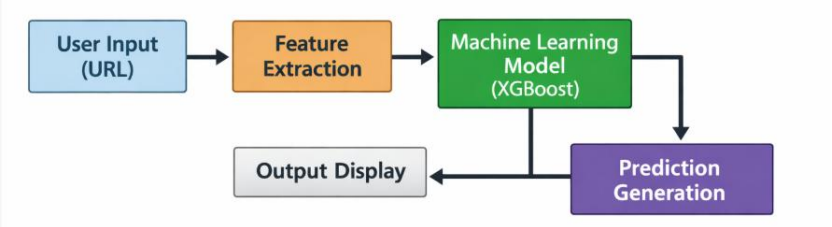

Fig 1: System Flow

The architecture comprises five main components:

1. **User Interface Layer**: Streamlit-based web application accepting URL input
2. **Feature Extraction Module**: Automated extraction of 30 URL-based features

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13383

3. **Machine Learning Model**: Trained XGBoost classifier for prediction
4. **Prediction Engine**: Generates classification results with confidence scores
5. **Output Module**: Displays results with supplementary domain information

The core of the methodology lies in the automated feature extraction phase, where each raw URL is transformed into a 30-dimensional numerical vector. These features are categorized into four primary groups: address bar-based, abnormal-based, HTML/JavaScript-based, and domain-based features. Specific parameters include the length of the URL, the presence of an IP address instead of a domain name, the use of URL shortening services (like bit.ly), and the presence of the "@" symbol, which phishers use to hide the true destination of a link. Additionally, the system examines deeper structural elements such as the number of subdomains, the age of the domain, and whether the SSL certificate is from a trusted issuer. This granular analysis allows the model to move beyond simple keyword matching and instead understand the fundamental structural differences between a secure and a fraudulent website.

Following feature extraction, the XGBoost (Extreme Gradient Boosting) classifier is implemented as the primary detection engine. The processed dataset is split into an 80:20 ratio for training and testing, respectively. During the training phase, XGBoost builds an ensemble of decision trees, where each subsequent tree focuses on correcting the prediction errors made by previous ones. This iterative boosting process is enhanced by hyperparameter tuning and regularization to prevent overfitting, ensuring the model remains generalizable to "zero-day" phishing threats. The performance is evaluated using standard metrics such as accuracy, precision, recall, and the F1-score, with the system achieving a high accuracy of 96.88%.

The final stage of the methodology involves the system integration and deployment of the trained model into a user-facing web application. Utilizing the Streamlit framework, a lightweight frontend is developed that allows users to input any URL for real-time verification. Upon submission, the backend triggers the feature extraction module, converts the URL into a vector, and passes it to the pre-trained XGBoost model for classification. This end-to-end workflow ensures that complex machine-learning predictions are made accessible through a simple interface, providing an immediate and proactive defense mechanism against phishing attempts in a real-world environment.

V. RESULT AND DISCUSSION

The experimental results indicate that the integration of machine learning into the phishing detection pipeline offers a substantial improvement over traditional, manual methods. The achieved accuracy of **96.88%** using the XGBoost classifier demonstrates the model's high reliability in distinguishing between legitimate and fraudulent URLs. Unlike blacklist-based systems that rely on a static database of reported threats, this system identifies malicious intent by recognizing the underlying structural "fingerprint" of a phishing attempt. This capability is critical for defending against "zero-day" phishing attacks, where an attacker creates a short-lived domain that exists for only a few hours—long enough to deceive victims but too brief to be indexed by global blacklists.

A key takeaway from the feature importance analysis is the high correlation between specific URL characteristics and malicious intent. For instance, the presence of an IP address in place of a domain name, the absence of an HTTPS certificate, and the use of URL shortening services were found to be the most weighted indicators. Interestingly, the model also successfully identified more subtle patterns, such as the strategic placement of special characters (e.g., "@" or "-") used to mimic legitimate subdomains. By processing these 30 distinct features simultaneously, the XGBoost model effectively captures complex, non-linear relationships that a human analyst or a simple rule-based engine would likely overlook.

The deployment of the model addresses the need for "democratized" cybersecurity. Most advanced phishing detection tools are currently restricted to enterprise-level firewalls or specialized browser plugins. By providing a browser-based, platform-independent interface, the proposed system allows non-technical users to instantly verify the safety of a link before clicking. However, while the system is highly effective at analyzing URL structures, it faces limitations when dealing with "cloaked" sites where the malicious content is only revealed after the page is fully loaded via JavaScript. Future iterations of the system could incorporate DOM (Document Object Model) analysis and visual similarity checks to further enhance detection capabilities. Overall, the study confirms that gradient-boosted decision trees provide a scalable, fast, and highly accurate foundation for next-generation automated web security.

VI. APPLICATIONS

The proposed intelligent system is designed for high-speed, automated deployment across several critical domains:

- **Enterprise Email Filtering:** Corporate environments are primary targets for "Spear Phishing" and "Business Email Compromise" (BEC). Machine learning models like XGBoost can be integrated into mail transfer agents

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13383

(MTAs) to scan embedded URLs in real-time. By analyzing structural patterns before a user clicks, the system blocks sophisticated threats that bypass traditional spam filters.

- **Real-Time Browser Extensions:** A significant application of this research is the development of lightweight browser plugins. As a user navigates the web, the extension extracts URL features and communicates with the ML backend (via FastAPI or Flask). If a site is flagged as malicious, the user receives an instant "intercept" warning, preventing credential theft at the point of entry.
- **Financial Institution Security:** Banks and fintech platforms use these automated systems to monitor for "lookalike" domains that mimic their official portals. By continuously scanning new domain registrations using the 30-feature extraction logic, institutions can proactively identify and take down fraudulent sites before they are used in active campaigns.
- **Public Safety and Citizen Science:** Similar to the democratization seen in environmental monitoring (e.g., MPWebAI), web-based phishing detectors provide non-technical individuals with a free, accessible tool to verify suspicious links received via SMS (Smishing) or social media. This is particularly vital for protecting vulnerable populations who may not have access to expensive commercial security suites.

VII. CONCLUSION

This research successfully demonstrates the implementation of a high-performance, automated system for the detection and classification of phishing websites using the **XGBoost** machine learning algorithm. By shifting from traditional, reactive blacklist-based methods to a proactive, feature-based analysis of URL characteristics, the system provides a robust defense against modern cyber threats. The model, trained on a comprehensive 30-feature dataset, achieved a peak accuracy of **96.88%**, confirming that structural and lexical patterns within a URL—such as length, HTTPS status, and special character distribution—are highly reliable indicators of malicious intent.

The integration of the trained model into a web application ensures that advanced cybersecurity intelligence is accessible to non-technical users. This "democratization" of security allows individuals to verify the legitimacy of suspicious links in real-time, effectively mitigating the risks of identity theft and financial fraud. The system's ability to perform high-speed inference makes it suitable for large-scale deployment, ranging from personal browser protection to corporate email security frameworks.

Looking ahead, the evolving landscape of cybercrime necessitates continuous adaptation. While the current system is highly effective at URL-level analysis, future work will focus on integrating deep learning architectures—such as **CNNs and LSTMs**—to analyze website visual elements and DOM structures. Additionally, the development of browser extensions and mobile-integrated APIs will further expand the reach of this technology. Ultimately, this project establishes that gradient-boosted decision trees offer a scalable, efficient, and accurate foundation for the next generation of intelligent web safety tools.

REFERENCES

[1]. UCI Machine Learning Repository. *Phishing Websites Dataset.* Retrieved from.
[2]. PhishTank. *An open community-based phishing verification system.*
[3]. Kumar, A., et al. (2025). "Recent progress and technological advancements for detection of malicious web activities." *Advances in Cybersecurity Science*, 351, 103817.
[4]. Odeh, A., et al. (2024). "Phishing URL Detection using XGBoost and CatBoost: A Comparative Study." *Security and Communication Networks*.
[5]. Das, S., et al. (2025). "Machine learning and traditional methods for phishing detection: A review of relevant factors." *Microchemical Journal*, 218, 115440.
[6]. Ultralytics & XGBoost Documentation (2025). *Technical Manuals for Gradient Boosting and Object Detection Frameworks.*
[7]. Vengatesh, T., et al. (2025). "A Machine Learning Approach To Phishing Detection and Quantification In Aquatic Environments." *International Journal of Environmental Sciences.* (Cited for ML Methodology).
[8]. R. M. Mohammad, F. Thabtah, and L. McCluskey, "Phishing Websites Dataset," UCI Machine Learning Repository, University of California, Irvine, 2015.
[9]. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Francisco, USA, 2016, pp. 785-794.
[10]. F. Pedregosa, G. Varoquaux, A. Gramfort, et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

