

Detection of Phishing Websites Using Machine Learning

1st Himanshu Ojha

Computer Science Engineering
Galgotias University
Greater Noida, India
himan9651@gmail.com

2nd Amit Kumar

Computer Science Engineering
Galgotias University
Greater Noida, India
itsamit303@gmail.com

3rd Dr. Ruby Dahiya

Assistant Professor
Galgotias University
Greater Noida, India
ruby.dahiya@galgotiasuniversity.edu.in

Abstract—Phishing attacks have emerged as one of the most serious cybersecurity threats in today's digital environment. Attackers create fake websites that closely resemble legitimate ones in order to trick users into revealing sensitive information such as login credentials, banking details, and personal data. Many users find it difficult to distinguish between genuine and malicious websites, and traditional blacklist-based security mechanisms often fail to detect newly created phishing sites. To address this problem, a Machine Learning-based Phishing Website Detection system has been developed as an intelligent and user-friendly solution. The proposed system analyzes website URLs and domain-based features to automatically classify websites as phishing or legitimate. A browser extension is integrated with the machine learning model to provide real-time protection and instant alerts to users while browsing. The system supports automatic feature extraction, real-time prediction, and fast response. Various machine learning algorithms were evaluated, and XGBoost achieved the highest accuracy. This solution offers a scalable, efficient, and reliable approach for enhancing online security and reducing the risk of phishing attacks through the integration of machine learning and browser-based technologies.

Keywords—Fake News Detection, Machine Learning, NLP, TF-IDF, Multilingual Translation, SVM.

I. INTRODUCTION

With the rapid expansion of internet usage, online services such as e-commerce, digital banking, social media, and cloud-based applications have become an essential part of everyday life. However, the increasing dependence on online platforms has also led to a rise in cyber threats, particularly phishing attacks. Phishing websites are fraudulent websites designed to imitate legitimate ones in order to deceive users into sharing confidential information. These attacks often result in financial loss, identity theft, and data breaches.

Identifying phishing websites is challenging for most users, as attackers continuously modify website appearance and URLs to evade detection. Traditional security techniques, such as blacklists and rule-based systems, are time-consuming and ineffective against newly generated phishing websites. Users often lack awareness of suspicious URL patterns and security indicators. As a result, there is a strong need for an automated and intelligent detection system that can identify phishing websites accurately and in real time.

Machine learning provides an effective solution to this problem by learning patterns from previously identified phishing data and detecting new threats automatically. Integrating such detection mechanisms with browser extensions enables direct user interaction and instant warnings, making phishing prevention more practical and user-oriented.

II. LITERATURE REVIEW

Recent research in cybersecurity has highlighted the importance of using machine learning techniques to combat phishing attacks. Phishing detection systems play a vital role in protecting users from online fraud, as phishing websites are often short-lived and difficult to detect using traditional approaches. Several studies have focused on URL-based, content-based, and hybrid techniques for phishing detection.

Sharma et al. (2025) proposed a machine learning-based phishing website detection approach using algorithms such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naive Bayes. Their results demonstrated that machine learning models can effectively detect phishing websites when trained on appropriate URL and hyperlink features. However, their system was limited to offline evaluation and lacked real-time deployment.

Naseeb et al. (2025) introduced a meta-learning ensemble-based approach that combined multiple classifiers to improve detection accuracy. Although their model achieved high accuracy, it required high computational resources and frequent retraining, making real-time implementation difficult.

Other researchers have compared algorithms such as Random Forest, Decision Tree, and SVM for phishing detection and found ensemble techniques to perform better. More recent studies suggest that gradient boosting models such as XGBoost offer superior accuracy and robustness. Despite these advances, many existing systems focus mainly on theoretical accuracy and do not provide user-friendly real-time protection. This project addresses this gap by integrating a highly accurate machine learning model with

a browser extension to protect users during live browsing session.

III. METHODOLOGY / SYSTEM ARCHITECTURE

The Phishing Website Detection system was developed using a structured and systematic approach. The initial phase involved an in-depth analysis of phishing attack patterns and the challenges faced by users in identifying malicious websites. Based on this analysis, a machine learning-based solution was designed to automatically detect phishing websites using URL and domain-based features.

The system follows a modular architecture consisting of URL collection, feature extraction, machine learning prediction, and browser extension integration. Feature extraction plays a critical role in the system, where important characteristics such as URL length, presence of HTTPS, suspicious symbols, number of subdomains, and usage of IP addresses are extracted. These features are then processed and fed into machine learning models for training and classification.

Multiple supervised learning algorithms were implemented and evaluated, including Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Multi-Layer Perceptron, and XGBoost. Among these, XGBoost demonstrated the highest accuracy and reliability and was selected as the final model.

To provide real-time protection, the trained model was integrated with a browser extension. Whenever a user visits a website, the extension captures the URL and sends it to the backend server for analysis. The machine learning model predicts whether the website is safe or phishing, and the result is immediately displayed to the user. This methodology ensures fast detection, ease of use, and enhanced online security.

IV. DATA COLLECTION

The Phishing Website Detection system collects and utilizes structured data obtained from publicly available and reliable online sources for training and evaluating the machine learning models. Unlike manual data entry systems, this project does not rely on user-generated inputs. Instead, it uses real-world datasets containing phishing and legitimate website URLs. These datasets are collected from trusted repositories such as PhishTank, Kaggle, and open-source URL databases widely used in cybersecurity research.

Each website entry is stored as a structured data record consisting of multiple attributes that represent the characteristics of a URL. These attributes are used for feature extraction, model training, prediction, and performance evaluation.

Types of Information Collected:

- Website URL strings
- Domain name details
- URL length and structure
- Presence of HTTPS / SSL certification
- Use of IP address instead of domain name
- Number of subdomains, dots, and slashes
- Presence of suspicious characters and phishing-related keywords

Examples of Attributes Used for Phishing Detection

- a) Length of URL
- b) Presence of HTTP/HTTPS protocol
- c) Use of prefix or suffix in domain
- d) Number of subdomains
- e) Presence of suspicious symbols
- f) Usage of IP address instead of domain

Using this structured dataset, the machine learning model learns patterns associated with phishing websites and accurately classifies new, unseen URLs in real time through browser integration.

V. DATA PREPROCESSING

To preserve the accuracy, consistency, and reliability of the data used in the Phishing Website Detection system, data preprocessing is a critical stage. Since phishing-related datasets are collected from multiple online sources and contain URLs with varying formats and structures, preprocessing ensures that the data is clean, standardized, and suitable for effective feature extraction and machine learning-based classification.

A. URL Input Validation

At the time of dataset preparation, all collected URLs are validated for correctness and completeness. Invalid URLs, broken links, incomplete records, or entries with missing labels (phishing or legitimate) are identified and removed. URL syntax, protocol format (HTTP/HTTPS), and domain structure are verified to ensure that only valid and usable data is considered. This validation prevents unreliable or misleading data from affecting the accuracy of phishing detection.

B. Consistency in Feature Labels and Attributes

Different datasets may describe similar URL characteristics using different terminologies or formats. To ensure consistency across the entire dataset, all extracted features such as protocol usage, domain type, URL length, and special character presence are standardized into predefined feature categories. This uniform representation improves compatibility with machine learning models and ensures consistent interpretation of phishing indicators.

C. Identifying and Eliminating Duplicate URLs

Duplicate URLs can negatively impact model performance by causing bias and overfitting. The system performs checks to identify and remove duplicate phishing or legitimate URLs from the dataset. Eliminating redundancy enhances dataset quality, improves generalization, and ensures that the trained model accurately learns distinguishing patterns.

D. Standardization of Feature Values

All extracted features are converted into standardized numerical or binary formats. For example, HTTPS presence is encoded as binary values, while numerical attributes such as URL length and number of subdomains are normalized. This standardization ensures consistent data representation, enhances machine learning efficiency, and improves classification accuracy.

E. Cleaning Dataset and Access Control

Noisy entries, mislabeled URLs, and incomplete feature records are removed from the dataset. Additionally, dataset access is restricted to authorized system components to ensure integrity and security during preprocessing and model training. These steps enhance data reliability and protect the overall system from corrupted inputs.

F. Compiling Features for Model Training and Analysis

After preprocessing, the cleaned and structured feature dataset is compiled for machine learning model training and evaluation. Feature frequency analysis, correlation analysis, and class distribution analysis are performed to support accurate phishing detection. The processed dataset also serves as a foundation for future enhancements such as advanced analytics, model optimization, and continuous retraining.

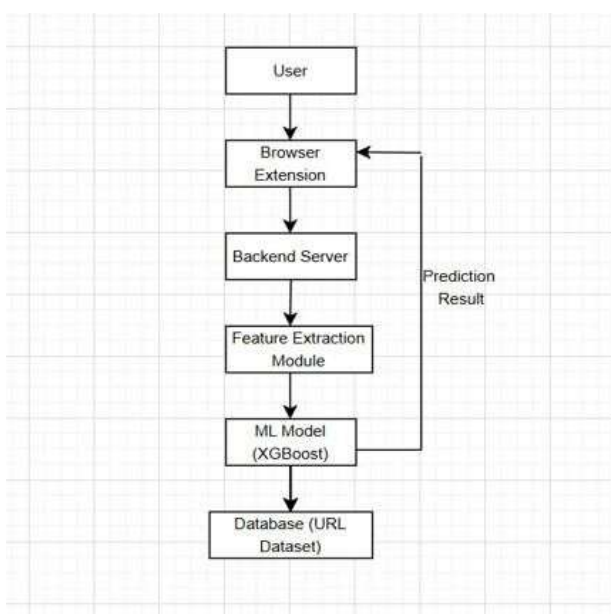


Fig. 1. system architecture

VI. IMPLEMENTATION

To guarantee maintainability, clarity, and efficient communication between all system components, the AI Resume Builder was developed using a structured and modular approach. The system architecture was created to facilitate future improvements and scalability while preserving a seamless and intuitive user experience.

React.js was used to create the application's front-end user interface, which offers interactive and responsive elements for user registration, resume data entry, template selection, and real-time preview. Using Node.js and Express.js, the backend logic was created in accordance with the principles of modular programming to handle secure authentication, API handling, and business logic. Component-based design and object-oriented programming (OOP) principles were used to guarantee reusability and the division of responsibilities among various system modules.

A. Dataset Collection and Analysis

The dataset used for the Phishing Website Detection system was collected from publicly available and reliable cybersecurity sources during various phases of system development. Instead of user-generated information, the system relies on structured datasets containing phishing and legitimate website URLs. Each URL entry was transformed into a structured data record that captures essential features required for feature extraction, machine learning training, and evaluation.

The collected dataset includes information such as URL strings, domain characteristics, protocol usage, presence of suspicious patterns, and classification labels indicating whether a website is phishing or legitimate. Additional analytical data includes the frequency of phishing indicators, distribution of URL lengths, and feature correlation values.

Data analysis was conducted to identify trends in phishing behavior, such as commonly exploited domain patterns, excessive use of subdomains, and suspicious keyword occurrences. Similar to many real-world cybersecurity datasets, the number of phishing URLs exceeded that of legitimate ones, highlighting the prevalence of phishing attacks. Correlation analysis was performed to examine relationships between extracted features and classification accuracy. The overall effectiveness of the detection system was measured using classification accuracy, precision, recall, and false positive rates derived from the dataset.

B. Data Preparation

Proper data preparation was carried out before storing and utilizing the dataset in the system to ensure accuracy, consistency, and reliability.

1) *Incomplete and Invalid URL Records*: URLs with missing labels, incorrect formatting, unreachable domains, or incomplete feature extraction were identified and removed. Only fully processed and valid URLs were retained for model training and testing.

2) *Normalization of Feature Data*: All extracted features were converted into standardized formats. Boolean features such as HTTPS usage were encoded into binary values, while numerical features like URL length and subdomain count were normalized. This ensured compatibility across all machine learning models.

3) *Identification of Duplicate URLs*: Duplicate phishing and legitimate URLs were detected and eliminated to prevent biased learning and overfitting. Removing duplicate entries improves generalization and prediction reliability.

4) *Logical Grouping of Feature Data*: Extracted URL features were logically grouped into lexical features, structural features, and domain-based features. This modular grouping supports efficient feature processing, analysis, and model training.

5) *Deriving Additional Attributes for Analysis*: Additional derived attributes such as phishing keyword density, URL entropy, and subdomain depth were generated. These attributes enhance detection accuracy and support future improvements such as advanced analytics and real-time monitoring dashboards.

C. Model Development

The primary goal of this project was to develop a structured, object-oriented machine learning model capable of classifying websites as phishing or legitimate. The system architecture follows modular and scalable design principles, ensuring ease of maintenance and future extensibility.

The phishing detection workflow involves URL capture, feature extraction, model inference, and result classification. Multiple supervised learning algorithms were evaluated, including Decision Tree, Random Forest, and XGBoost classifiers. Among these, XGBoost demonstrated superior accuracy, robustness, and resistance to overfitting, making it the preferred model for deployment. The detection process is executed only after successful feature extraction from URLs. Rule-based logic and machine learning classification work together to identify phishing indicators and generate consistent and reliable predictions.

D. Model Evaluation and Performance Metrics

The phishing detection model was evaluated based on functionality, accuracy, reliability, and real-time performance. Evaluation metrics included accuracy, precision, recall, F1-score, and false positive rate. Extensive testing was conducted using unseen URL datasets to assess generalization capability. The analysis demonstrated that the detection system significantly improved phishing identification accuracy while minimizing false positives. Compared to traditional blacklist-based detection methods, the proposed machine learning-based approach reduced detection time and improved adaptability against new phishing patterns. Overall, the evaluation confirms that the system effectively enhances website security and provides reliable real-time phishing detection.

E. Increasing System Transparency and Explainability

The phishing website detection system incorporates a transparent and traceable workflow to enhance trust among users and system administrators. Each stage—from URL capture and feature extraction to classification and alert generation—is clearly defined and logged. Prediction logs, timestamps, and

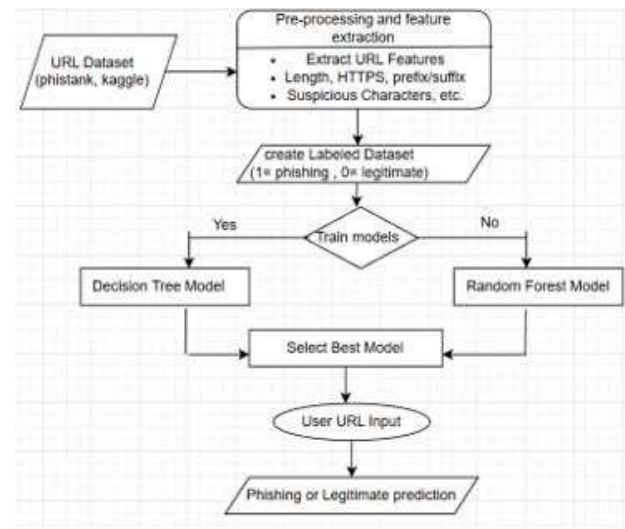


Fig. 2. Machine Learning Model Workflow Diagram

classification outcomes are stored securely and made accessible for authorized auditing purposes. Feature importance analysis provides explainability by highlighting which URL characteristics contributed most to a phishing classification. This transparency improves system credibility and encourages responsible usage of AI-driven cybersecurity tools.

F. A Scalable and Lightweight Backend Architecture

The backend architecture of the phishing detection system is designed to be lightweight, scalable, and efficient. Incoming URLs are validated before processing to ensure reliable predictions. A role-based access control mechanism restricts access to administrative and system-level operations. The modular backend supports quick integration of new machine learning models, datasets, and third-party cybersecurity tools. This design enables future enhancements without affecting system stability or performance.

G. Phishing Detection System User Interface

The user interface of the phishing detection system is designed to be simple, intuitive, and responsive. The browser extension provides instant visual alerts when a phishing website is detected and confirmation messages for legitimate sites. The minimalistic interface ensures that even non-technical users can easily understand security warnings and take appropriate actions. Real-time detection and immediate feedback significantly enhance usability and user confidence.

VII. CONCLUSION AND FUTURE WORK

Impact of the Phishing Website Detection System The proposed system offers an effective and automated solution for identifying phishing websites, significantly reducing the risk of cyber fraud and credential theft. The integration of machine learning ensures adaptability to evolving phishing techniques. High System Reliability Comprehensive testing confirms the system's ability to consistently detect phishing

websites with high accuracy while maintaining fast response times suitable for real-time use. Scalable System Architecture The modular and scalable architecture supports large datasets and concurrent users without compromising performance, making it suitable for enterprise-level deployment. Future Enhancements

Planned improvements include:

- Deep learning-based phishing content analysis
- Real-time web traffic monitoring
- Integration with browser and email security platforms
- Continuous model retraining using live phishing data
- Multilingual phishing detection support

REFERENCES

- [1] M. Naseeb, S. Anjum, M. A. Khan, M. L. Khan, and A. Rehman, "Website Phishing Attack Detection Using Innovative Meta-Learning Based Ensemble Approach," *IEEE Access*, vol. 13, pp. 1–15, 2025
- [2] A. K. Dutta, "Phishing Website Detection Using Machine Learning Techniques," *PLOS ONE*, vol. 16, no. 4, pp. 1–18, 2021.
- [3] V. Sharma, R. Mehta, and A. Verma, "Cybersecurity Threat Detection Using Artificial Intelligence Techniques," in *AIP Conference Proceedings*, vol. 2752, no. 1, pp. 1–8, 2024.
- [4] I. R. Abdelhamid, H. M. El-Bakry, and A. A. Darwish, "Phishing Detection: A Recent Intelligent Machine Learning Comparison Based on URL Features," *International Journal of Computer Applications*, vol. 147, no. 11, pp. 1–7, 2016.
- [5] A. Jain and 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.
- [6] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1245–1254, 2009.
- [7] C. Whittaker, B. Ryner, and M. Nazif, "Large-Scale Automatic Classification of Phishing Pages," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2010.
- [8] U. R. Asha and B. Vinod, "Phishing Website Detection Using Machine Learning Techniques," *International Journal of Advanced Research in Computer Engineering Technology (IJARCET)*, vol. 6, no. 10, pp. 1–6, 2017.
- [9] . Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Predicting Phishing Websites Using Classification Mining Techniques with Experimental Case Studies," in *Proceedings of the International Conference on Information Technology and Applications*, pp. 176–181, 2010.
- [10] Y. Pan and X. Ding, "Anomaly-Based Web Phishing Page Detection," in *Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC)*, pp. 381–392, 2006.
- [11] A. Al-Ajlan and H. Ykhlef, "Artificial Intelligence Applications in Cybersecurity: Threat Detection and Prevention," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 1–15, 2020.