IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.151079

Phishing Website Detection using Machine Learning

Prof. More N.N.[1], Wayal Akanksha Sandip[2], Walunj Sneha Bhausaheb[3]

Guide, Master of Computer Application, Samarth College of Engineering and Management, Belhe

Savitribai Phule Pune University, Pune, India[1]

Master of Computer Application, Samarth College of Engineering and Management, Belhe

Savitribai Phule Pune University, Pune, India[2,3]

Abstract: Phishing is a cyber-attack in which criminals create fake websites that imitate trusted banks, online stores, and government portals to steal passwords, card numbers, and other sensitive information. Blacklist-based protection reacts only after a phishing site has been reported, while new phishing sites appear every day and many remain online for only a few hours. This research proposes a Phishing Website Detection System using Machine Learning that classifies a website as phishing or legitimate by analysing its URL, domain, and page-content characteristics. The proposed system collects a labelled dataset, extracts features such as URL length, use of an IP address, the "@" symbol, number of sub-domains, HTTPS status, domain age, and the ratio of external links, and trains supervised classifiers including Logistic Regression, Support Vector Machine, Random Forest, and XGBoost. The models are compared using accuracy, precision, recall, F1-score, and the confusion matrix, and the best model is placed behind a web application in which a user enters a link and receives a verdict with the main reasons behind it. The system supports, and does not replace, user caution, and it is designed to protect user privacy and keep false alarms on genuine websites low. Overall, the system provides a data-driven framework for early and transparent detection of phishing websites.

I. INTRODUCTION

The internet is now used for banking, shopping, education, and government services, which makes it an attractive target for criminals. In a phishing attack the victim receives a message containing a link to a website that looks almost identical to a genuine one and is tricked into entering confidential information. Because the fake page copies the logo, layout, and language of the real organization, even careful users find it difficult to notice the difference.

Conventional defences such as blacklists, browser warnings, and manual reporting depend on a site being identified and verified before it is blocked. Phishing pages are cheap to create and are often abandoned within hours, so many of them are not listed in time. Machine learning offers a different approach: instead of remembering known malicious sites, a model learns the patterns that phishing sites tend to share, such as unusually long URLs, hidden IP addresses, many sub-domains, recently registered domains, and forms that send data to another domain. However, a prediction is useful only when users can understand it and when it does not frequently block genuine websites.

This research proposes a Phishing Website Detection System using Machine Learning. A user enters a website address, the system extracts a set of features, a trained classifier predicts whether the site is phishing or legitimate, and the result is displayed with the main reasons. By combining several algorithms, a comparison of their performance, and a simple web interface, the proposed system aims to support safer browsing, reduce the number of successful phishing attacks, and protect user data.

II. LITERATURE REVIEW

Phishing detection has been studied through blacklists, heuristic rules, and machine learning. Each approach uses different evidence to decide whether a website is malicious, and recent research increasingly relies on automatically learned classifiers. The following review examines relevant research and identifies its contribution to the proposed system.

Ma et al. (2009) showed that malicious websites can be identified from the lexical and host-based properties of their URLs without any manual list of known sites. Their work established that features available before a page is even loaded can support classification, which is the basis for the URL-based features used in the proposed system. Sahoo et al. (2017) surveyed malicious URL detection and described how feature extraction followed by supervised classification became the standard approach, while also noting the problem of attackers constantly changing their tactics.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.151079

Mohammad et al. (2014) studied the characteristics of phishing websites, including the use of an IP address in the URL, URL length, the "@" symbol, sub-domain count, and domain-related information, and made available the widely used UCI Phishing Websites dataset. Sahingoz et al. (2019) compared several machine learning algorithms on URL-based features and reported that tree-based ensembles such as Random Forest performed strongly. Random Forest (Breiman, 2001) and gradient-boosted trees such as XGBoost (Chen and Guestrin, 2016) are popular because they handle mixed feature types and provide feature importance scores, which also helps in explaining predictions.

A limitation of many predictive models is that their output is only a label. Explainability techniques such as SHAP (Lundberg and Lee, 2017) and LIME (Ribeiro et al., 2016) estimate the contribution of individual features to a prediction and can help a user understand why a website has been flagged. However, such explanations should be treated as supporting information and not as proof that a site is malicious.

The reviewed research establishes the importance of three elements: informative website features, accurate classification, and understandable output. However, many studies stop at reporting model accuracy on a fixed dataset. Findings from one dataset cannot automatically be generalised to new phishing sites, because attackers change their techniques over time. The proposed system builds on these research directions by combining feature extraction, a comparison of classifiers, reason-based verdicts, and a working web interface, while recognising the need for retraining, privacy, and human judgement.

III. PROBLEM STATEMENT

Phishing websites are created faster than they can be reported and blacklisted. Users usually decide whether a link is safe by appearance alone, and a convincing fake page gives no visible warning. Blacklist-based tools miss newly created sites, rule-based tools need constant manual updating, and many existing solutions do not explain why a site is considered dangerous, so users either ignore the warning or lose trust in the tool.

Although machine learning can classify websites, some solutions provide only a label without reasons, may block genuine websites because of a high false-positive rate, and may use data that quickly becomes outdated. Furthermore, models that rely on page content or external lookups may be slow or unsafe if a malicious page is loaded directly.

Therefore, there is a need for a Phishing Website Detection System using Machine Learning that analyses the URL, domain, and content features of a website, identifies phishing sites before the user shares any information, compares several classification algorithms to select a reliable one, and presents the verdict with understandable reasons. The central problem addressed by this research is how to combine accurate classification, low false alarms, and simple explanations in a system that ordinary users can operate.

IV. PROPOSED SYSTEM

The proposed system, Phishing Website Detection using Machine Learning, is designed to identify phishing websites from their URL, domain, and page characteristics and warn users before they enter sensitive information. The system integrates feature extraction, a comparison of supervised classifiers, and a reason-based result into one web application. The system accepts a website URL from the user, validates it, and extracts a feature vector. The vector is given to a trained classifier that returns the probability that the site is phishing. Based on the probability and a threshold established during development, the website can be shown as low, medium, or high risk. An explanation component uses feature importance or SHAP to identify the features that contributed most to the decision, such as the use of an IP address or a very recently registered domain, so that the user can understand the warning.

Main Components of the Proposed System

1. URL Input and Validation: Accepts the website address from the user and checks that it is well formed.
2. Dataset Collection: Maintains a labelled dataset of phishing and legitimate websites for training and evaluation.
3. Data Preprocessing: Removes duplicates, handles missing values, and prepares the features for learning.
4. Feature Extraction: Computes URL-based, domain-based, and content-based features of the website.
5. Classification Model: Applies a trained machine learning model to estimate the probability of phishing.
6. Explanation Module: Explains the main features influencing each prediction.
7. Risk Classification: Groups websites into validated low, medium, and high risk levels.
8. User Interface and History: Shows the verdict, reasons, and earlier scans in an understandable format.
9. Privacy and Security Management: Stores only the data needed, protects accounts, and avoids executing suspicious pages.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.151079

V. SYSTEM ARCHITECTURE

The system architecture illustrates the workflow of the Phishing Website Detection System. It consists of interconnected modules for URL submission and data collection, preprocessing, feature extraction, model training, classification, explanation, and user alerts.

System Components

1. URL Submission and Data Collection: Receives the website address and the labelled dataset used to train the model.
2. Data Preprocessing: Cleans the collected data, handles missing values, and scales the feature values.
3. Feature Extraction: Computes URL, domain, and content features such as URL length, IP address, "@" symbol, domain age, HTTPS status, and external link ratio.
4. Machine Learning Model: Analyses the feature vector to estimate the likelihood of phishing using a suitable trained classifier.
5. Classification and Risk Level: Applies a threshold to decide the class and groups websites into risk levels.
6. Explanation Module: Identifies the features that most influenced the prediction and presents them in plain language.
7. User Interface and Alerts: Presents the verdict, scan history, and administrator reports to authorised users.

Working Principle

The system processes the submitted URL and extracts a feature vector, which is passed to a trained machine learning model to estimate the probability of phishing. The explanation module then identifies the main features contributing to the prediction. Based on these results, the user is warned or informed that the site appears legitimate. The result is advice and not a guarantee, so the final decision remains with the user.

System Architecture Diagram

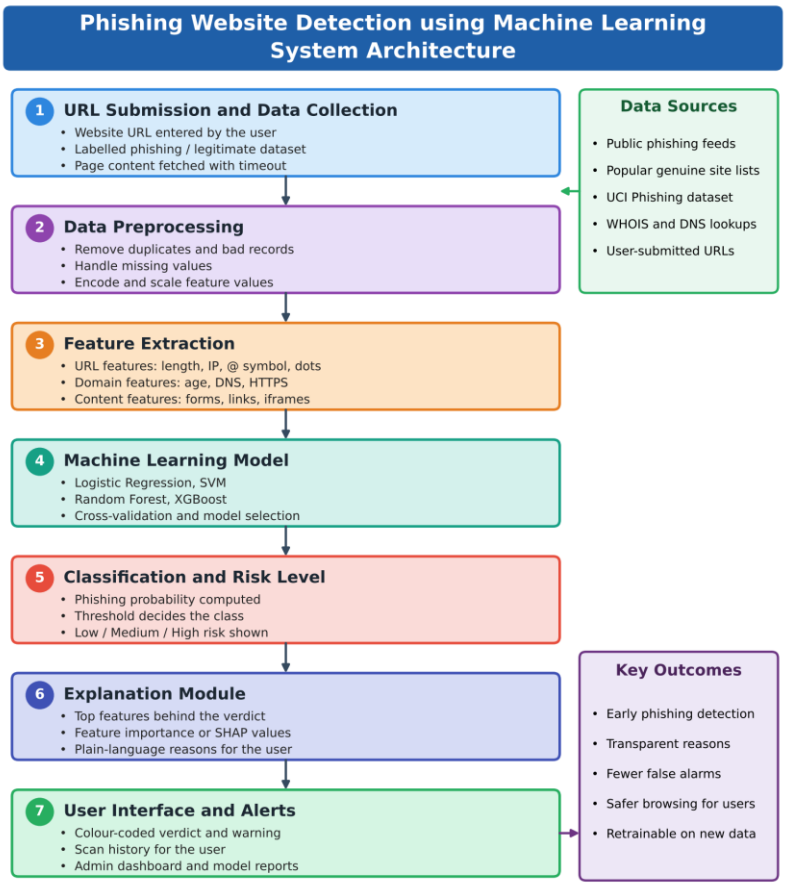


Fig. 1. System architecture of the proposed Phishing Website Detection System.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.151079

VI. RESEARCH METHODOLOGY

The research methodology describes the process of developing the Phishing Website Detection System using Machine Learning. The proposed approach combines dataset preparation, feature engineering, supervised classification, and explanation to identify phishing websites.

1. Data Collection and Preprocessing: A labelled dataset of phishing and legitimate websites is collected, and duplicate, malformed, and inconsistent records are removed. The class balance is checked and the data is split into training and test sets using a stratified split.
2. Feature Extraction: URL-based features (length, IP address, "@" symbol, dots, hyphens, suspicious words), domain-based features (domain age, DNS record, HTTPS, number of sub-domains), and content-based features (external link ratio, form action, hidden frames) are computed.
3. Model Development: Logistic Regression, Support Vector Machine, Random Forest, and XGBoost are trained on the same split. Hyper-parameters are tuned using grid search and k-fold cross-validation.
4. Phishing Prediction: The selected model estimates the phishing probability for a submitted URL. Websites may be classified into low-, medium-, and high-risk categories using defined thresholds.
5. Explainability Analysis: Feature importance or SHAP is used to show the features influencing individual predictions, helping users understand why a website has been flagged.
6. Evaluation and Monitoring: Model performance is assessed using accuracy, precision, recall, F1-score, specificity, and a confusion matrix on a held-out test set. Response time and the usefulness of the explanations are also examined.

The methodology aims to provide an accurate, interpretable, and privacy-aware approach to phishing detection. The effectiveness of the system must be established through actual experimental evaluation rather than assumed from its proposed design.

VII. IMPLEMENTATION

The proposed Phishing Website Detection System using Machine Learning is designed to integrate data processing, feature extraction, machine learning prediction, and explainable output into a unified framework.

1. URL Input Module: The user enters or pastes a website address, which is validated and normalised before processing.
2. Feature Extraction Module: The system computes URL, domain, and page-content features using libraries such as Requests, BeautifulSoup, tldextract, and python-whois, with a short timeout and without executing scripts from the page.
3. Data Processing Module: Missing or inconsistent values are handled and the features are arranged in the same order as the training data.
4. Prediction Module: A trained classifier, such as Random Forest or XGBoost, loaded from a saved model file, estimates the phishing probability.
5. Explanation Module: Feature importance or SHAP identifies the features contributing to the individual prediction.
6. Risk Assessment Module: Based on predefined thresholds, the system categorises the website as low, medium, or high risk.
7. Reporting Module: A Flask-based web interface displays the verdict, reasons, and scan history, and an administrator page allows the dataset to be uploaded, the models to be retrained, and the model reports to be viewed.

The implementation emphasises modular design, usability, data security, and responsible use. User accounts are protected with hashed passwords, only the minimum data required for history is stored, and predictions support, and do not replace, user judgement. The selected technologies and actual implementation results should be documented according to the tools and methods used in the developed project.

VIII. EXPERIMENTAL RESULTS AND DISCUSSION

The system extracts website features, applies machine learning to predict whether a site is phishing, and interprets the prediction. At this stage no experiments have been run, so no numerical results are claimed.

A. Experimental Results

The system performance will be evaluated using accuracy, precision, recall, F1-score, and specificity. Accuracy is the share of all websites classified correctly; precision is the share of flagged websites that are truly phishing; recall is the share of phishing websites that are detected; F1-score balances precision and recall; and specificity is the share of genuine

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.151079

websites correctly passed. These metrics measure the model's ability to identify phishing sites and to avoid blocking genuine ones.

Evaluation Metric	Measured Result
Accuracy	To be recorded after testing
Precision	To be recorded after testing
Recall	To be recorded after testing
F1-score	To be recorded after testing
Specificity	To be recorded after testing

B. Graphical Analysis

After the models are trained, a bar chart comparing the evaluation metrics of all classifiers and ROC curves will be prepared to visualise performance. Recall is particularly important because it measures the proportion of actual phishing websites detected by the system. Precision indicates how reliable the phishing warnings are, while the F1-score provides a balance between precision and recall. Figure 2 shows the confusion matrix that will be used to calculate these metrics.

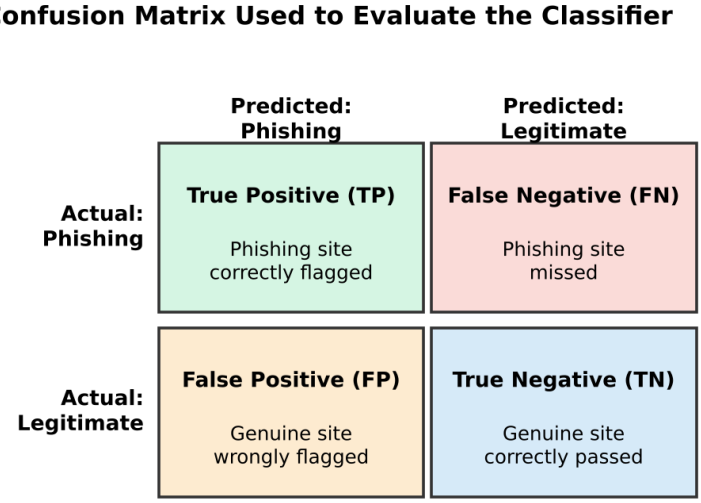


Fig. 2. Confusion matrix layout used to calculate the evaluation metrics.

C. Discussion

The proposed approach combines phishing prediction with understandable reasons to support informed decisions by users. By examining URL, domain, and content features, the system can show why a website has been classified as risky. Feature importance and SHAP can identify which features contribute to individual predictions. Features such as URL length, use of an IP address, domain age, HTTPS status, and the ratio of external links are commonly considered informative in the literature, but the actual ranking must be determined from the trained model and the dataset used.

However, the reliability of the system depends on data quality, model selection, and validation on unseen websites. Evaluation should also consider false-negative predictions, which let phishing sites pass, false-positive predictions, which block genuine sites, and the usefulness of the explanations for users. Actual protection against phishing must be established through further testing on newly reported phishing sites rather than inferred from test-set metrics alone.

IX. APPLICATIONS

The proposed Phishing Website Detection System using Machine Learning has the following applications:

1. Safe Browsing for Individuals: Lets users check a link before entering passwords or payment details.
2. Online Banking and Payment Protection: Helps customers verify links received in messages before logging in to financial services.
3. Education and Awareness: Helps students and staff understand which signs make a website suspicious through the displayed reasons.
4. Organizational Security: Allows an organization to screen links reported by employees and reduce credential theft.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.151079

5. Browser Extension Support: The trained model can be connected to a browser extension to warn users in real time.
6. E-mail and Message Filtering: Links in e-mails and messages can be scanned to support spam and phishing filters.
7. Security Research: Provides a platform to compare algorithms and retrain models as new phishing datasets become available.
8. Transparent Decision-Making: Uses feature importance or SHAP to clarify the factors behind a warning and support informed user decisions.

X. LIMITATIONS AND FUTURE WORK

Limitations

1. Dataset Age: Phishing techniques change quickly, so a model trained on an old dataset may lose accuracy.
2. Data Quality and Reliability: Mislabelled or duplicate records in the dataset may reduce prediction accuracy.
3. External Dependencies: Features such as domain age depend on WHOIS and DNS services, which may be unavailable or slow.
4. Evasion by Attackers: Attackers may imitate the features of genuine sites to avoid detection.
5. Limited Explainability: Feature importance and SHAP provide useful insights, but they may not fully capture complex relationships within the model.
6. Scope: The prototype supports public web pages only and does not handle pages behind a login or heavily script-based pages.

Future Work

1. Real-Time Browser Extension: A browser extension could warn users before a page is loaded.
2. Advanced Models: Deep learning models that read page text and visual appearance could be evaluated while keeping the output interpretable.
3. Continuous Retraining: Newly reported phishing sites could be used to retrain the model automatically.
4. E-mail and SMS Link Scanning: The system could be extended to scan links received through messages.
5. Multilingual Interface: A regional-language interface, including Marathi, could make the system accessible to more users.
6. Privacy and Robustness Enhancement: Privacy-preserving processing and tests against adversarial examples could strengthen the system.

XI. CONCLUSION

This research presents a Phishing Website Detection System using Machine Learning to address the challenge of identifying fake websites that steal user information. Unlike approaches that depend only on blacklists, the proposed system learns the features of phishing websites and can judge sites that have not yet been reported. By analysing URL, domain, and content features and comparing several classifiers, the system aims to provide early warnings together with understandable reasons.

The key contribution of this research is the integration of feature extraction, model comparison, explanation, and a simple web interface within a single framework. It can help users avoid unsafe links and help reviewers understand why a website was flagged. However, its practical value must be established through reliable testing on up-to-date data and evaluation of false-positive and false-negative rates. With appropriate safeguards for privacy and regular retraining, the proposed approach offers a promising direction for developing more proactive and transparent phishing protection.

REFERENCES

[1]. J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond blacklists: Learning to detect malicious web sites from suspicious URLs,” in Proc. ACM SIGKDD, 2009, pp. 1245–1254.
[2]. D. Sahoo, C. Liu, and S. C. H. Hoi, “Malicious URL detection using machine learning: A survey,” arXiv:1701.07179, 2017.
[3]. R. M. Mohammad, F. Thabtah, and L. McCluskey, “Predicting phishing websites based on self-structuring neural network,” Neural Computing and Applications, vol. 25, no. 2, pp. 443–458, 2014.
[4]. 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.
[5]. L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.151079

[6]. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. ACM SIGKDD, 2016, pp. 785–794.
[7]. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems 30, 2017.
[8]. M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in Proc. ACM SIGKDD, 2016, pp. 1135–1144.