International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

Intelligent Browser-Extension For Real-Time Phishing Detection Using Hybrid Machine Learning Models

Aishwarya S. Sanap[1], Vidya B. Kale[2], Archana S. Kolhe[3],Prof.(Dr.)N.R.Wankhade[4], Prof S.R.Agrawal[5]
[1,2,3]PG, Dept. of computer Eng, Late G.N.Sapkal College of Engineering,Kalyani Hills, Anjaneri, Trimbakeshwar Road, Nashik,
[4,5]professor, Dept. of computer Eng, Late G.N.Sapkal College of Engineering,Kalyani Hills, Anjaneri, Trimbakeshwar Road, Nashik,
Emails: *aishwaryasanap9011@gmail.com[1], vidyakale005@gmail.com[2], archanakolhe59@gmail.com[3], hodcomp.lgnscoe@sapkalknowledgehub.org[4], s.ragrawal1976@gmail.com[5]*

Abstract
Phishing attacks remain a significant cybersecurity threat due to their evolving nature and reliance on social engineering techniques. While machine learning-based phishing detection models have demonstrated high accuracy in offline evaluations, their real-time deployment in client-side environments remains challenging. This paper presents a hybrid phishing detection framework that integrates offline machine learning analysis with real-time browser-based deployment. Multiple machine learning classifiers are evaluated using URL based features, and the Random Forest model is identified as the most effective classifier. Feature importance analysis is employed to extract the most influential phishing indicators, which are subsequently translated into a lightweight rule-weighted detection mechanism. This mechanism is implemented as a browser extension to enable real-time phishing detection without relying on external servers. Experimental results demonstrate that the proposed approach achieves high detection accuracy while maintaining low computational overhead. The system provides explainable detection decisions, preserves user privacy, and effectively bridges the gap between machine learning research and practical phishing defense systems suitable for real-world deployment.

Keywords: *Phishing detection, Machine learning, URL based analysis, Random Forest, Feature importance, Browser extension, Real-time detection, Cybersecurity*

1. Introduction
The rapid expansion of internet-based services such as online banking, e-commerce, cloud platforms, and social networking has significantly increased users' exposure to cyber threats. Among these threats, phishing attacks remain one of the most prevalent and damaging attack vectors, exploiting social engineering techniques to deceive users into revealing sensitive information. According to the Verizon Data Breach Investigations Report 2024, phishing continues to be a leading cause of data breaches worldwide, emphasizing the need for robust and adaptive phishing detection mechanisms [1]. Conventional phishing detection approaches primarily rely on blacklist-based and rule-based mechanisms. While these methods are computationally efficient, they suffer from inherent limitations, including delayed blacklist updates, inability to detect zero-day phishing attacks, and poor adaptability to rapidly evolving phishing strategies [2], [3]. As attackers continuously register new domains and modify URL structures, static defense techniques become increasingly ineffective. To overcome these limitations, machine learning (ML)-based phishing detection approaches have gained significant research attention. ML models are capable of learning discriminative patterns from large-scale datasets and generalizing to unseen phishing URLs. Several studies have demonstrated the effectiveness of machine learning algorithms such as Decision Trees, Support Vector Machines, Random Forests, and ensemble classifiers for phishing detection using URL-based features [4]–[5]. URL-based approaches are particularly suitable for real-time detection due to their low computational overhead and independence from webpage content loading. Recent works have further explored deep learning and hybrid approaches

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

to enhance phishing detection performance. Convolutional Neural Networks and layered hybrid models incorporating URL, textual, and visual features have shown promising results [6], [7]. However, such approaches often require higher computational resources and are primarily evaluated in offline environments, limiting their applicability for real-time client-side deployment. Despite the growing body of research on phishing detection. limited attention has been given to bridging the gap between high-accuracy offline machine learning models and practical real-time deployment. Client-side solutions such as browser extensions provide a promising direction for immediate phishing detection and user warning, yet integrating ML intelligence into lightweight browser-based systems remains challenging [8]. In this work, we propose an intelligent phishing detection framework that combines offline machine learning analysis with real-time browser extension deployment. Multiple machine learning models are trained and evaluated using URL based features extracted from publicly available phishing datasets [9], Feature importance analysis using Random Forest is employed to identify the most influential phishing indicators. These ML-derived insights are then translated into a lightweight, rule-weighted detection mechanism implemented as a browser extension, enabling real-time phishing detection without relying on external servers. The main contributions of this paper are summarized as follows:

- A comparative evaluation of multiple machine learning models for phishing website detection using URL-based features.
- An explainable feature importance analysis to identify key phishing indicators influencing detection performance.
- A real-time client-side phishing detection system implemented as a browser extension, bridging the gap between offline ML models and practical deployment.
- Experimental validation demonstrating high detection accuracy with minimal computational overhead.

1.1. Methods of Phishing Detection System

1.1.1. Dataset Collection

The phishing detection system uses a publicly available phishing URL dataset containing 11,055 URL instances. The dataset consists of both phishing and legitimate websites represented using 30 URL-based features such as URL length, SSL certificate status, domain registration length, subdomain usage, and hyperlink-related attributes.

1.1.2. Data Preprocessing

Before training the machine learning models, preprocessing is performed to improve data quality and consistency. This includes removal of redundant attributes, handling inconsistent values, feature preparation, and splitting the dataset into training and testing sets using an 80:20 ratio.

1.1.3. URL Feature Extraction

The system extracts lexical, structural, and security-related URL features that are commonly associated with phishing attacks. Features such as presence of IP address, abnormal URL length, use of special characters, suspicious subdomains, SSL certificate validity, and redirecting patterns are analyzed to identify malicious behavior.

Machine Learning Model Training

Multiple machine learning algorithms are trained and compared for phishing website detection. The implemented classifiers include:

- Logistic Regression (LR)
- Decision Tree (DT)
- Support Vector Machine (SVM)
- K-Nearest Neighbor (KNN)
- Random Forest (RF)

These models are evaluated using accuracy, precision, recall, and F1-score metrics to determine the best-performing classifier.

1.2. Browser Extension Implementation

The real-time phishing detection module is implemented as a Google Chrome browser extension using JavaScript. The extension continuously monitors the currently visited URL, extracts phishing-related features, computes a phishing score, and immediately warns users if suspicious behavior is detected.

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

1.3. System Architecture

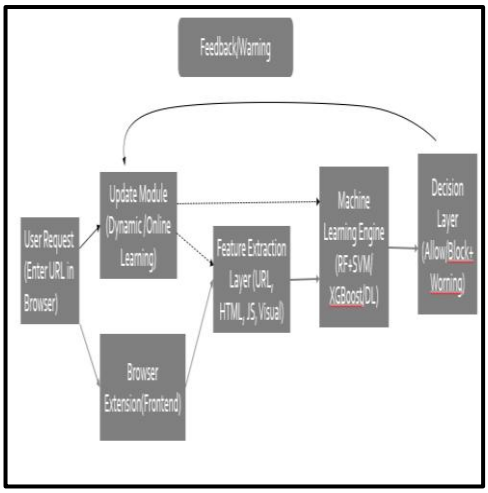

Figure 1 Architecture of Browser Extension

The architecture separates offline ML training from online real-time detection. In the offline layer, the UCI Phishing Websites Dataset is preprocessed and used to train five ML models. The Random Forest classifier's feature importance scores are normalized and embedded as JavaScript constants. In the online layer, the Chrome extension uses these weights to score every URL in real time without any server communication shown in Figure 1.

2. Experimental Setup

This section describes the experimental configuration used to evaluate the performance of the proposed phishing detection framework. The setup includes details of the dataset, feature set, training and testing strategy, machine learning models, evaluation metrics, and implementation environment.

2.1. Dataset Description

The experiments were conducted using publicly available phishing website datasets containing both legitimate and phishing URLs. After preprocessing, the final dataset consisted of 11,055 URL instances represented by 30 URL-based features. These features capture lexical, structural, and host-based characteristics commonly associated with phishing behavior, such as SSL certificate status, presence of IP addresses in URLs, abnormal URL structures, domain registration properties, and hyperlink-related attributes.

2.2. Feature Set

The feature set used in this study includes 30 URL-based attributes extracted from each website. These features are designed to capture various phishing indicators without requiring webpage content loading [1-5]. Examples of such features include SSL final state, URL length, use of IP address, presence of special characters, subdomain usage, domain age, web traffic rank, and link-related characteristics. The use of URL-based features enables efficient real-time detection and minimizes computational overhead.

2.3. Training and Testing Strategy

To ensure unbiased performance evaluation, the dataset was divided into training and testing subsets using an 80:20 split. The training set was used to learn model parameters, while the testing set was reserved exclusively for performance evaluation. This split ensures that the reported results reflect the generalization capability of the trained models on unseen data.

2.4. Machine Learning Models

Five machine learning classifiers were implemented and evaluated in this study:

- Logistic Regression (LR)
- Decision Tree (DT)
- Support Vector Machine (SVM)
- K-Nearest Neighbor (KNN)
- Random Forest (RF)

These models were selected due to their widespread use in phishing detection literature and their ability to handle different types of feature relationships.

2.5. Evaluation Metrics

The performance of each classifier was assessed using standard classification metrics commonly employed in cyber security research. These metrics include accuracy, precision, recall, and F1-score, which are defined mathematically as follows.

Accuracy measures the overall correctness of the classifier and is defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Precision represents the proportion of correctly identified phishing websites among all instances predicted as phishing:

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

Recall (also known as detection rate) measures the proportion of actual phishing websites that are correctly identified:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

F1-score is the harmonic mean of precision and recall, providing a balanced measure of classification performance:

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

In the above equations, T P denotes true positives, T N denotes true negatives, F P denotes false positives, and F N denotes false negatives. These metrics provide a comprehensive evaluation of detection performance, particularly in the
presence of class imbalance shown in Table 1.

Table 1 Performance Comparison of Machine Learning Models

Model	Accuracy	Precision	Recall	F1-score
Logistic Regression	0.9276	0.9343	0.9000	0.9168
Decision Tree	0.9697	0.9760	0.9551	0.9654
SVM	09498	0.9588	0.9265	0.9424
KNN	0.9484	0.9520	0.9306	0.9412
Random Forest	0.9765	0.9803	0.9663	0.9733

2.6. Random Forest Classification Analysis

A detailed classification report for the Random Forest model is presented in Fig. 2. The model achieved an overall accuracy of 98%, with high precision and recall for both phishing and legitimate classes. This demonstrates the robustness of the model in minimizing both false positives and false negatives shown in Figure 2.

MODEL COMPARISON RESULTS

	Model	Accuracy	Precision	Recall	F1-Score
2	Random Forest	0.976481	0.980331	0.966327	0.973279
1	Decision Tree	0.969697	0.976017	0.955102	0.965446
3	SVM	0.949796	0.958817	0.926531	0.942398
4	KNN	0.948440	0.951983	0.930612	0.941176
0	Logistic Regression	0.927635	0.934322	0.900000	0.916840

RANDOM FOREST CLASSIFICATION REPORT

	precision	recall	f1-score	support
0	0.97	0.98	0.98	1231
1	0.98	0.97	0.97	980
accuracy			0.98	2211
macro avg	0.98	0.98	0.98	2211
weighted avg	0.98	0.98	0.98	2211

Figure 2 Classification report and model comparison results

2.7. Confusion Matrix Analysis

The confusion matrix of the Random Forest classifier is shown in Fig. 2. The results indicate that the majority of phishing and legitimate websites are correctly classified. Only a small number of misclassifications are observed, highlighting the reliability of the proposed approach.

2.7.1. Feature Importance Analysis

Feature importance analysis was conducted using the Random Forest model to identify the most influential URL-based phishing indicators. Fig. 3 illustrates the top contributing features.

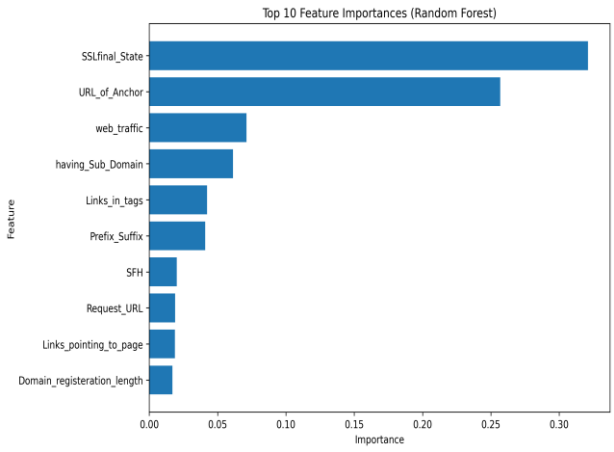

Figure 3 Confusion matrix of the Random Forest classifier

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

The effectiveness of the extension was evaluated by testing both legitimate and phishing-like URLs. Figure 3. illustrates normal browsing behavior on a legitimate website, where no warning is triggered.

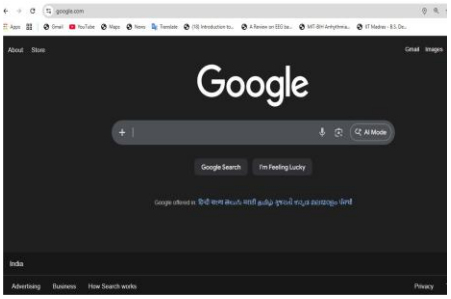

Figure 4 Normal browsing on a legitimate website

3. Results and Discussion

3.1.Results

Overall, the Random Forest classifier emerged as the most effective model for phishing detection. Security-related and URL-structure features were identified as the most influential indicators. The browser extension successfully detected phishing websites in real time, validating the practical usefulness of the proposed system shown in Figure 5.

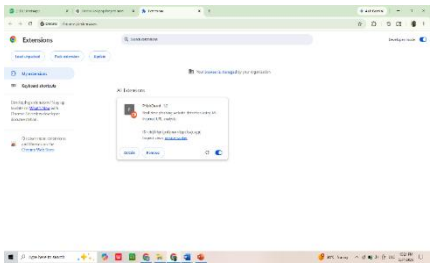

Figure 5 PhishGuard Pro v3.0 listed in Chrome Extensions manager (chrome://extensions)

7

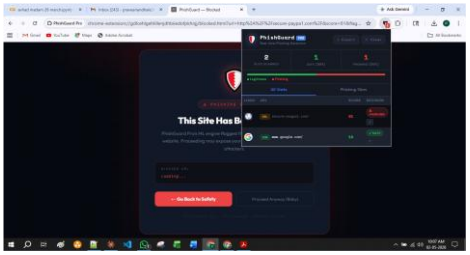

Figure 6 PhishGuard Pro blocking secure-paypa1.com (score: 81/100) with popup dashboard showing 2 sites scanned — 1 safe, 1 phishing

3.2. Discussion

The experimental results demonstrate that the proposed framework effectively bridges the gap between high-accuracy offline machine learning models and practical real-time deployment. The Random Forest model provides strong detection performance and interpretable feature importance, while the browser extension enables immediate phishing warnings with minimal computational overhead [6-9]. This hybrid design ensures high accuracy, explainability, user privacy, and real-world applicability, making the proposed system suitable for deployment in everyday browsing environments shown in Figure 7.

3.3. Implementation Code

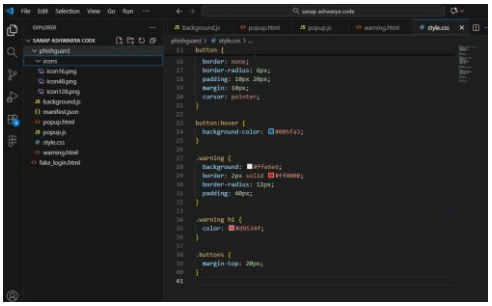

Figure 7 Extension code showing feature extraction and ML scoring engine

Conclusion

This work presented an efficient and practical phishing detection framework that bridges the gap between high accuracy offline machine learning models and real-time client-side deployment. Multiple machine learning classifiers were evaluated using URL-based features, and the Random Forest model achieved the best overall performance in terms of accuracy, precision, recall, and F1-score. Feature importance analysis further enabled interpretability and identification of the most influential phishing indicators. To ensure real-world applicability, the learned knowledge was translated into a lightweight rule-weighted detection mechanism and deployed as a browser extension.

Future Work

Future work will focus on incorporating adaptive learning mechanisms to dynamically update feature weights based on evolving phishing strategies. The

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0348
e ISSN: 2584-2854
Volume: 04 Issue: 06
June 2026
Page No: 2417 - 2422

integration of webpage content, visual similarity analysis, and deep learning-based embeddings is another promising direction. Furthermore, extending the system to support multiple browsers and mobile platforms, as well as conducting large-scale user studies, will further enhance its robustness and practical impact.

References

[1]. Kumar,A.,et.al: "PhishCatchter: client Side Defence Against Web Spoofing Attack Using Machine Learning ", International Journal of Computer Application (2022).
[2]. J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs,” ACM SIGKDD, pp. 1245–1254, 2009.
[3]. M. Adebowale, K. T. Lwin, and E. Sanchez, “Intelligent web phishing detection using Random Forest and Neural Networks,” Future Generation Computer Systems, vol. 115, pp. 62–71, 2021.
[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]. N. Abdelhamid, A. Ayesh, and F. Thabtah, “Phishing detection based associative classification,” Expert Systems with Applications, vol. 41, no. 13, pp. 5948–5959,2014.
[6]. X. Zhang, J. Zhou, and H. Wang, “Phishing detection using multi-feature fusion and ensemble learning,” IEEE Access, vol. 8, pp. 68348–68358, 2020.
[7]. W. Niu and Y. Zhang, “URL Representation Learning for Phishing Detection,” IEEE Transactions on Information Forensics and Security, vol. 16, pp. 3789–3803, 2021.
[8]. J. Feng and Y. Zou, “Phishing detection using attention-based deep neural networks,” Computers & Security, vol. 102, p. 102152, 2021.
[9]. Y. Huang and C. Qian, “Hybrid deep learning model for phishing detection,” IEEE Access, vol. 10, pp. 33421–33433, 2022.