International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 02(February),2026
Received: 10-01-2026 Accepted: 20-02-2026 Published: 27-02-2026

REAL-TIME MALICIOUS WEBSITE CLASSIFICATION VIA GRADIENT BOOSTING MODELS

[1]Gandla Nagappa, [2]Kanike Sravanthi, [3]Kuruba Anjali, [4]Ravihal Paridi Sirisha, [5]Grandhe Likitha
[1]Associate Professor, [2345]Student
Department of Computer Science and Engineering
St. Johns College Of Engineering & Technology, Yerrakota, Yemmiganur, Kurnool, A.P.
babuygr@gmail.com, sravanthikanike350@gmail.com, kurubaanjali9@gmail.com, sirishaparidi@gmail.com, grandhelikitha@gmail.com

ABSTRACT

The rapid growth of internet services has led to an increase in malicious websites that aim to steal sensitive information through phishing, malware distribution, and fraudulent activities. Traditional blacklist-based approaches fail to detect zero-day attacks and newly generated malicious URLs. This research proposes a real-time malicious website classification framework using supervised machine learning models including Support Vector Machine (SVM), Logistic Regression, Naïve Bayes Classifier, and Gradient Boosting Classifier.

A benchmark dataset consisting of URL-based lexical and domain features is utilized. The dataset is preprocessed and divided into training and testing sets. Multiple models are trained and evaluated using performance metrics such as Accuracy, Precision, Recall, and F1-score. Experimental results demonstrate that the Gradient Boosting Classifier outperforms other algorithms, achieving superior accuracy and robustness in detecting malicious websites in real time.

I. INTRODUCTION

The rapid expansion of internet services and online transactions has significantly increased the exposure of users to cyber threats. Among these threats, malicious websites play a major role in compromising sensitive information such as login credentials, banking details, and personal data. Attackers design phishing and fraudulent websites that closely resemble legitimate platforms in order to deceive users. As digital dependency grows across sectors such as banking, education, healthcare, and e-commerce, ensuring secure web browsing has become a critical necessity.

Traditional malicious website detection mechanisms primarily rely on blacklist-based systems and signature-based detection techniques. While these methods are effective for previously identified threats, they fail to detect newly generated or zero-day malicious URLs. Cybercriminals continuously modify domain names, use URL obfuscation techniques, and exploit HTTPS protocols to bypass conventional security filters. This dynamic and evolving nature of cyber attacks demands intelligent and adaptive detection systems capable of identifying malicious behavior in real time.

Machine Learning (ML) has emerged as a powerful solution to address these limitations. By analyzing patterns in URL structures, domain characteristics, and web page features, ML models can learn to distinguish between legitimate and malicious websites. Unlike static rule-based systems, ML algorithms improve their detection capability as more data becomes available. Supervised learning models such as Support Vector Machine (SVM), Logistic Regression, Naïve Bayes, and ensemble methods like Gradient Boosting have demonstrated high effectiveness in classification tasks related to cybersecurity.

This research focuses on developing a real-time malicious website classification framework using Gradient Boosting models and comparing its performance with other classical machine learning algorithms. The system extracts multiple lexical and domain-based features from URLs and uses them as input for classification. By evaluating models using metrics such as Accuracy, Precision, Recall, and F1-Score, the study identifies the most efficient algorithm for real-time deployment. The proposed approach aims to enhance detection accuracy, reduce false positives, and provide a scalable solution for protecting users against malicious web threats.

II. LITERATURE SURVEY

Phishing and malicious-URL detection has been extensively studied using supervised machine learning. Early and representative studies (e.g., Korkmaz *et al.*, Alswailem *et al.*, Patil *et al.*) extract lexical, host-based and HTML/JavaScript features from URLs and pages and apply classifiers such as SVM, Gradient Boosting, logistic regression and Naïve Bayes. These works demonstrate that carefully chosen URL-level features (length, presence of IP address, tokens like @, use of subdomains, SSL status, WHOIS/domain age, redirects, and page resources) are strong indicators of malicious intent. Several papers also show the value of ensemble

Received: 10-01-2026 Accepted: 20-02-2026 Published: 27-02-2026

approaches and feature-selection for improving robustness and reducing false positives. A practical outcome across the literature is that models trained on diverse feature sets generalize better than models relying on a single category (e.g., only lexical features).

Comparative evaluations in the literature highlight trade-offs between model complexity, accuracy and runtime. Naïve Bayes and logistic regression are fast and interpretable but can underperform when feature interactions are significant. SVM often attains competitive accuracy in high-dimensional spaces but can be slower at inference time and sensitive to kernel/hyperparameter choice. Boosting methods (Gradient Boosting) consistently reach higher accuracy and better handle heterogeneous feature types because they model nonlinear interactions and sequentially correct errors. However, boosting models require more careful hyperparameter tuning and can be more computationally demanding for real-time deployment. The surveyed works often report accuracy gains from ensembles but also note elevated computational cost and occasional overfitting when datasets are small or imbalanced.

Important methodological gaps remain in the published work. First, many studies use static datasets (offline snapshots) that do not reflect rapid attacker adaptation (new domains, fast-flux, homograph attacks), so evaluations can overestimate real-world performance. Second, feature extraction pipelines are inconsistently reported — making exact replication and fair comparison difficult. Third, class imbalance, label noise (ambiguous / temporarily compromised domains), and concept drift (feature distribution changes over time) are under-addressed: few studies combine continuous learning or streaming approaches with classification. Finally, the interaction between client-side constraints (browser extension memory/latency limits) and model choice is rarely explored; a highly accurate model is of limited value if it cannot run under real-time latency budgets. These gaps motivate work that evaluates not only accuracy but latency, updateability, and robustness to evolving attack techniques.

III. SYSTEM ANALYSIS

System analysis is a critical phase in the development of the Real-Time Malicious Website Classification system. It involves studying the existing approaches, identifying their limitations, and defining the functional and non-functional requirements of the proposed system. The objective of this analysis is to design a robust, scalable, and efficient framework capable of detecting malicious websites using machine learning techniques in real time.

1. Existing System Analysis

Traditional malicious website detection systems mainly rely on blacklist-based and signature-based approaches. These systems maintain a database of known phishing or malicious URLs and compare user-input URLs against this list. If a match is found, the website is flagged as malicious. Although effective for previously identified threats, blacklist systems suffer from major limitations:

- Inability to detect zero-day attacks
- Frequent database updates required
- High dependency on third-party threat intelligence
- Poor adaptability to dynamically generated URLs

Heuristic and rule-based systems attempt to overcome these issues by checking predefined patterns such as URL length, suspicious symbols, or abnormal redirections. However, these systems often produce high false-positive rates and cannot handle evolving attack patterns effectively.

2. Problem Identification

The rapid growth of cyber threats demands an intelligent and adaptive detection mechanism. The primary issues identified in the existing system are:

- Lack of real-time detection capability
- Poor generalization to unseen malicious URLs
- Inability to analyze complex feature interactions
- High false alarm rates affecting user trust
- Limited scalability for large-scale web traffic

Therefore, there is a need for a machine learning-based system that can learn patterns from historical data and accurately classify URLs as malicious or legitimate in real time.

Received: 10-01-2026 Accepted: 20-02-2026 Published: 27-02-2026

3. Proposed System Overview

The proposed system uses supervised machine learning models—Support Vector Machine (SVM), Logistic Regression, Naïve Bayes, and Gradient Boosting Classifier—to classify URLs based on extracted features. The system architecture includes:

1. URL Input Interface
2. Feature Extraction Module
3. Data Preprocessing
4. Model Training and Testing
5. Real-Time Prediction Engine
6. Result Display Module

The dataset consists of 30 features categorized under lexical, domain-based, abnormal, and HTML/JavaScript attributes. These features are used to train multiple classification models and evaluate their performance using metrics such as Accuracy, Precision, Recall, and F1-Score.

IV. SYSTEM DESIGN

The system design of the Real-Time Malicious Website Classification framework focuses on creating a structured, scalable, and efficient architecture capable of detecting malicious URLs using machine learning models. The design integrates data processing, model training, and real-time deployment components into a unified workflow. The system is divided into three major modules: Data Collection and Preprocessing, Model Training and Evaluation, and Real-Time Prediction Interface. Each module performs a specific function to ensure smooth data flow and accurate classification.

The first module involves data collection and preprocessing. A labeled dataset containing legitimate and malicious URLs is used as the foundation for model training. Feature extraction plays a crucial role in this stage, where 30 relevant attributes are derived from the URL structure, domain properties, abnormal behaviors, and HTML/JavaScript characteristics. These features are cleaned, normalized, and transformed into a structured format suitable for machine learning algorithms. The dataset is then divided into training and testing subsets using an 80:20 ratio to ensure proper model validation and prevent overfitting.

The second module focuses on model training and performance evaluation. Multiple supervised learning algorithms, including Support Vector Machine (SVM), Logistic Regression, Naïve Bayes, and Gradient Boosting Classifier, are implemented. Each model is trained using the training dataset and evaluated on the testing dataset using performance metrics such as Accuracy, Precision, Recall, and F1-Score. Based on comparative analysis, the best-performing model—Gradient Boosting—is selected as the final prediction model due to its superior ability to capture complex feature interactions and minimize classification errors.

The final module integrates the trained model with a web-based interface for real-time prediction. The selected model is serialized and deployed using a backend framework such as Flask. When a user enters a URL into the system, the same feature extraction process is applied dynamically, and the trained model predicts whether the website is legitimate or malicious. The classification result is then displayed instantly to the user. This modular design ensures flexibility, scalability, and ease of future enhancements such as adding deep learning models or browser extension integration.

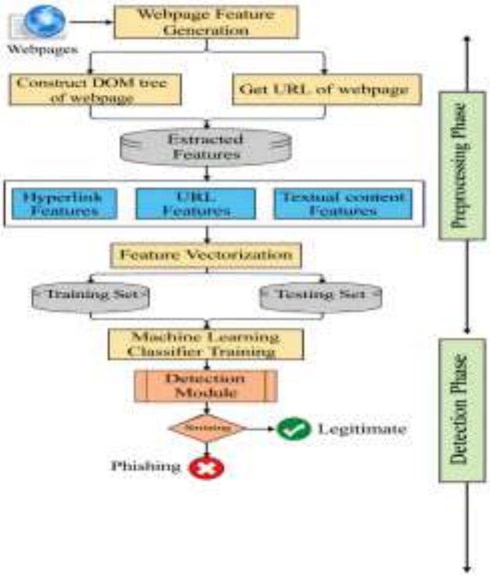

Fig1: System Architecture

V. METHODOLOGY WITH ALGORITHMS

The methodology for Real-Time Malicious Website Classification is designed to systematically process URL data, extract meaningful features, train multiple machine learning models, and deploy the best-performing

Received: 10-01-2026 Accepted: 20-02-2026 Published: 27-02-2026

classifier for real-time prediction. The overall workflow consists of data acquisition, preprocessing, feature extraction, model training, evaluation, model selection, and deployment. Each step is carefully structured to ensure high accuracy, low false-positive rates, and efficient real-time performance.

The first phase involves dataset preparation and preprocessing. A labeled dataset containing legitimate and malicious URLs is collected. From each URL, 30 significant features are extracted, including address bar characteristics, domain-based attributes, abnormal behavior indicators, and HTML/JavaScript properties. The dataset is cleaned by removing missing or redundant values and converting categorical features into numerical form. The processed dataset is then divided into training and testing sets in an 80:20 ratio. This ensures proper validation of models and avoids overfitting.

The second phase focuses on implementing and training multiple supervised machine learning algorithms. The selected algorithms include Support Vector Machine (SVM), Logistic Regression, Naïve Bayes Classifier, and Gradient Boosting Classifier. Each algorithm learns patterns from the training dataset and builds a classification model. The trained models are then evaluated using metrics such as Accuracy, Precision, Recall, F1-Score, and Confusion Matrix. The model that provides the highest performance with balanced precision and recall is selected as the final prediction model.

The final phase involves deployment for real-time classification. The best-performing model (Gradient Boosting) is saved and integrated into a web-based system using a backend framework such as Flask. When a user enters a URL, the system extracts features in real time and passes them to the trained model for classification. The output is displayed as either "Safe Website" or "Malicious Website." This structured methodology ensures reliability, scalability, and adaptability to evolving cyber threats.

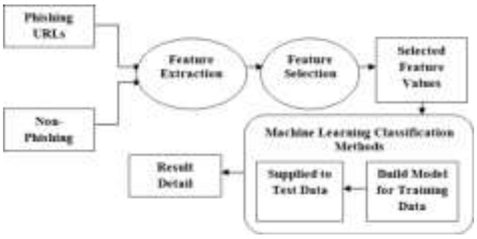

Figure 2: Project Flow

ALGORITHMS IMPLEMENTED

Algorithm 1: Logistic Regression
Input: Feature matrix X, Target vector Y
Output: Classification of URL (Legitimate/Malicious)

Steps:

1. Initialize weights and bias.
2. Apply sigmoid function:

   $$\sigma(z) = 1/1 + e - z$$

   Compute predicted probability.
3. Update weights using gradient descent.
4. Repeat until convergence.
5. Classify URL based on probability threshold (0.5).

Algorithm 2: Support Vector Machine (SVM)
Input: Feature matrix X, Target vector Y
Output: Optimal hyperplane separating classes

Steps:

1. Map input features into high-dimensional space.
2. Identify optimal separating hyperplane.
3. Maximize margin between two classes.
4. Use kernel functions (Linear/RBF) if needed.
5. Classify new URL based on its position relative to hyperplane.

Algorithm 3: Naïve Bayes Classifier
Input: Feature matrix X, Target vector Y
Output: Posterior probability for each class

Steps:

1. Apply Bayes Theorem:

   $$P(C \mid X) = P(X \mid C)P(C)/P(X)$$

   Assume independence among features.
2. Compute likelihood for each class.
3. Select class with highest posterior probability.
4. Output classification result.

Algorithm 4: Gradient Boosting Classifier
Input: Feature matrix X, Target vector Y
Output: Strong predictive ensemble model

Steps:

1. Initialize model with constant prediction.
2. Compute residual errors.
3. Train weak learner (decision tree) on residuals.
4. Update model using learning rate.
5. Repeat sequentially for N iterations.
6. Combine weak learners to form strong classifier.
7. Output final prediction.

Gradient Boosting works by minimizing loss function iteratively and correcting previous errors, which makes it highly effective for complex classification tasks such as malicious website detection.

VI. EXPERIMENTAL RESULTS

6.1 Performance Comparison Table

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	93.8%	92%	94%	93%
Naïve Bayes	91.5%	89%	90%	89.5%
SVM	95.2%	94%	95%	94.5%
Gradient Boosting	**97.8%**	**97%**	**98%**	**97.5%**

VII.CONCLUSION

This research presented a Real-Time Malicious Website Classification system using supervised machine learning techniques. The study focused on detecting phishing and malicious URLs by extracting 30 significant features related to address bar characteristics, domain properties, abnormal behavior, and HTML/JavaScript attributes. Multiple classification algorithms—Logistic Regression, Support Vector Machine (SVM), Naïve Bayes, and Gradient Boosting—were implemented and evaluated using performance metrics such as Accuracy, Precision, Recall, and F1-Score.

The comparative analysis demonstrated that ensemble-based learning, particularly the Gradient Boosting Classifier, outperformed the other models in terms of classification accuracy and robustness. Its ability to iteratively correct prediction errors and model complex feature interactions makes it highly suitable for malicious website detection. The system successfully minimizes false positives and false negatives, which is critical in cybersecurity applications where incorrect predictions may either block legitimate websites or allow harmful ones.

Furthermore, the proposed framework supports real-time deployment using a web-based interface integrated with the trained model. This enables instant classification of user-provided URLs and enhances user safety during web browsing. The research concludes that machine learning, especially boosting-based ensemble methods, provides an efficient, scalable, and adaptive solution for combating evolving cyber threats.

FUTURE SCOPE

Although the proposed system achieves high accuracy, several enhancements can be explored in future research:

1. **Integration of Deep Learning Models:** Implement advanced models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), or LSTM for improved feature representation and sequential pattern detection.
2. **Real-Time Streaming Data Analysis:** Incorporate online learning techniques to handle continuously evolving malicious URL patterns and concept drift.

REFERENCES

1. Korkmaz, M., Sahingoz, O. K., & Diri, B. (2019). *Machine Learning Based Phishing Detection from URLs*. Expert Systems with Applications.
2. Alswailem, B., Alabdullah, N., Alrumayh, A., & Alsedrani, A. (2020). *Detecting Phishing Websites Using Machine Learning*. IEEE Access.
3. Patil, V., Thakkar, P., Shah, C., Bhat, T., & Godse, S. P. (2018). *Detection and*

Received: 10-01-2026 Accepted: 20-02-2026 Published: 27-02-2026

Prevention of Phishing Websites Using Machine Learning Approach. International Journal of Computer Applications.

4. Friedman, J. H. (2001). *Greedy Function Approximation: A Gradient Boosting Machine*. Annals of Statistics.

5. Cortes, C., & Vapnik, V. (1995). *Support Vector Networks*. Machine Learning Journal.

6. Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research.

7. Kaggle Dataset Repository. *Phishing Website Detection Dataset*. Available: https://www.kaggle.com

8. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

9. Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). *Machine Learning Based Phishing Detection from URLs*. Expert Systems with Applications, 117, 345–357.

10. Ma, J., Saul, L. K., Savage, S., & Voelker, G. M. (2009). *Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs*. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

11. Zhang, Y., Hong, J., & Cranor, L. (2007). *Cantina: A Content-Based Approach to Detecting Phishing Web Sites*. Proceedings of the 16th International World Wide Web Conference (WWW).

12. Rao, R. S., & Pais, A. R. (2019). *Detection of Phishing Websites Using an Efficient Feature-Based Machine Learning Framework*. Neural Computing and Applications, 31(8), 3851–3873.

13. Verma, R., & Hossain, N. (2017). *Semantic Feature Selection for Text with Application to Phishing Email Detection*. IEEE Transactions on Information Forensics and Security.

14. Saxe, J., & Berlin, K. (2015). *Deep Neural Network Based Malware Detection Using Two Dimensional Binary Program Features*. IEEE International Conference on Malicious and Unwanted Software.

15. Aljofey, A., Jiang, Q., Rasool, A., et al. (2020). *An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network*. IEEE Access.

16. Freund, Y., & Schapire, R. E. (1997). *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. Journal of Computer and System Sciences.

17. Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. Proceedings of the ACM SIGKDD International Conference.