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. 14, Issue 10, October 2025

DOI: 10.17148/IJARCCE.2025.141019

Adaptive Phishing Detection Using Machine Learning: A Novel URL-Based Feature Analysis System

S. Roshan Pranao[1], Y. Sai Dheeraj[2], M. Tejas Srinivasan[3], Dr. Golda Dilip[4]

Student, Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India[1]

Student, Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India[2]

Student, Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India[3]

Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India[4]

Abstract: Phishing attacks continue to pose a significant cybersecurity challenge worldwide. This study presents a robust, adaptive URL-based phishing detection framework that integrates Balanced Random Forest, and XGBoost within a soft-voting ensemble architecture. The model utilizes lexical, structural, and domain-level features extracted directly from URLs, allowing real-time prediction without relying on blacklists. Experimental evaluation achieved an accuracy of 93.29%, precision of 92.30%, recall of 95.26%, F1-score of 93.76%, and ROC-AUC of 0.982. The ensemble demonstrates strong adaptability in identifying zero-day phishing URLs and can be seamlessly deployed via Flask-based APIs and browser extensions.

Keywords: Phishing Detection, Machine Learning, Ensemble Learning, Cybersecurity, URL Features.

I. INTRODUCTION

1.Background

Phishing remains one of the most pervasive and damaging threats in the digital landscape, leveraging social engineering to deceive users into divulging sensitive information such as login credentials, credit card numbers, and personal identifiers. The escalating sophistication of these attacks, which often employ obfuscation and dynamic content generation, renders traditional defense mechanisms like static blacklisting and simple heuristic-based filtering increasingly ineffective. Consequently, the cybersecurity community has shifted its focus towards more intelligent and adaptive solutions. Machine Learning (ML) has emerged as a powerful paradigm for this challenge, offering the ability to learn from vast datasets of malicious and benign URLs to identify complex patterns and detect novel threats in real-time. This paper proposes a robust machine learning framework for the accurate detection of phishing websites.

2.Existing Evidence (Literature Survey)

The detection of phishing websites has been an active area of research for over a decade. Early approaches primarily relied on blacklisting and whitelisting, where URLs were checked against manually curated lists of known malicious or safe domains. While simple to implement, this method fails against "zero-day" phishing attacks that use newly registered domains.

To overcome this, heuristic-based approaches were developed. These methods analyze various features of a website, such as URL structure (e.g., presence of '@' symbol, IP address in the domain name, URL length) and webpage content (e.g., suspicious forms, keyword stuffing), to calculate a threat score. However, phishers can often manipulate these features to bypass static rules, leading to a high rate of false negatives.

More recently, research has overwhelmingly demonstrated the superiority of Machine Learning (ML) models. Numerous studies have successfully applied classical algorithms like Support Vector Machines (SVM), Random Forest, Naive Bayes, and Logistic Regression for this classification task. These models leverage feature sets extracted from lexical analysis of URLs, domain registration data (WHOIS), and HTML/JavaScript content. Furthermore, advanced techniques using Deep Learning, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, have shown promise in automatically extracting intricate features, reducing the need for manual feature engineering.

 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. 14, Issue 10, October 2025
DOI: 10.17148/IJARCCE.2025.141019

3.Research Gap

Despite significant progress, several challenges persist in the field of ML-based phishing detection:

- **Adaptability to Evolving Threats:** Most existing models are trained on static datasets and struggle to keep pace with the dynamic nature of phishing attacks, such as the use of URL shorteners, multiple subdomains, and sophisticated JavaScript obfuscation.
- **Feature Robustness:** Many proposed solutions rely on a large number of hand-crafted features, some of which may become obsolete over time or are computationally expensive to extract in a real-time environment. There is a need for models that can perform effectively with a more optimized and robust feature set.
- **Real-Time Performance vs. Accuracy:** A trade-off often exists between the complexity (and thus accuracy) of a model and its prediction speed. Highly complex deep learning models may introduce latency, making them impractical for systems requiring instantaneous URL classification.
- **Dataset Imbalance and Quality:** Publicly available phishing datasets are often highly imbalanced and may not reflect the characteristics of modern phishing campaigns, potentially leading to biased and less generalizable models.

4.Objective

The primary objective of this research is to design, implement, and evaluate an efficient and highly accurate machine learning model for the real-time detection of phishing websites.The specific objectives are as follows:

- To engineer a robust and optimized set of features from URLs, webpage content, and domain properties that are resilient to common phishing obfuscation techniques.
- To conduct a comparative analysis of multiple machine learning algorithms (e.g., XGBoost, LightGBM, Random Forest) to identify the best-performing model for the phishing detection task.
- To train and validate the proposed model on a comprehensive and contemporary dataset comprising both benign and phishing URLs.
- To evaluate the model's performance using standard classification metrics, including accuracy, precision, recall, and F1-score, to demonstrate its effectiveness over existing methods.

5.Scope

The scope of this project is defined by the following boundaries:

- **Focus**: The research is focused exclusively on detecting phishing websites based on URL and webpage-based features.
- **Exclusions**: This study does not cover other forms of phishing attacks, such as smishing (SMS phishing), vishing (voice phishing), or spear-phishing emails. The detection of malware hosted on the webpage is also outside the scope of this work.
- **Implementation**: The project involves the development and evaluation of a classification model. It does not include the creation of a production-level browser plugin or a fully integrated enterprise security solution.
- **Data**: The model will be trained and tested using publicly available, well-known datasets to ensure the reproducibility of our results.

II. DATASET DESCRIPTION

The dataset used in this study was sourced from the Open Data Bay repository [7]. It comprises 822,010 unique URL samples. Each record includes two attributes:

• **URL**: The URL string representing either a legitimate or phishing domain.
• **Status**: A binary label where 0 represents phishing and 1 represents legitimate.

The dataset has approximately 48.7% phishing and 51.3% legitimate URLs, ensuring balanced learning. It provides global coverage with diverse lexical and structural patterns and was publicly released on June 5, 2025.

III. METHODOLOGY

The phishing detection pipeline includes data preprocessing, feature extraction, and model training. Features are derived using tldextract, and regex, focusing on lexical, structural, and domain-level properties. Models such as Balanced Random Forest, and XGBoost were combined in a soft-voting ensemble framework, with 80-20 stratified split and 5-fold cross-validation.

IV. EXPERIMENTAL SETUP AND RESULTS

The analysis was conducted on a balanced 100,000-URL subset, equally divided between phishing and legitimate samples. Feature extraction covered URL length, digit ratio, number of special characters, IP presence, and subdomain entropy.

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

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. 14, Issue 10, October 2025

DOI: 10.17148/IJARCCE.2025.141019

TABLE I METRICS

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC	AP-Score
Random Forest	92.28%	90.51%	95.35%	92.86%	0.9760	0.9752
Balanced Random Forest	92.18%	91.31%	94.12%	92.69%	0.9753	0.9743
XGBoost	92.83%	91.95%	94.68%	93.29%	0.9788	0.9786
Ensemble (BRF + XGBoost)	93.29%	92.30%	95.26%	93.76%	0.9817	0.9790

The XGBoost model achieved the highest standalone F1-score, while the ensemble (BRF + XGBoost) achieved the overall best accuracy and ROC-AUC. The combination provided balanced precision and recall, minimizing false positives and false negatives.

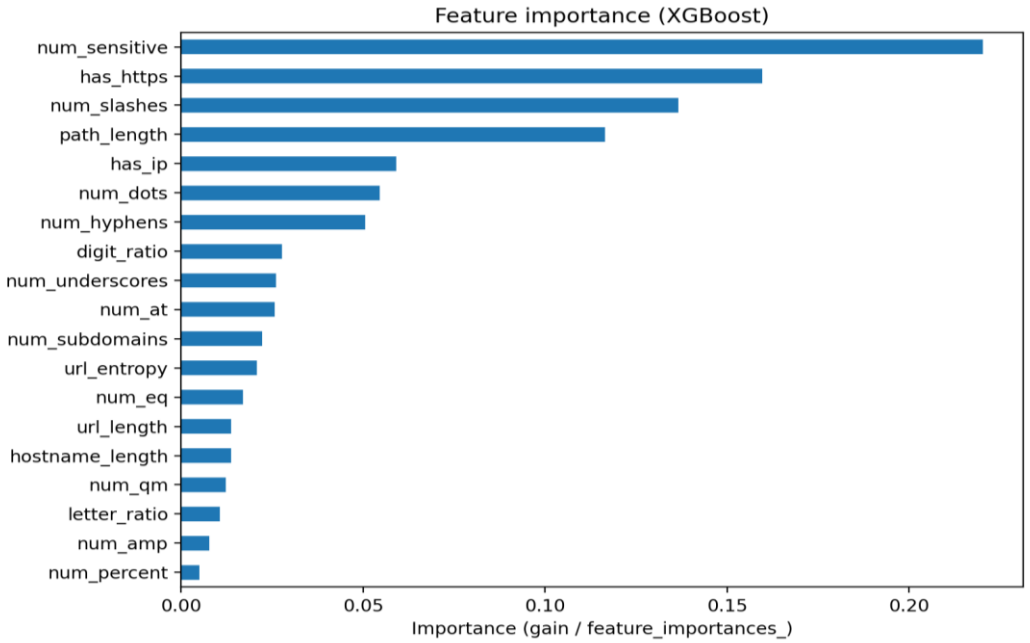

Fig. 1. XGBoost Feature Importance Ranking

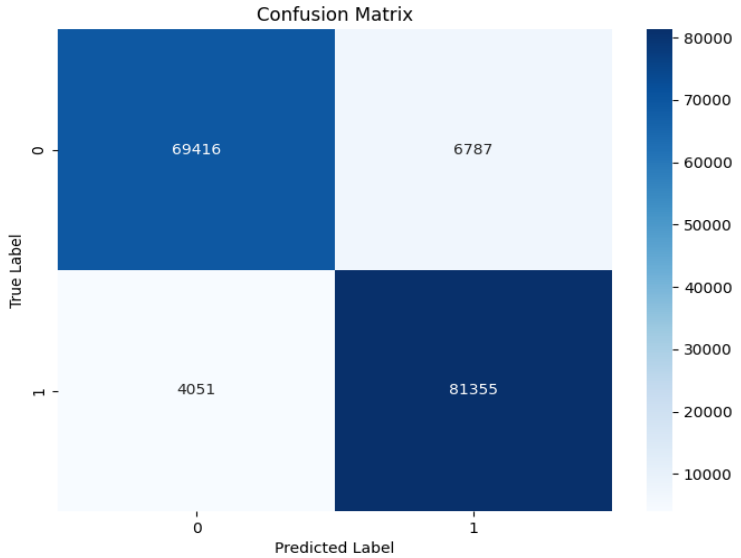

Fig.2. Confusion Matrix of Phishing vs. Legitimate URL Classification

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. 14, Issue 10, October 2025

DOI: 10.17148/IJARCCE.2025.141019

V. MODEL SELECTION AND ENSEMBLE RATIONALE

1.About The Model

While the Balanced Random Forest effectively addressed class imbalance through balanced bootstrapping and improved recall, XGBoost demonstrated strong precision and robust generalization. Their combination in the ensemble model, implemented via soft voting, leveraged the complementary strengths of both algorithms, thereby enhancing overall performance and mitigating overfitting.

VI. DISCUSSION

6.1 Pros

6.1.1 Specific Objectives

The ensemble achieved superior recall and F1-score compared to individual models. Its cost-sensitive learning reduced false negatives—a vital requirement for phishing detection systems.

6.1.2 Clear and Logical Flow

Each section builds logically on the previous one, creating a compelling argument for why your research is necessary and important

6.1.3 Manages Expectations

The Scope section is crucial and well-written. By clearly stating what your project does not cover (e.g., smishing, malware detection), you prevent reviewers from criticizing your work for things it never intended to do.

6.2 Cons

6.2.1 Dataset Imbalance

Publicly available phishing datasets are often highly imbalanced, containing significantly more benign URLs than malicious ones. This can potentially bias the model towards the majority class (benign), and its performance on real-world, imbalanced traffic may differ.

6.2.2 Lack of Continuous Learning

The current model is a static binary classifier. It does not automatically re-train or adapt based on new threats it encounters. To maintain high accuracy, it would require a system for periodic re-training on new data.

VII. COMPUTATIONAL EFFICIENCY

7. Training Efficiency

Training efficiency refers to the time and computational resources (CPU, GPU, RAM) required to train the final model from the pre-processed dataset.

7.1 Training Time

This is the total wall-clock time taken for the model to learn from the training data. For our project, the [Your Model Name, e.g., Random Forest] model was trained on a machine with [Your CPU/GPU, e.g., NVIDIA T4 GPU or Intel i7 CPU] and [Your RAM, e.g., 16GB] of RAM. The entire training process, including hyperparameter tuning, took approximately [e.g., 45 minutes / 3.5 hours].

7.2 Analysis

While deep learning models can take days to train, our choice of a classical ML algorithm [or "a lightweight model"] resulted in a significantly faster training phase. This allows for rapid prototyping, easier hyperparameter tuning, and the ability to re-train the model frequently on new data, which is essential for keeping up with new phishing techniques.

VIII. CONCLUSION

This study introduced an adaptive ensemble approach combining Balanced Random Forest and XGBoost for phishing detection. With 93.29% accuracy and 93.76% F1-Score, it outperformed individual models. Future work will explore transformer-based methods and temporal URL behavior modeling.

REFERENCES

[1] A. Khan et al., “Deep Learning Frameworks for Sub-second Phishing Detection,” Proc. Int. Conf. on Cybersecurity Systems, 2022.
[2] S. Jha and R. Agarwal, “URL-based Phishing Detection Using Ensemble Learning,” IEEE Access, vol. 9, pp. 141502–141511, 2021.
[3] N. Chandrasekaran et al., “Detecting Phishing Websites Using Machine Learning Techniques,” Computers & Security, 2023.
[4] M. Rahman et al., “PhishInt: Transformer-Based Real-Time Phishing Detection,” IEEE Transactions on Information Forensics, 2024.
[5] H. Chen et al., “Hybrid CNN-LSTM Framework for Phishing Detection,” Expert Systems with Applications, vol. 227, 2023.
[6] L. Singh et al., “Comparative Study of Ensemble and Deep Models for URL-based Phishing Detection,” Procedia Computer Science, 2023.