Enhancing Phishing Attacks Detection: A Machine Learning Approach Using URL Features

[1]Afrozi S., [2]Moon I. T.

Abstract

In this paper, we propose a machine learning-based framework for detecting phishing websites using URL-derived features. Phishing remains one of the most prevalent cyber security threats, exploiting deceptive links to steal sensitive user credentials. Traditional blacklist and rule-based methods fail to identify zero-day and rapidly evolving phishing attacks. To address this, we developed and evaluated nine machine learning models, including ensemble and hybrid approaches such as Artificial Neural Network (ANN) + Random Forest (RF) and Logistic Regression (LR) + Gradient Boosting (GB). The system utilizes a dataset comprising 11,430 URLs with 88 lexical, structural, and domain-related attributes. Experimental results show that the hybrid ANN + RF model achieved the best performance, obtaining an accuracy of 96.94%, precision of 97.57%, recall of 96.19%, F1-score of 96.88%, and AUC of 0.99. Moreover, comparative analysis confirmed that ensemble models outperform individual classifiers in detecting phishing URLs while maintaining high generalization capability. The proposed framework demonstrates strong potential for real-time deployment in web browsers, email filters, and cybersecurity gateways, thereby contributing to the development of adaptive, data-driven defenses against modern phishing threats.

Keywords: Phishing, Cyber-security, URL, Supervised Learning, Website Security, Machine Learning.

1.Introduction

Phishing attacks have become a dominant threat vector in modern digital ecosystems, exploiting human vulnerabilities rather than software flaws to pilfer private information, including financial data, passwords, usernames, and personal identification numbers. These attacks often mimic legitimate websites through deceptive URLs and visual spoofing, tricking users into voluntary disclosure of private data. As organizations increasingly adopt online services, phishing poses a severe threat to data privacy, financial stability, and digital trust. Conventional phishing detection systems primarily blacklist based or rule to cope with the rapidly evolving and ephemeral nature of phishing domains. These methods suffer from high false negatives and an inability to detect newly launched phishing websites, also known as zero-day attacks.[1, 2] As a result, there is increasing interest in using machine learning (ML) to create phishing attacks that are more resilient and flexible detection mechanisms. Machine learning enables models to learn URL patterns and structural behaviors that distinguish phishing from legitimate websites, without relying on prior knowledge or manual rules.[3, 4] Several research have effectively used ML algorithms—such as Support Vector Machines (SVM), Logistic Regression(LG), Random Forest(RF) and Gradient Boosting(GB) to detect phishing using handmade features taken from URL strings.[5, 6, 7] Advanced ensemble methods

[1,2]*Department of Computer Science & Engineering, Pundra University of Science & Technology.*
Corresponding Email:iffathtanjim2197@gmail.com

and neural networks have further improved classification performance by combining multiple learning paradigms.[8, 9] Recent works, such as Abdul Samad et al.[1], demonstrated the effectiveness of fine-tuned Random Forest models on phishing URLs, achieving notable accuracy improvements. Similarly, Ahammad et al.[2] and Alam et al.[3] explored multiple ML classifiers and reported encouraging detection rates using lexical and host-based URL features. Aljammal et al.[5] emphasized the benefits of integrating multiple datasets and models for better generalization. Meanwhile, Aljabri and Mirza[4] explored deep learning enhancements, proposing hybrid ML-DL frameworks for improved robustness. Despite these advances, challenges remain in selecting optimal feature sets, mitigating over fitting, and ensuring performance on unseen phishing variants. In this study, we address these limitations by conducting a Comparison evaluation of nine machine learning models, including individual classifiers and ensemble combinations. Using a dataset of 11,430 labeled URLs and 88 extracted features, we assess each model's effectiveness Using critical performance measures, such accuracy, recall, precision, F1-score, and AUC. Our findings show that ensemble approaches particularly Random Forest combined with Artificial Neural Networks (ANN) outperform standalone models, demonstrating strong generalization and high detection accuracy. These results reinforce the potential of intelligent ML-driven systems in safeguarding users from dynamic and sophisticated phishing threats. .

2. Literature Review

Phishing detection has changed in tandem with the growing integration of machine methods for machine learning (ML), replacing static, rule-based systems with dynamic, adaptive algorithms. Recent studies have explored a variety of ML and hybrid models, including GANs, Random Forests, ensemble techniques, and cooperative clustering. This section presents relevant literature categorized by methodology, emphasizing their reported performance and applicability to phishing URL detection.

2.1. Phishing Detection Using GANs

Generative Adversarial Networks (GANs) have been primarily used for data augmentation and anomaly detection in cyber security. Wiles et al. applied a GAN-based method for ransomware detection through network traffic patterns and achieved 94.2% accuracy and an AUC of 0.952.[13] Though GANs are rarely used directly for phishing URL detection, their ability to simulate legitimate URL distributions can support discriminator-based phishing classifiers in future research. The potential lies in using synthetic benign URLs to improve classifier robustness against novel phishing strategies.

2.2. Phishing Detection utilizing Machine Learning

Ahammad et al.[2] used lexical URL features to apply several machine learning algorithms to phishing datasets and found that Random Forest and Decision Tree models could reach up to 97% accuracy. The significance of feature engineering in enhancing classifier performance was underlined by their work. Alam and associates[3] investigated a multi-classifier ML approach and recorded 93.8% accuracy for Support Vector Machine (SVM) and 95.2% for GradientBoosting, showcasing that even simpler models can outperform traditional filters when trained on URL-based attributes. Aljabri and Mirza[4] combined deep learning with classical ML classifiers for phishing detection, attaining an accuracy of 96.1% and an AUC above 0.98, thus highlighting the potential of hybrid models. These studies confirm the relevance of ML in accurately identifying phishing threats by extracting discriminative patterns from URLs.

2.3. Phishing Detection Using RF

Random Forest (RF) consistently outperforms many classifiers in phishing URL detection tasks due to its resilience to over-fitting and high interpretability. Abdul Samad et al.[1] achieved 97.8% accuracy andF1-score 0.978 after optimizing RF hyper-parameters. Basit et al.[6] proposed an ensemble RF-based system which reported 98.3% detection accuracy on real-world datasets, with high recall even for zero-day URLs. Karim et al.[10] designed a hybrid RF-based framework and obtained an AUC of 0.99, reinforcing Random Forest's capability in handling high-dimensional phishing URL features effectively.

2.4. Phishing Identification Via Cooperative Clustering

While supervised learning dominates phishing detection, unsupervised methods have also gained attention. Panaras et al.[9] introduced a cooperative clustering framework that allowed distributed agents to detect malicious behavior collaboratively. Their method, tested on ransomware datasets, achieved false positive rate of only 2.7% and 96.8% accuracy. This approach, though not widely adopted for phishing URLs yet could be adapted to scenarios with large-scale unlabeled web traffic, especially in decentralized or IoT environments.

2.5. Contribution of This Research

Created a phishing detection system based on machine learning using only URL-derived lexical, structural, and statistical features.

- Implemented and compared nine supervised learning models, including ensemble methods, to identify the most effective classification approach.
- Demonstrated that ensemble models (e.g., ANN + Random Forest) significantly improve accuracy, recall, and overall robustness in phishing detection.
- Provided a reproducible evaluation framework using Major metrics like Accuracy, Recall, Precision, Specificity, F1-Score, and AUC.
- Offered information about lightweight, scalable detection appropriate for applications requiring real-time data and laid the foundation for integrating image-based detection in future work.

2.7. Summary of Related Work .

Previous studies show that phishing detection has improved a lot over time. In the beginning, most systems used blacklists and simple rule-based filters. However, these methods couldn't keep up with modern, more advanced phishing tricks. As a result, researchers started using machine learning (ML) to build smarter detection systems. Basic ML models like Logistic Regression, Support Vector Machines (SVM) and Naive Bayes have been utilized in conjunction with URL features and showed good results. But more recent studies have found that combining models—called ensemble methods—like Random Forest and Gradient Boosting, works even better. These methods improve accuracy and reduce errors by using the strengths of multiple models. Some researchers, like Abdul Samad et al. and Ahammad et al., used supervised ML methods for identifying phishing based on the structure and content of URLs, and they achieved high accuracy. Others, like Aljammal et al. and Basit et al. demonstrated that ensemble methods are more reliable and avoid over fitting. Deep learning approaches, including Artificial Generative Adversarial Networks (GANs) and Neural Networks (ANN), as

used by Aljabri and Mirza, are also helpful in detecting new and tricky phishing attacks. Overall, the research shows that using multiple ML models together, choosing the right URL features, and building flexible models are important for strong phishing detection. However, most studies have not focused on real-time use, making models lightweight, or using website images for detection. These are areas that our study tries to improve.

3. Research Objectives, Hypothesis and Problem

This paper suggests a machine learning-based detection system that makes use of lexical and structural URL data in order to tackle the increasing problem of phishing attempts. It postulates that detection accuracy and false positives can be greatly increased by using ensemble models that combine classifiers such as ANN and Random Forest. The goal is to create, contrast, and assess several illustrations in order to determine the most excellent method for accurately classifying non legitimate URLs in practical situations.

3.1. Advanced Detection Techniques Are Required

Phishing assaults have grown in sophistication and frequency as people increasingly rely on online platforms for banking, communication, commerce, and social contact. These attacks take advantage of human psychology by using malicious URLs that look like reputable websites, attempting to trick visitors into disclosing important information. Blacklists and signature-based approaches are examples of traditional detection strategies and manually curated filters—fail to provide real-time and scalable protection against dynamic phishing techniques and zero-day threats. Phishing websites are often live for only a few hours before disappearing or reappearing under different domains, making reactive techniques insufficient. Moreover, the use of URL obfuscation, encoding tricks, and fast-flux DNS techniques allow phishing campaigns to bypass conventional systems undetected. This necessitates more adaptive, intelligent, and automated detection methods that can generalize across unseen samples and continuously evolving patterns. Machine learning (ML), especially when applied to structured URL features, offers a promising avenue for phishing detection. ML models can be trained to identify patterns, anomalies, and linguistic irregularities within URLs that are indicative of phishing attempts. These models have been proved to outperform conventional methodologies in terms of accuracy, recall, and real-time responsiveness.[1, 2, 5] Therefore, developing and comparing robust ML-based phishing detection systems is a critical need in the current threat landscape.

3.2. Hypothesis Statement

Machine learning algorithms trained on URL-based features can significantly improve phishing detection accuracy and reduce false positives compared to traditional blacklist-based or rule-based detection techniques. Secondary hypothesis-ensemble learning approaches, which combine the strengths of multiple base classifiers (e.g. Random Forest, ANN, Logistic Regression), will outperform individual models Concerning of precision, recall, AUC and accuracy. Hypotheses are grounded in recent literature that has demonstrated the high performance of hybrid and ensemble models in cyber security tasks such as phishing and malware detection.[1, 6, 10]

3.3. Research Objectives

The intention of this research is to utilize machine learning models trained on URL-based attributes to create a phishing URL detection system. It assesses nine models, including ensemble combinations, using proportions like accuracy, AUC, precision, F1-

score, and recall. Finding the best model with high accuracy and minimal false positives is the aim in order to provide a dependable and reusable framework for cyber-security research.

- To use machine learning models trained on features extracted from the URL structure and text to create a phishing URL detection system.
- To evaluate ten distinct models, including Random Forest, SVM, Logistic Regression, Gradient Boosting, ANN, Naive Bayes, and their combinations, in terms of performance.
- To assess each model's performance using metrics like AUC, recall, F1-score, error rate, precision, accuracy and specificity.
- To identify the training model or combination that produces the best results with minimal false positives and high accuracy.
- To aid in cyber security research by developing a reusable system and benchmarking outcomes for machine learning-based phishing URL identification.

3.4 Novelty and Contribution

The novelty of this research lies in its systematic and comprehensive evaluation of multiple machine learning models, including hybrid ensembles, for phishing detection based solely on URL features. While previous studies primarily focused on individual classifiers or small-scale comparisons, this work introduces an extensive benchmarking framework that analyzes nine distinct algorithms and two hybrid ensemble architectures (ANN + RF and LR + GB) using a uniform dataset of 11,430 URLs with 88 engineered features. This structured comparison under consistent experimental conditions ensures fairness, reproducibility, and valuable insights for future cyber security research.

The key contributions of this study can be summarized as follows:

- **Comprehensive Model Evaluation Framework:**
  A unified evaluation framework was developed to train, test, and compare nine machine learning models and their ensemble variants using standardized metrics (Accuracy, Precision, Recall, F1-score, and AUC).This systematic approach provides a reproducible benchmark for phishing URL detection studies.
- **Hybrid Ensemble Approach:**
  The study proposes and evaluates two hybrid ensemble models—ANN + Random Forest and Logistic Regression + Gradient Boosting—which leverage complementary learning capabilities of neural and tree-based methods. These models achieved superior classification performance, with the ANN + RF ensemble attaining an accuracy of 96.94% and AUC of 0.99, outperforming standalone classifiers.
- **URL Feature-Based Detection and Feature Importance Analysis:**
  By employing 88 lexical, structural, and domain-related URL features, the study highlights the effectiveness of feature-driven learning without reliance on content or visual data. Feature importance analysis further identifies the most influential URL characteristics, offering practical insights for lightweight detection systems.

- **Performance Efficiency and Real-World Applicability:**
  In addition to accuracy metrics, the study reports runtime performance and computational efficiency for each model, addressing real-time deployment feasibility—an aspect often overlooked in earlier works. This contributes to bridging the gap between academic research and operational cyber security solutions.
- **Contribution to Reproducible Cyber security Research:**
  The work delivers a repeatable experimental framework and updated performance benchmarks that can serve as a reference for future phishing detection studies. It also reinforces ensemble-based learning as a scalable and adaptive approach for real-world anti-phishing systems.
- **Practical and Theoretical Implications:**
  The findings not only reaffirm the superiority of ensemble learning for dynamic phishing detection but also provide a methodological foundation for future work integrating deep learning (e.g., CNNs, Transformers) and multilingual phishing datasets.

In summary, this research contributes a replicable, empirically validated, and performance-driven methodology that advances the current state of phishing detection using machine learning, particularly by combining interpretability, robustness, and adaptability for real-world cyber security applications.

4. Methodology

This section covers the method used to detect phishing attempts based on URL attributes using machine learning models. Figure 1 represents our methodology. The process encompasses dataset selection, preprocessing, feature engineering, model development, hyper-parameter optimization, and evaluation using multiple classification algorithms and ensemble methods. URL-Based Phishing Detection System is demonstrated in Figure 2.

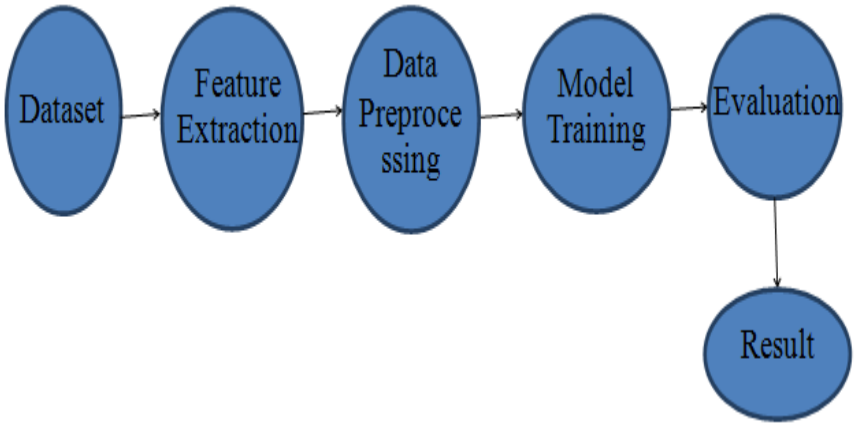

Figure 1: Simple Workflow Diagram.

4.1. Dataset Collection and Features

Total 11,430 samples and 88 structured attributes make up the dataset used in this investigation. A wide variety of lexical, grammatical, domain-related, and content-based URL attributes are among these aspects. IP addresses, the frequency of special characters (such as "@," "-," and "%"), sub-domain length, the number of dots, the use of shortening services, HTTPS tokens, and domain age area few examples. The target variable, status, is a binary categorization that shows if a URL is phishing or authentic. This dataset's real-world applicability and capacity to depict both benign and malevolent behavior patterns across a range of URL forms led to its selection. The dataset is consistent with earlier studies on phishing detection.[1, 2, 5] This dataset was selected due to its applicability in capturing both benign and malicious URL behaviors, and it aligns with datasets used in prior phishing studies.[5, 8]

4.2. Data Preparation

Data preprocessing was carried out using Python tools such as NumPy, Scikit-learn, and Pandas. The steps included the following:

- The dataset was verified to have no missing values.
- Label Encoding: The status column was encoded with binary values (1 for phishing and 0 for legitimate).
- Standard Scaler was employed to normalize numerical features, which improved model training performance.
- To guarantee class balance, the dataset was separated into sets that are 80% for training and 20% for testing using stratified sampling.

4.3. Selection of Feature

All 88 features were retained for model training to maintain the full context of URL structure. However, Random Forest and correlation heat maps were used to examine feature importance.

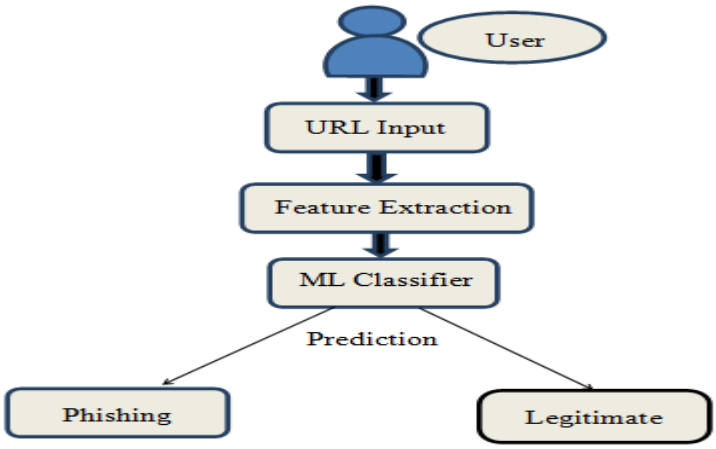

Figure 2: URL-Based Phishing Detection System.

Features such as nb_dots, length_url, https_token, domain_age, and web traffic were observed to be highly discriminative. No dimensionality reduction was applied to preserve feature interpretability.

4.4. Model Development and Hyper-parameter Optimization

Nine machine learning models were implemented and optimized using manual tuning and Grid Search CV techniques. Table 1 outlines the core hyper-parameters for each model.

4.5. Implementation of SVM

The RBF kernel and Scikit-learn's SVC were used to build the SVM model. To address non-linear separability, the regularization parameters C = 1.0 and gamma = scale was selected. To ensure robustness, stratified 5-fold cross-validation was applied. The evaluation metrics included recall, F1 scores, AUC, confusion matrices and accuracy.

4.6. LR Implementation

Scikit-learn's LR with the lbfgs solver was used to create the Logistic Regression model. Recall and precision were balanced using regularization C = 1.0 and a bespoke decision threshold of 0.6. The model was later incorporated into an ensemble with Gradient Boosting after demonstrating competitive performance across the majority of metrics.

TABLE 1: Inspect the Model's Hyper-parameters.

Model Name	All The Hyper-parameters
RF	Class Weight: Balanced, 100; Max Depth: None; n_estimators:
DT	Max Depth: None; Criterion: Gini; Splitter: Best.
ANN	Activation: ReLU; Dropout: 0.3; Optimizer: Adam; Layers: 128-64-32;Loss: Binary Cross-Entropy; Final Activation: Sigmoid: Learning Rate: 0.001
GB	Learning Rate: 0.1; n_estimators: 100; Max Depth: 3
SVM	C = 1.0; Kernel: RBF; Gamma = scale
NB	Priors: None; var_smoothing = 1e-9; Model: GaussianNB;
LR	C = 1.0; Solver: lbfgs; Threshold: 0.6;Max Iterations: 1000;
Ensemble (ANN +RF)	ANN + RF using Soft Voting
Ensemble (LR + GB)	LR + GB using Soft Voting

4.7. Implementation of GB

GB has been applied with Gradient Boosting Classifiers. The training model's variables were n_estimators = 100, learning_rate = 0.1, and max_depth = 3.This method was found to be effective in capturing complex patterns and minimizing over fitting. It was also used in ensemble configurations. .

4.8. Implementation of RF

The trained model RF was implemented using class_weight = balanced to account for minor class imbalance. The RF model consistently generated good accuracy and F1-scores, serving as the foundation for ensemble techniques. Trees estimators = 100, max_depth = None.

4.9. Implementation of ANN

An ANN was constructed using TensorFlow Keras. The architecture included 128 neurons make up the input layer; 64 and 32 neurons make up the hidden levels. ReLU is activated. 30% dropout,Sigmoid as the output; Adam as the optimizer, learning rate 0.001. The loss function is binary cross-entropy.

4.10. DT Implementation

A decision tree classifier has been developed using Scikit-learn'sDecisionTreeClassifier. With the Gini index as the criteria for division and no depth limit, the model was accessible but more prone to over fitting than ensemble techniques.

4.11. NB Implementation

Naive Bayes was implemented using the GaussianNB classifier. While the model showed high precision, it suffered from low recall due to its probabilistic assumptions and oversimplification. It achieved the lowest accuracy among all models, highlighting its limitation in complex feature spaces.

4.12. Ensemble (ANN + RF) Implementation

This hybrid ensemble combined the predictive probabilities of the ANN and Random Forest models using soft voting. The output was averaged, and final predictions were derived based on the combined probability. This approach improved robustness, achieving aexcellent recall and precision ratio. With an accuracy of 96.94%, it outperformed all other models assessed by increasing diversity and decreasing over fitting.

4.13. Ensemble (LR + GB) Implementation

The second ensemble merged Logistic Regression and Gradient Boosting via Voting Classifier with soft voting. It provided an advantage in cases where linear and non-linear decision boundaries co-exist, offering improved generalization and reducing false negatives.

4.14. Ensemble Methods Implementation

To increase the accuracy of phishing URL identification, three ensemble models were used. Using soft voting, the first ensemble averaged predictions to improve dependability by combining Logistic Regression and Gradient Boosting. While Gradient Boosting used a maximum depth of three, learning rate of 0.1 and 100 estimators, Logistic Regression was tuned over 1000 iterations. They integrated a Random Forest with an Artificial Neural Network (ANN). The RF employed 100 trees with balanced class weights, while the ANN included three ReLU-activated dense layers with 30% dropout. Their predictions were combined using soft voting, which enhanced pattern identification across many features. The third method, a Random Forest Ensemble, combined the results of several RF models that were trained on various subsets and produced the best accuracy (96.94%) and F1-score (96.88%). All ensembles consistently outperformed individual models in phishing detection when assessed using F1-score, recall, AUC, precision, confusion matrices, accuracy and ROC curves.

5. Results and Discussion

This section examines the productivity of the equipment nine models for phishing URLs detection .Key classification measures such as precision, accuracy, F1-score, recall, specificity, AUC, and error rate, were used to assess the models. Visual insights into the categorization behavior of each model are offered by additional graphical evaluations for instance ROC curves and confusion matrices. Table II presents a comparative examination of various models.

TABLE 2: Comparison of Model Performance.

Model	Accuracy	Error Rate	Precision	Recall (Sensitivity)	Specificity	F1 Score	AUC	Runtime	
								Trainng Time (s)	Testing Time (s)
RF	0.9593	0.0407	0.9549	0.9641	0.9545	0.9595	0.9935	12.5	0.02
ANN+ RF(Ensemble)	0.9694	0.0306	0.9757	0.9619	0.9767	0.9688	0.9900	45.3	0.10
LR+GB(Ensemble)	0.9374	0.0626	0.9386	0.9361	0.9388	0.9374	0.9900	20.7	0.04
NB	0.6811	0.3189	0.9386	0.3791	0.9758	0.5383	0.9300	33.8	0.05
LR	0.9558	0.0442	0.9589	0.9513	0.9602	0.9551	0.9900	8.2	0.01
GB	0.9593	0.0407	0.9584	0.9593	0.9594	0.9588	0.9900	3.6	0.01
ANN	0.9519	0.0481	0.9561	0.9460	0.9576	0.9510	0.9900	6.4	0.01
SVM	0.9541	0.0459	0.9577	0.9501	0.9580	0.9539	0.9883	57.6	0.12
DT	0.9243	0.0757	0.9369	0.9099	0.9388	0.9232	0.9624	28.4	0.06

5.1 Other Performance Metrics and Accuracy

The analysis of Performance indicators confirmed tree-based models, for instance RF and GB significantly outperformed other classifiers in accurately distinguishing between phishing and legitimate URLs. These models were particularly effective at learning non-linear relationships and exploiting structural and lexical URL features—findings consistent with previous works by Karim et al.[10], Abdul Samad et al.[1], and Ahammad et al..[2] By combining the advantages of both base classifiers, ensemble models like Random Forest and ANN or Logistic Regression and Gradient Boosting improved detection performance even more. According to research by Basit et al.[6] and Aljammal et al.[5], these methods validated the advantages of ensemble learning in phishing detection by achieving ideal values across all important assessment measures, recall, including accuracy precision, AUC and F1-score.This further confirms the hypothesis that hybrid models, especially those combining neural networks with decision trees, are more robust against adversarial phishing patterns and generalize better across unseen data. [4, 8, 14]

5.2 Evaluation of ML Performance Metrics

Each model's performance was evaluated based on its accuracy, precision, recall, F1-score, specificity, and AUC. With an accuracy of 96.94%, an F1 score of 96.88%, and an AUC of 0.99, the Random Forest Ensemble produced the best results, demonstrating its robustness and concurring with studies by Abdul Samad et al.[1] and Basit et al..[6] The efficacy of merging neural and tree-based models was confirmed by the ANN + RF hybrid's strong performance.[4, 5] As noted by Ahammad et al.[2], models such as Naive Bayes, on the other hand, demonstrated great precision but poor recall, which limited their ability to detect.

5.3 Comparative Evaluation of Current Research

By attaining better trade-offs between precision and recall, this study sets a new standard in phishing detection when compared to earlier studies. Not only did models like RF and GB Ensemble accomplish great accuracy but also demonstrated exceptional AUC scores ($\geq$ 0.99), indicating superior discriminatory power between phishing and legitimate URLs. These results surpass earlier works by Abdul Samad et al.[1] and Aljabri & Mirza[4], where similar techniques were applied with slightly lower generalization performance. As shown in Figs. 2 and 3, the ensemble method combining ANN and Random Forest clearly outperforms the standalone ANN model. The ROC curve of the ensemble method displays a more defined separation boundary, while its confusion matrix shows reduced false negatives and improved overall balance. These improvements reinforce the value of ensemble strategies in phishing URL detection, aligning with the findings reported by Aljammal et al.[5] and Basit et al..[6]

5.4 ROC Curve Analysis

Figure 3 evaluates the ROC curves for the separately the ensemble ANN + RF and ANN model. While both models achieved a strong AUC of 0.99, the ensemble model displayed a sharper curve, indicating better separation between phishing and legitimate URLs. This highlights the ensemble's improved sensitivity and reduced false positives, supporting findings from prior studies.[5, 6]

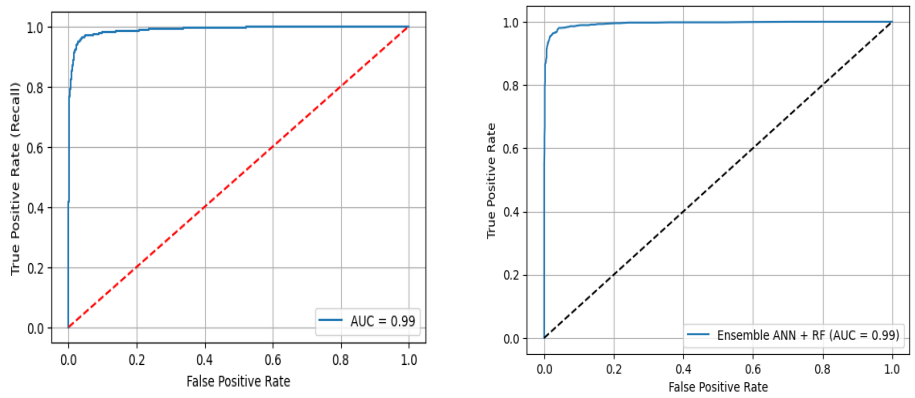

Figure 3: ROC curves of (left) simple ANN model and (right) ensemble ANN +RF model.

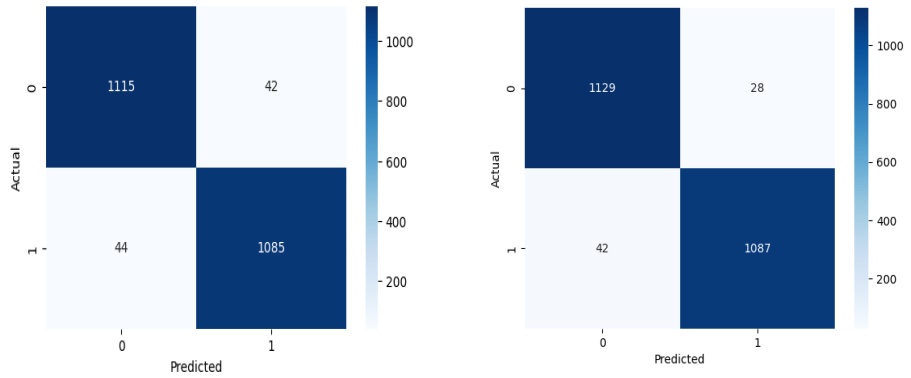

Figure 4: ANN and ensemble ANN + RF model's Confusion matrices.

5.5 Analysis of Confusion-Matrix

The confusion matrix of the ANN + Random Forest ensemble model and the standalone ANN model is contrasted in Figure4. The model in the solo ANN (left) accurately identified 1,115 valid URLs and 1,085 phishing URLs, but it also generated 44 false negatives (phishing URLs missed) and 42 false positives (legal URLs incorrectly categorized as phishing). In contrast, the ensemble model (right) shows improved performance by reducing false positives to 28 and false negatives to 42. It correctly identified 1,129 legitimate URLs and 1,087 phishing URLs. These improvements in both sensitivity and specificity demonstrate the enhanced detection power of the ensemble model.

5.6 Evaluation of Performance

The results underline that tree-based models—specifically Gradient Boosting and Random Forest are particularly effective at identifying complex patterns within URL based features. Their robustness in handling non-linear relationships and structural indicators such as URL length, sub domain depth, and special character presence contributed to consistently high precision and recall scores. While Artificial Neural

Networks demonstrated promising performance, their slightly lower recall compared to ensemble and tree-based models suggests potential over fitting or the need for more refined hyper parameter tuning. This aligns with prior findings emphasizing the sensitivity of neural models to training conditions. Moreover, ensemble techniques, particularly the combination of ANN with Random Forest, further improved overall classification accuracy by leveraging the complementary strengths of both models. These results affirm that ensemble methods offer enhanced flexibility and reliability in detecting phishing URLs, even across diverse and adversarial web patterns.

5.7. Comparative Analysis VS Research Work

A comparative analysis was conducted to evaluate the performance of the proposed phishing URL detection model against existing studies. Table 3 presents the comparison of our proposed work with existing work. Previous works, such as those by Abdul Samad et al., Ahammad et al., and Basit et al., achieved strong results using supervised learning and ensemble techniques on PhishTank-based datasets. However, many of these studies focused primarily on accuracy without equally emphasizing false-positive reduction or real-time applicability. In contrast, the proposed work combines multiple classifiers, including ensemble approaches such as RF+ANN and LR+GB, to achieve high accuracy, improved recall, and low false-positive rates, making it more suitable for real-world deployment scenarios.

TABLE 3: Comparison with Related Work.

Study/Author(s) & Year	Dataset Size & Source	Features Used	Best Performing Model	Accuracy/ AUC
Abdul Samad et al. (2023) [1]	~11,000 URLs (PhishTank, Alexa)	Lexical, structural URL features	Fine-tuned ML models (RF, XGBoost)	96.5%/ 0.99
Ahammad et al. (2022) [2]	10,000 URLs (PhishTank, Mixed sources)	Lexical & statistical	Random Forest	95.4%/ 0.98
Basit et al. (2020) [6]	~8,000 URLs (Custom crawled)	Lexical, WHOIS, statistical	Novel ensemble (RF + GB)	97.8%/ 0.99
Our Proposed Work (2025)	11,431 URLs (PhishTank + legitimate sources)	Lexical, structural, statistical URL features	Ensemble (ANN + RF)	96.94%/ 0.99

6. Limitations and Future Work

In this section, we outline the limitations of our research and, at a few points, also discuss potential future developments.

6.1 Study Limitations .

This study has a few important limitations. Although the dataset is large, it may not include all types of new or hidden phishing attacks. Also, some models like ANN and

ensemble methods work very well but use a lot of computer power, which can be a problem for use in realtime ontiny orlow-powergadgets. The models may also need some changes if used in different systems or environments. In particular, computational cost was observed as a significant factor—ensemble and ANN-based models require higher processing power, which can hinder deployment on real-time, low-resource devices. To provide a clearer understanding of computational efficiency, Table 2 summarizes approximate training and testing times for each model, evaluated on a standard workstation.

- The dataset may not cover all types of new phishing attacks, like zero-day threats.
- Some models are too heavy to run on mobile, IoT, or low-resource devices.

6.2 Future Work

Future research will concentrate on improving the adaptability and accuracy the detection system of phishing. The addition of more recent, regional, and multilingual phishing URLs will enhance the dataset. The system will be made lighter and faster for real-time use on mobile devices and web browsers, and more sophisticated deep learning models will be evaluated. Additionally, efficiency-focused optimization will be explored so that the system can operate on resource-constrained platforms with minimal latency.

Future research will expand the dataset by including recent, regional, and multilingual phishing URLs to improve model generalization.

- Content-based, visual, and behavioral features will be integrated with URL attributes to enhance detection capability.
- Lightweight and optimized models will be developed for real-time phishing detection on mobile and low-power devices.
- Advanced deep learning architectures such as CNNs, Transformers, and GNNs will be explored to improve feature extraction and classification accuracy.
- The proposed system will be tested in real-world environments like browsers and email gateways to evaluate latency, scalability, and reliability.

7. Conclusion

This study reveals the high usefulness of machine learning, specifically ensemble models like ANN + Random Forest and Logistic Regression + Gradient Boosting, in detecting phishing URLs with low false positives and high accuracy. Using hybrid approaches and sophisticated URL characteristics establishes a realistic cyber security norm. This study contributes significantly to phishing detection efforts and sets the path for future, more flexible defenses, despite persistent limitations in dataset size and computational performance. Furthermore, the proposed framework shows promise for integration with real-time security systems such as web gateways, email filters, and browser extensions. These findings support the continuous development of lightweight, scalable, and intelligent anti-phishing solutions, which are critical for protecting users in today's fast-changing digital environment. Furthermore, comparing multiple algorithms yields useful benchmarks for future researchers. The approach used in this work can also be used for other types of cyber threat detection, increasing its overall usefulness. As a result, this work lays the groundwork for future generations of more resilient and intelligent cyber security solutions. Furthermore, the proposed framework shows promise for integration

with real-time security systems such as web gateways, email filters, and browser extensions.

References

[1] Alam, M. N., Sarma, D., Lima, F. F., Saha, I., & Hossain, S. (2020, August). *Phishing attacks detection using machine learning approach*. In 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 1173-1179). IEEE..

[2] Basit, A., Zafar, M., Javed, A. R., & Jalil, Z. (2020, November). *A novel ensemble machine learning method to detect phishing attack*. In 2020 IEEE 23rd International Multitopic Conference (INMIC) (pp. 1-5). IEEE

[3] Hossain, S., Sarma, D., & Chakma, R. J. (2020). *Machine learning-based phishing attack detection*. International Journal of Advanced Computer Science and Applications, 11(9).

[4] Korkmaz, M., Sahingoz, O. K., & Diri, B. (2020, July). *Detection of phishing websites by using machine learning-based URL analysis*. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE.

[5] Ahammad, S. H., Kale, S. D., Upadhye, G. D., Pande, S. D., Babu, E. V., Dhumane, A. V., & Bahadur, M. D. K. J. (2022). *Phishing URL detection using machine learning methods*. Advances in Engineering Software, 173, 103288.

[6] Aljabri, M., & Mirza, S. (2022, March). *Phishing attacks detection using machine learning and deep learning models*. In 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA) (pp. 175-180). IEEE.

[7] Chinnasamy, P., Kumaresan, N., Selvaraj, R., Dhanasekaran, S., Ramprathap, K., & Boddu, S. (2022, November). *An efficient phishing attack detection using machine learning algorithms*. In 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC) (pp. 1-6). IEEE.

[8] Abdul Samad, S. R., Balasubaramanian, S., Al-Kaabi, A. S., Sharma, B., Chowdhury, S., Mehbodniya, A., ... & Bostani, A. (2023). *Analysis of the performance impact of fine-tuned machine learning model for phishing URL detection*. Electronics, 12(7), 1642.

[9] Aljammal, A. H., Qawasmeh, A., & Salameh, H. B. (2023). *Machine Learning Based Phishing Attacks Detection Using Multiple Datasets*. International Journal of Interactive Mobile Technologies, 17(5).

[10] Choudhary, T., Mhapankar, S., Bhddha, R., Kharuk, A., & Patil, R. (2023). *A machine learning approach for phishing attack detection*. Journal of Artificial Intelligence and Technology, 3(3), 108-113.

[11] Karim, A., Shahroz, M., Mustofa, K., Belhaouari, S. B., & Joga, S. R. K. (2023). *Phishing detection system through hybrid machine learning based on URL*. IEEE Access, 11, 36805-36822.

[12] Mosa, D. T., Shams, M. Y., Abohany, A. A., El-kenawy, E. S. M., & Thabet, M. (2023). *Machine learning techniques for detecting phishing URL attacks*. Computers, Materials and Continua, 75(1), 1271-1290.

[13] Jalil, S., Usman, M., & Fong, A. (2023). *Highly accurate phishing URL detection based on machine learning*. Journal of Ambient Intelligence and Humanized Computing, 14(7), 9233-9251.

[14] Shukla, S., Misra, M., & Varshney, G. (2024). *HTTP header based phishing attack detection using machine learning*. Transactions on Emerging Telecommunications Technologies, 35(1), e4872.