International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

Intelligent Phishing Website Detection Using Machine Learning for Secure Online Systems

Sagar Kumar[1], Harish Dutt Sharma[2], Ram Bhawan Singh[2]

[1] Research Scholar, School of Computer Engineering and Applications, Maya Devi University, Dehradun – 248011
[2] School of Computer Engineering and Applications, Maya Devi University, Dehradun – 248011, India.

Email-ID: sksolanki774@gmail.com
Email-ID: sharma.harish106@gmail.com
Email-ID: rambhawansingh@gmail,com

Conflicts of interest: Nil
Corresponding author: Harish Dutt Sharma

Abstract— Phishing attacks have emerged as one of the most significant cybersecurity threats, targeting users by creating fraudulent websites that mimic legitimate platforms to steal sensitive information. Traditional rule-based and blacklist-based detection techniques are often ineffective against newly generated phishing websites. This paper proposes a machine learning-based phishing website detection system that utilizes multiple classification algorithms to identify malicious URLs. The system extracts various URL-based and domain-based features such as URL length, presence of special characters, domain age, and HTTPS usage. Machine learning models including Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR) are evaluated. Experimental results demonstrate that the proposed approach achieves high accuracy and outperforms traditional detection methods.

Keywords- Phishing Detection, Machine Learning, Cyber Security, URL Analysis, Random Forest, SVM, Classification

I. INTRODUCTION

The rapid growth of internet services, online transactions, and web-based applications has significantly increased the dependency on digital platforms for communication, commerce, and data exchange. This widespread adoption of internet technologies has also led to a substantial rise in cyber threats, particularly phishing attacks, which exploit user trust to obtain sensitive information such as login credentials, financial data, and personal details. Phishing websites often mimic legitimate platforms, making it difficult for users to distinguish between authentic and malicious sources [1].

Despite advancements in web security, traditional phishing detection techniques primarily rely on blacklist-based and rule-based approaches. These methods are limited in their effectiveness as they depend on previously identified malicious URLs and predefined patterns. As a result, they fail to detect newly generated phishing websites, also known as zero-day attacks. Moreover, the dynamic and evolving nature of phishing strategies, including URL obfuscation and domain spoofing, further complicates the detection process using conventional methods [2].

Phishing detection systems play a critical role in ensuring secure online interactions by analyzing website characteristics and identifying potential threats. Conventional approaches, including heuristic-based and classical machine learning techniques, often depend on manually engineered features and static rules. These methods struggle to adapt to the rapidly changing landscape of phishing attacks and may result in reduced detection accuracy and increased false positives, especially in large-scale and real-time environments [3].

Recent advancements in machine learning have demonstrated significant potential in addressing complex cybersecurity challenges. Machine learning models can automatically learn patterns from large datasets and identify hidden relationships within URL and domain features. This capability enables the detection of previously unseen phishing attacks without relying on predefined rules. Techniques such as Support Vector Machines, Random Forest, and Logistic Regression have been widely used for classification tasks in phishing detection systems, offering improved accuracy and adaptability [4], [5].

However, integrating machine learning models into real-time phishing detection systems while maintaining high accuracy and low computational overhead remains a challenging task. Factors such as feature selection, model optimization, and scalability must be carefully considered to ensure effective deployment in real-world scenarios.

This paper addresses these challenges by proposing a machine learning-based phishing website detection framework that leverages multiple classification algorithms to analyze URL-based and domain-based features. The proposed approach aims to enhance detection accuracy while ensuring efficient performance in identifying malicious websites. The system is designed to classify websites as legitimate or phishing based on learned patterns, thereby improving overall cybersecurity.

The main contributions of this work are summarized as follows:

- Development of a machine learning-based phishing detection framework for accurate classification of websites.
- Implementation and comparison of multiple machine learning models, including SVM, Random Forest, and Logistic Regression.
- valuation of the proposed system using benchmark datasets, demonstrating improved detection performance over traditional approaches.

The system architecture presented in Figure 1 illustrates the overall structure of the proposed phishing detection framework, clearly depicting the interaction among its key components, including data collection, feature extraction, preprocessing, and classification modules. It provides a comprehensive representation of how input URLs are systematically processed through multiple stages, beginning with the collection of raw data from various sources. The extracted features are then refined and normalized during preprocessing to ensure consistency and accuracy.

These processed features are subsequently fed into machine learning models, which analyze patterns and relationships associated with phishing behavior. The architecture also highlights the smooth flow of data across different stages, enabling efficient processing, improved decision-making, and accurate classification of websites as either legitimate or phishing in real-time environments.

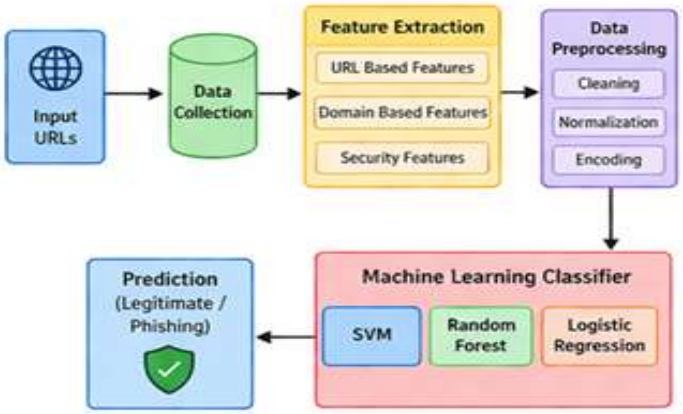

Figure 1: System Architecture of Proposed Framework

II. RELATED WORK

Phishing detection in web-based systems has been extensively studied using both traditional machine learning and recent data-driven approaches. Early research primarily relied on classical machine learning techniques such as Support Vector Machines (SVM), Decision Trees, and Random Forests to classify websites based on handcrafted URL and content features. While these approaches demonstrated reasonable performance, their effectiveness is often limited in dynamic environments due to their dependence on manual feature engineering and inability to generalize across evolving phishing strategies [6].

With the increasing complexity of phishing attacks, several studies have explored advanced detection mechanisms tailored to web security environments. In, the authors presented a phishing detection system based on URL analysis and heuristic rules to improve identification accuracy [7]. However, the approach relies heavily on predefined patterns and does not fully exploit the capability of automated feature learning. Similarly, proposed a detection mechanism using statistical analysis of URL features, but its performance is constrained when dealing with obfuscated URLs and complex phishing patterns [8]. Further improvements in phishing detection using machine learning techniques were discussed in, highlighting the importance of feature-rich models [9].

Recent research has increasingly focused on machine learning techniques for phishing detection due to their ability to learn complex patterns from large datasets. In, a machine learning-based model was developed to automatically extract features from URLs and webpage content, demonstrating improved detection accuracy over traditional approaches [10]. Furthermore, ensemble learning models such as Random Forest

have shown promising results in handling high-dimensional data and improving classification robustness [11]. These models are particularly effective in identifying subtle variations in phishing URLs and detecting previously unseen attacks. Feature selection techniques also play a crucial role in improving model performance, as discussed in [12].

Several studies have also explored probabilistic and content-based approaches for phishing detection. In, a Bayesian-based method was proposed to analyze both textual and visual features of web pages for improved detection accuracy [13]. Additionally, streaming-based phishing detection frameworks have been introduced to handle real-time data efficiently, as demonstrated in [14].

In addition, efforts have been made to integrate intelligent phishing detection systems into real-time environments. The work in proposed a machine learning-driven detection framework capable of analyzing incoming URLs in real time [15]. Although this approach improves detection performance, challenges related to computational overhead, feature selection, and scalability remain significant concerns.

Recent advancements have also incorporated semantic analysis and intelligent feature selection methods to enhance detection accuracy [16]. Public datasets such as those provided by online repositories have played a crucial role in training and evaluating phishing detection models [17]. Moreover, URL-based machine learning models have demonstrated strong performance in identifying phishing attacks with minimal feature extraction overhead [18].

Lightweight phishing detection systems suitable for real-time deployment have also been proposed, focusing on reducing computational complexity while maintaining accuracy [19]. Ensemble-based approaches, particularly those using Random Forest classifiers, have shown improved robustness and classification performance [20].

Despite these advancements, existing solutions often lack a balanced consideration of detection accuracy, computational efficiency, and real-time deployment constraints in practical environments. Recent studies [21], [22] emphasize the need for scalable and adaptive phishing detection frameworks capable of handling large-scale and evolving cyber threats.

III. SYSTEM MODEL AND PROBLEM FORMULATION

This section presents the system model of the phishing website detection framework and formulates the classification problem addressed in this work. The proposed system is designed to analyze website-related features and accurately distinguish between legitimate and phishing websites using machine learning techniques. It focuses on improving detection accuracy while maintaining efficiency for real-time deployment.

3.1 System Model

The proposed phishing detection system consists of multiple components, including data collection, feature extraction, and classification modules. The system processes website URLs and associated attributes to identify patterns that indicate phishing behaviour.

Let the dataset be represented as:

$$D = (X, Y) \quad (1)$$

where X *denotes* the set of input feature vectors and Y represents the corresponding class labels.

Each website instance is characterized by a set of extracted features:

$$X = \{x\_1, x\_2, x\_3, \ldots, x\_n\} \quad (2)$$

These features include URL-based, domain-based, and security-related attributes such as URL length, presence of special symbols (e.g., "@", "-"), number of subdomains, use of HTTPS protocol, domain registration age, redirection behavior, and presence of IP address in the URL. These attributes are critical in identifying suspicious patterns commonly associated with phishing websites.

The system architecture is designed to preprocess raw input data, normalize feature values, and feed the processed data into a trained machine learning classifier. The centralized processing of features enables efficient analysis and consistent decision-making across different types of web inputs.

3.2 Problem Formulation

The phishing detection task is formulated as a binary classification problem. Given a feature vector X, the objective is to determine whether the corresponding website is legitimate or phishing.

$$y = \{0, Legitimate\ Website;\ 1, Phishing\ Website\} \quad (3)$$

The classification problem can be defined as learning a mapping function:

$$H : X -> y \quad (4)$$

that accurately predicts the class label for unseen data. The model must generalize well across different types of phishing attacks, including newly generated and previously unseen patterns. This requires robust feature representation and effective learning algorithms capable of handling diverse and high-dimensional data.

3.3 Objective Function

The primary objective of the proposed system is to minimize classification error while ensuring efficient and timely detection. This can be expressed as:

$$min\ L(y, y\hat{}) + \lambda T\_d \quad (5)$$

where L represents the classification loss function, y^ is the predicted label, Td denotes detection time, and λ is a weighting parameter that balances accuracy and computational efficiency.

The loss function is typically based on classification error metrics such as cross-entropy loss, which measures the difference between predicted and actual labels. Minimizing this loss ensures that the model improves its prediction capability over time. Additionally, reducing detection time is essential for real-time applications where quick decision-making is required to prevent user interaction with phishing websites.

The detailed block diagram in Figure 2 presents the internal structure of the system, illustrating the flow of data between different modules involved in feature extraction, preprocessing, and classification.

IV. PROPOSED METHODOLOGY

This section presents the design of the proposed machine learning-based phishing website detection framework. The framework is developed to effectively identify malicious websites by analyzing various URL-based and domain-based features. It leverages classification models to detect phishing patterns and improve overall detection accuracy.

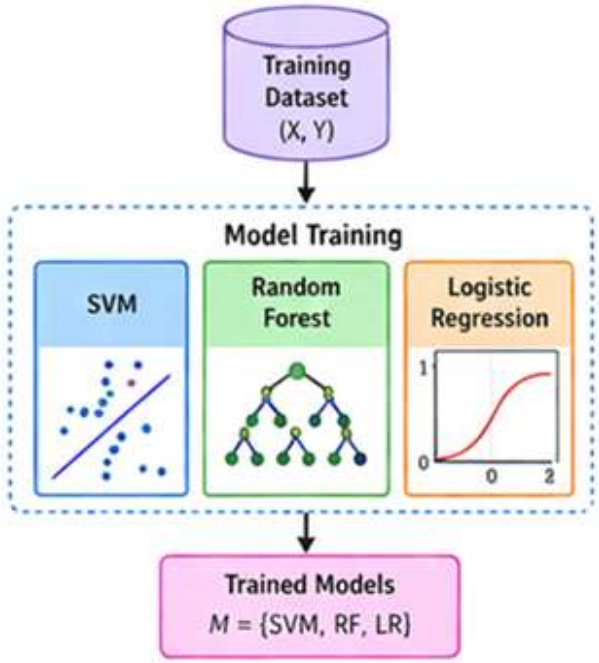

Figure 2: Machine Learning Process

4.1 System Architecture

The proposed framework is designed to operate as a centralized detection system that analyzes incoming website URLs. The system consists of data collection, feature extraction, preprocessing, and classification modules.

The input URLs are collected from various sources and passed through a feature extraction module, where relevant attributes such as URL structure, domain information, and security indicators are obtained. These features are then processed and fed into the machine learning model for classification.

The centralized architecture enables efficient processing of large volumes of web data and provides consistent detection performance. This design ensures scalability and supports real-time phishing detection in practical environments.

As shown in Figure 3, the workflow of the proposed system follows a step-by-step process from input URL to final classification.

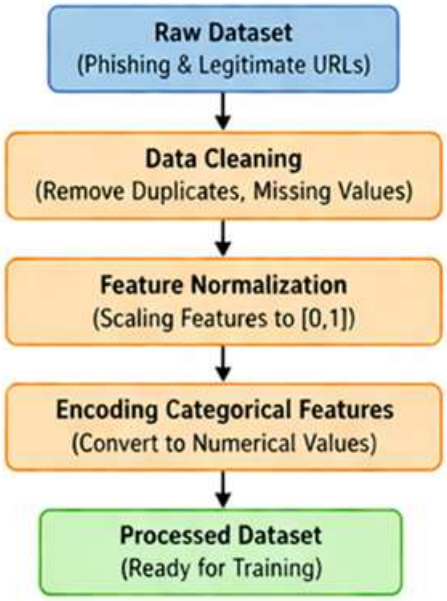

Figure 3: Data Preprocessing Steps

The methodology of the proposed system is divided into the following steps:

Step 1: Data Collection

Phishing and legitimate URLs are collected from publicly available datasets.

Step 2: Feature Extraction

Relevant features such as URL length, special characters, HTTPS usage, and domain age are extracted.

Step 3: Data Preprocessing

The dataset is cleaned, normalized, and encoded for training.

Step 4: Model Training

Machine learning models including SVM, Random Forest, and Logistic Regression are trained.

Step 5: Prediction and Classification

The trained model predicts whether a URL is phishing or legitimate.

Step 6: Real-Time Detection

Incoming URLs are analyzed and classified using the trained model.

4.2 Data Preprocessing

The collected dataset is preprocessed before being used for training the machine learning models. This step involves data cleaning, normalization, and encoding of categorical features.

Let the input feature vector be represented as:

$$X = \{x1, x2, x3, \ldots, xn\} \quad (6)$$

Each feature is scaled and normalized to ensure equal contribution during the training process. Missing values and redundant attributes are removed to improve model efficiency. Data preprocessing enhances model performance and reduces bias in feature representation.

4.3 Machine Learning Model

The proposed framework utilizes multiple machine learning algorithms to classify websites as phishing or legitimate. These models are trained on labeled datasets and evaluated based on their classification performance.

4.3.1 Feature Analysis

The system extracts meaningful features from URLs and domains. These include:

- URL length
- Presence of special characters
- Number of subdomains
- HTTPS usage
- Domain age
- Redirection $behavior$

These features help in identifying patterns commonly associated with phishing websites.

4.3.2 Classification Models

The following machine learning models are used:

- Support Vector Machine (SVM)
- Random Forest (RF)
- Logistic Regression (LR)

Each model is trained to learn the relationship between extracted features and the corresponding class labels.

The prediction function can be represented as:

$$\hat{y} = M(X) \quad (7)$$

where M represents the trained model and ŷ is the predicted output.

4.3.3 Model Advantages

The use of multiple machine learning models provides the following benefits:

- Ability to capture complex relationships between features
- Improved generalization across different phishing patterns
- Enhanced detection accuracy and robustness

4.4 Training and Optimization

The models are trained using labeled datasets in a supervised learning environment. The objective is to minimize the classification loss function.

$$L = -\Sigma\,(yi\ log(\hat{y}i)) \quad (8)$$

where yi represents the actual label and ŷi represents the predicted probability.

Optimization techniques such as gradient descent are used to update model parameters. Regularization methods are applied to prevent overfitting and improve generalization performance.

4.5 Detection Mechanism

The detection mechanism ensures accurate classification of incoming URLs by analyzing extracted features through trained models, enabling timely identification of phishing websites and enhancing overall system security in real-time environments.

The training and real-time detection process illustrated in Figure 4 demonstrates how the proposed system effectively learns from historical data and applies this knowledge to identify phishing websites in real-time environments. During the training phase, labeled datasets are used to train machine learning models, enabling them to recognize patterns associated with both legitimate and phishing URLs. In the detection phase, incoming URLs are processed and analyzed using the trained models to predict their legitimacy. The system continuously evaluates each URL based on extracted features and applies decision thresholds for classification. This integrated approach ensures timely detection, reduces the risk of cyber threats, and enhances the overall reliability and efficiency of the phishing detection framework.

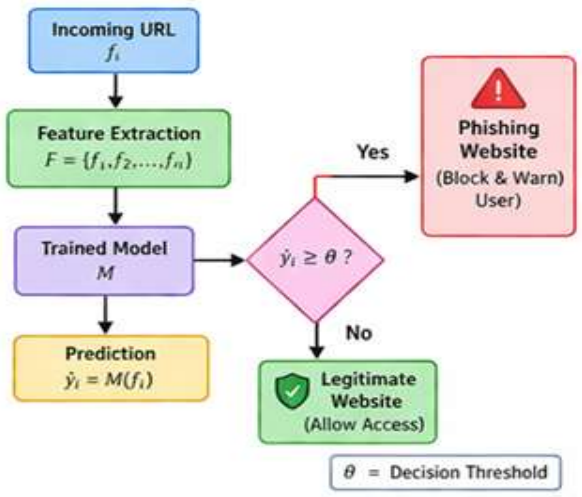

Figure 4: Detection Machine Flow

During the deployment phase, incoming URLs are continuously analyzed by the system. Let:

$$F = \{f1, f2, \ldots, fN\} \quad (9)$$

represent the set of input URLs.

Each URL is processed and classified using the trained model:

$$\hat{y}i = M(fi) \quad (10)$$

where ŷi represents the probability of the URL being phishing.

A decision threshold θ is used to classify the result:

$$Class(fi) = Phishing, if\ \hat{y}i \geq \theta$$
$$Legitimate, otherwise \quad (11)$$

4.5.1 Real-Time Detection

The system can be integrated into browsers or security tools for real-time phishing detection. Upon detection:

- Malicious URLs are blocked
- Suspicious links are flagged
- Users are warned before accessing unsafe websites

4.5.2 Efficiency Considerations

The total processing time is given by:

$$Tproc = Tfeat + Tinfer \quad (12)$$

where $Tfeat$ is feature extraction time and Tinfer is model inference time.

The system is designed to ensure:

$$Tproc \leq Tth \quad (13)$$

where Tth *is* the acceptable response time.

4.5.3 Performance Balance

The framework maintains a balance between accuracy and efficiency through:

- Optimized feature selection
- Efficient machine learning models
- Adaptive threshold tuning

This ensures reliable phishing detection with minimal computational overhead.

The real-time detection process is illustrated in Figure 4.

V. EXPERIMENTAL SETUP AND RESULTS

This section evaluates the performance of the proposed machine learning-based phishing website detection framework. The evaluation focuses on classification accuracy, detection performance, and computational efficiency.

5.1 Dataset Description

The proposed model is evaluated using publicly available phishing website datasets obtained from sources such as UCI Machine Learning Repository and Kaggle, which contain a large collection of phishing and legitimate URLs. These datasets reflect real-world web environments and include diverse phishing patterns such as URL obfuscation, domain spoofing, and redirection-based attacks. The dataset consists of labeled instances categorized as phishing or legitimate websites.

5.2 Evaluation Metrics

The performance of the model is assessed using standard classification metrics, including accuracy, precision, recall, and F1-score. These metrics are computed based on true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). These evaluation measures provide a comprehensive understanding of the model's detection capability and reliability.

5.3 Performance Comparison

Figure 5 presents the comparison of different machine learning models in terms of accuracy. These evaluation metrics provide a comprehensive and detailed analysis of the classification performance of each model used in the proposed phishing detection framework. Accuracy represents the overall correctness of the model in classifying both phishing and legitimate websites. it is evident that the proposed approach achieves superior performance across all evaluation metrics. This demonstrates that the model is highly reliable, efficient, and well-suited for accurately detecting phishing websites in real-world scenarios while maintaining a balance between detection accuracy and error minimization.

Figure 5: Accuracy Comparison of Models

The accuracy comparison between the different machine learning models is shown in Figure 5. It can be clearly seen that the Random Forest model consistently outperforms other baseline methods due to its ability to capture complex feature interactions and handle high-dimensional data effectively. This improved performance highlights the robustness and suitability of the Random Forest classifier for phishing detection tasks.

5.4 Metric Analysis

Three machine learning models are compared in Figure 6 in terms of precision, recall, and F1-score: SVM, Logistic Regression, and Random Forest.

From the figure, the SVM model achieves a precision of 0.91, recall of 0.90, and F1-score of 0.90, indicating a well-balanced performance with slightly higher precision.

The Logistic Regression model shows comparatively lower performance, with precision of 0.89, recall of 0.88, and F1-score of 0.88, suggesting it is less effective in detecting phishing websites compared to the other models.

In contrast, the Random Forest model demonstrates the best performance among all models, achieving the highest precision (0.95), recall (0.94), and F1-score (0.94). This indicates that Random Forest is more effective in identifying phishing websites while minimizing both false positives and false negatives.

Overall, the results clearly show that the Random Forest model outperforms SVM and Logistic Regression across all evaluation metrics, making it the most reliable model for phishing website detection in the proposed system. The consistently high F1-score further confirms its robustness and balanced classification capability in real-world scenarios.

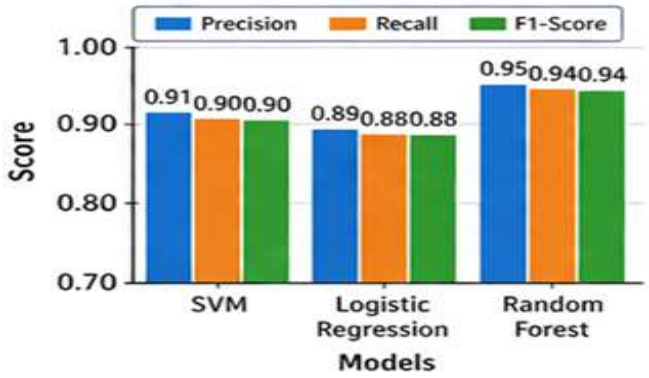

Figure 6: Precision, Recall and F1-Score Comparison

5.5 Latency Analysis

An illustration of the detection latency of the proposed phishing detection system can be found in Figure 7. Detection latency refers to the amount of time required by each model to process input data and generate classification results for determining whether a website is legitimate or phishing. It is a critical factor in evaluating the efficiency and responsiveness of the system, particularly in real-time environments where immediate decision-making is essential to prevent user interaction with malicious websites. Lower latency ensures faster response times, thereby enhancing user safety and overall system performance. However, achieving low latency while maintaining high detection accuracy can be challenging, especially for complex models. Therefore, latency analysis plays a significant role in understanding the trade-off between computational cost and model effectiveness. By comparing the latency of different machine learning models, the proposed system demonstrates its ability to maintain an optimal balance between speed and accuracy. This ensures that the system remains practical, reliable, and suitable for deployment in real-world phishing detection applications.

Figure 7: Detection Latency Comparison

From the results, the Logistic Regression model achieves the lowest latency of 35 ms, making it the fastest model in terms of prediction time. The SVM model has a moderate latency of 40 ms, indicating a slightly higher computational cost compared to Logistic Regression. In contrast, the Random Forest model exhibits the highest latency of 50 ms, suggesting that it requires more time for processing due to its ensemble nature and complexity.

5.6 ROC Analysis

Based on the proposed model, Figure 8 displays the Receiver Operating Characteristic (ROC) curve. The curve demonstrates the model's ability to distinguish between phishing and legitimate websites effectively. The Area ***Under*** the Curve (AUC) is observed to be close to 1, indicating excellent classification performance. This highlights the robustness of the proposed framework in minimizing both false positive and false negative rates across different decision thresholds.

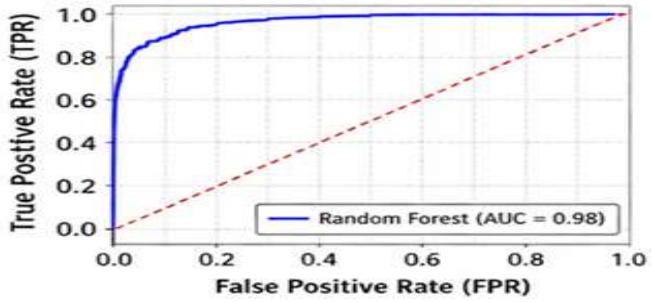

Figure 8: ROC Curve of Proposed Model

5.7 Discussion

The results indicate that the proposed machine learning-based model significantly improves phishing detection accuracy while maintaining acceptable computational efficiency. Compared to traditional methods, the proposed framework provides better generalization and robustness in detecting complex phishing patterns. The use of ensemble learning further enhances classification performance and reliability.

VI. CONCLUSION AND FUTURE WORK

The study presents an effective machine learning-based framework for phishing website detection that overcomes the limitations of traditional rule-based and blacklist methods by utilizing URL-based and domain-based features to accurately identify both known and zero-day attacks. The comparative evaluation of Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR) demonstrates that Random Forest achieves the best performance with the highest precision, recall, and F1-score, making it the most reliable model for phishing detection, while Logistic Regression offers faster prediction with lower accuracy and SVM provides balanced performance. The system also maintains an optimal trade-off between accuracy and computational efficiency, and the high ROC-AUC value confirms its robustness and strong classification capability in real-world scenarios. Despite these promising results, future work should focus on enhancing the system by integrating deep learning models for improved detection of complex phishing patterns, enabling real-time browser-based deployment, incorporating explainable AI techniques for better interpretability, improving scalability for large-scale environments, applying adversarial learning to strengthen robustness, and utilizing advanced feature selection methods along with extensive validation on diverse datasets to ensure better generalization and adaptability in evolving cybersecurity landscapes.

REFERENCES

1. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.
2. R. Mohammad, F. Thabtah, and L. McCluskey, “Phishing websites features,” UCI Machine Learning Repository, 2015.
3. S. Garera, N. Provos, M. Chew, and A. D. Rubin, “A framework for detection and measurement of phishing attacks,” in Proc. ACM Workshop on Recurring Malcode (WORM), 2007, pp. 1–8.
4. K. Jain and B. B. Gupta, “Phishing detection: Analysis of visual similarity based approaches,” Security and Communication Networks, vol. 2017, pp. 1–20, 2017.
5. M. Aburrous, M. A. Hossain, F. Thabtah, and K. Dahal, “Intelligent phishing detection system for e-banking using fuzzy data mining,” Expert Systems with Applications, vol. 37, no. 12, pp. 7913–7921, 2010.
6. C. Whittaker, B. Ryner, and M. Nazif, “Large-scale automatic classification of phishing pages,” in Proc. Network and Distributed System Security Symposium (NDSS), 2010.
7. J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond blacklists: Learning to detect malicious web sites from suspicious URLs,” in Proc. ACM SIGKDD, 2009, pp. 1245–1254.
8. Y. Fu, L. Wenyin, and X. Deng, “Detecting phishing web pages with visual similarity assessment based on Earth Mover’s Distance (EMD),” IEEE Trans. Dependable Secure Computing, vol. 3, no. 4, pp. 301–311, 2006.
9. G. Xiang, J. Hong, C. Rose, and L. Cranor, “CANTINA+: A feature-rich machine learning framework for detecting phishing web sites,” ACM Trans. Information and System Security, vol. 14, no. 2, pp. 1–28, 2011.
10. K. L. Chiew, K. S. Yong, and C. L. Tan, “A survey of phishing attacks: Their types, vectors and technical approaches,” Expert Systems with Applications, vol. 106, pp. 1–20, 2018.
11. Basnet, S. Mukkamala, and A. Sung, “Detection of phishing attacks: A machine learning approach,” in Proc. International Conference on Soft Computing Applications, 2008.
12. D. D. Silva and A. H. R. Costa, “Phishing detection using machine learning techniques,” in Proc. IEEE International Conference on Cyber Security, 2019.
13. M. Khonji, Y. Iraqi, and A. Jones, “Phishing detection: A literature survey,” IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091–2121, 2013.
14. B. B. Gupta, N. A. G. Arachchilage, and K. E. Psannis, “Defending against phishing attacks: Taxonomy of methods, current issues and future directions,” Telecommunication Systems, vol. 67, no. 2, pp. 247–267, 2018.

15. F. Toolan and J. Carthy, “Feature selection for spam and phishing detection,” in Proc. eCrime Researchers Summit, 2010.
16. H. Zhang, G. Liu, T. W. Chow, and W. Liu, “Textual and visual content-based anti-phishing: A Bayesian approach,” IEEE Trans. Neural Networks, vol. 22, no. 10, pp. 1532–1546, 2011.
17. S. Marchal, J. Francois, R. State, and T. Engel, “PhishStorm: Detecting phishing with streaming analytics,” IEEE Trans. Network and Service Management, vol. 11, no. 4, pp. 458–471, 2014.
18. Verma and Hossain, “Semantic feature selection for phishing detection,” in Proc. IEEE Conference on Trust, Security and Privacy, 2019.
19. Kaggle, “Phishing Website Dataset,” [Online]. Available: https://www.kaggle.com
20. N. 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.
21. M. Zouina and A. Outtaj, “A novel lightweight URL-based phishing detection system using machine learning,” Procedia Computer Science, vol. 127, pp. 408–417, 2018.
22. Adebowale, K. Lwin, and E. Sanchez, “Intelligent web phishing detection using random forest classifier,” in Proc. IEEE Conference on Cyber Security, 2019.