Grenze International Journal of Engineering and Technology, June Issue

Malware URL Detection for Enhancing Web Quality

Rohan Kokane[1], Shivani Rothe[2], Gaurav Suryawanshi[3], Pranay Chahankar[4], Purva Thote[5] and Sanjay Pande[6]
[1-6]Department of Computer Technology, Yeshwantrao Chavan College of Engineering, Wanadongri, Nagpur, India
Email: rohan.kokane11@gmail.com, rotheshivani@gmail.com, gauravsuryawanshi660@gmail.com, pranaychahankar6682@gmail.com, purvathote@gmail.com, sanjaypande2001@gmail.com

***Abstract*—The proliferation of internet-based services has fundamentally transformed daily communications and transactions, yet this digital transformation has simultaneously created unprecedented opportunities for cybercrime through malicious URLs. Particularly concerning is the rising prevalence of fraudulent links disseminated through SMS and messaging platforms, which pose significant threats to user privacy, financial security, and data integrity. Despite the critical nature of this challenge, current detection mechanisms often lack real-time capability and user accessibility. This research proposes a novel hybrid framework that leverages advanced machine learning and deep learning architectures to detect and classify malicious URLs in real-time messaging environments. The proposed system integrates a Random Forest classifier with a Convolutional Neural Network (CNN) to analyze URL characteristics across multiple dimensions, including lexical features, host-based attributes, and content-based patterns. Our approach uniquely combines traditional machine learning's interpretability with deep learning's pattern recognition capabilities, achieving enhanced detection accuracy while maintaining computational efficiency. The framework is implemented through a user-centric mobile application that seamlessly integrates with existing messaging platforms, providing immediate threat assessment of incoming URLs. Preliminary results indicate promising detection rates across various attack vectors, including phishing, malware distribution, and domain spoofing. This research contributes to the cybersecurity domain by offering an adaptive, scalable solution for protecting users against evolving URL-based threats in their daily digital interactions.**

***Keywords*— Malicious URL detection, machine learning, deep learning, cybersecurity, mobile security, real-time threat detection, SMS security, neural networks.**

I. INTRODUCTION

The evolution of internet technology has revolutionized connectivity and convenience, with critical online services like e-commerce and banking facilitating the exchange of sensitive information. However, this increase in internet usage has also led to a surge in cyberattacks, with malicious URLs emerging as a significant threat. These URLs serve as gateways to harmful content, causing financial losses, data breaches, and system vulnerabilities. Malicious URL attacks take various forms, including phishing, malware, and clickjacking. Phishing, a dominant attack vector, uses social engineering to extract sensitive information, while malware disrupts systems or enables unauthorized access. The increasing sophistication of such attacks, marked by obfuscation and rapid evolution, has made detection increasingly challenging. A robust detection method must ensure real-time analysis to alert users before accessing harmful sites, identify new and previously unseen URLs, and maintain high accuracy with minimal false positives and negatives.

Grenze ID: 01.GIJET.11.2.32_30

However, challenges persist, including dependency on dataset quality, lack of feature prioritization, and underexplored feature evaluation processes, particularly in cases of missing or ambiguous data. To address and break these challenges, we've proposed a hybrid model for vicious URL detection. By combining predefined static feature classifications with advanced techniques, the model enhances URL authenticity identification. This research contributes to advancing web security by tackling key issues and improving the detection of evolving threats.

II. Literature Survey

A. S. Rafsanjani, N. B. Kamaruddin et al. [1] in malicious URL detected a critical area of research, with various approaches addressing this challenge. RF and SVM are the traditional machine learning methods which rely on lexical and host-based features for classification.
It uses 42 features across blacklist, lexical, host based and content-based categories. These approaches perform well on structured datasets but struggle to generalize for unseen or dynamically generated URLs. Real time detection remains a challenge due to intensity of analyzing numerous URLs simultaneously.
Aljabri, M., & Salah, K. et at.[2] categorized malicious URL as phishing, spam, malware and defacement URLs and it used different detection approaches like supervised, unsupervised and semi-supervised learning. The review highlights the potential of machine learning algorithms to perform traditional blacklist-based methods. The dependency on datasets and manually crafted features results in overfitting and it reduces the ability to detect new unseen malicious URLs.
Roopalakshmi, R., Shukla, A., Karthikeyan, J., & Banerjee, K. et at.[3] used hybrid supervised learning approach, and integrated support vector regression (SVR) with advanced tuning techniques to address challenge like overfitting and improving accuracy. This framework demonstrates strong accuracy detection but its scalability and efficiency in real time applications are not fully explored and also it relies on many small and less diverse datasets which may not capture the variability of real-world malicious URLs.
Patil, D. R., & Patil, J. B et at.[4] used feature-based methods emphasize extracting meaningful lexical and host-based features, such as URL length, keywords, domain age, and IP reputation. Despite improving interpretability, these methods lack mechanisms to prioritize the most critical features, leading to inconsistent results.
Yahya, F., Isaac W., Mahibol et at.[5] used machine learning based framework for detecting phishing websites uses diverse algorithms like support vector machine, Decision trees and Random forests. It focuses on URL to extract URL characteristics and content-based attributes. Since multiple machines leaning algorithms are used for testing but exploration of ensemble methods or deep learning approaches might enhance the accuracy. Real-time detection approaches aim to provide swift responses but often face challenges in scalability and accuracy, especially with evolving cyber threats.
In [6] Hong, J., Kim detected the harmful URL by lexical features and integration of banned domain data. The framework incorporates 19 dimensions of features, including URL length, string entropy etc. The reliance on blacklist data poses a limitation since phishing websites frequently use newly created domain that might not appear on these lists. This involves scalability issues since framework performance in real-time, large scale internet environments was not extensively tested.
Existing methods reveal several gaps, including reliance on specific datasets, inadequate feature prioritization, and difficulty handling unseen URLs. This study addresses these issues by proposing a hybrid model combining Random Forest and CNN techniques, leveraging predefined static feature classifications for enhanced accuracy and adaptability to evolving threats.

III. Proposed Methodology

The development of an effective malicious URL detection model requires a comprehensive approach that encompasses feature extraction, model architecture, and rigorous training and evaluation processes. This section details each of these components to provide a clear understanding of the methods employed to achieve high accuracy in predicting malicious URLs.

A. Feature Extraction

Feature extraction is a crucial step in preparing the data for machine learning models. For this project, we focused on extracting meaningful features from URLs that can effectively differentiate between benign and malicious URLs. The dataset used, "malicious_phish.csv," consists of URLs labeled into four categories: benign, phishing, defacement, and malware.

B. Lexical Features

URL Length: The length of the URL, as malicious URLs are often longer to obfuscate their intent.
Number of Dots: The number of dots in the URL, since malicious URLs tend to have more subdomains.
Presence of Hyphens: Hyphens are commonly used in malicious URLs to mimic legitimate domains.
Presence of IP Address: URLs containing IP addresses are often indicative of malicious activity.

URL Length=len(url) (1)

C. Host-based Features

Age of Domain: The age of the domain, as malicious sites tend to have shorter lifespans.
Alexa Rank: Popular websites are less likely to be malicious.
WHOIS Information: Information about the domain's registrant can provide clues about its legitimacy.
Age of Domain=Current Date-Domain Registration Date

D. Content-based Features

Presence of Suspicious Keywords: Keywords such as "login", "update", "secure" can indicate phishing attempts.
Length of Path and Query: The length of the path and query components of the URL.
Entropy of URL: Measures the randomness in the URL string, where higher entropy might indicate obfuscation.

$$Entropy = -\sum \quad p(x)p(x) \tag{2}$$

These features were then normalized and encoded as necessary to prepare them for input into the machine learning models.

B. Model Architecture

In this project, we combined a Random Forest (RF) model with a Convolutional Neural Network (CNN) to leverage the strengths of both traditional machine learning and deep learning approaches.

Random Forest (RF)

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. It is robust to overfitting and effective for high-dimensional data.

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} \quad f_i(x) \tag{3}$$

Convolutional Neural Network (CNN)

CNNs are particularly effective for image recognition but can also be applied to sequential data such as URLs. The CNN model in this project consists of:

- Input Layer: Takes the preprocessed URL data.
- Embedding Layer: Converts categorical variables into dense vectors.
- Convolutional Layers: Extracts local patterns from the URL sequences.
- Pooling Layers: Reduces the dimensionality while retaining the most important features.
- Fully Connected Layers: Integrates features for final classification.

$$y = f(W \cdot x + b) \tag{4}$$

Combined Model

The combined model uses the Random Forest model to generate initial predictions, which are then fed into the CNN as additional input features. This hybrid approach allows the model to capture both global patterns (through RF) and local patterns (through CNN).

$$P_{final} = \alpha \cdot P_{RF} + \beta \cdot P_{CNN} \tag{4}$$

Where P_{final} is the final prediction probability, and α and β are weights assigned to the RF and CNN predictions, respectively.

C. Training and Evaluation

Training the model involves splitting the dataset into training and testing sets, optimizing hyperparameters, and evaluating performance using various metrics.

Data Splitting
The dataset was split into 80% training and 20% testing. Stratified sampling was used to ensure that each class was proportionally represented in both sets.

$$Training\ Set\ Size = 0.8 \times Total\ Data$$

$$Testing\ Set\ Size = 0.2 \times Total\ Data$$

Model Training
- Random Forest: The number of trees, maximum depth, and minimum samples per leaf were optimized using grid search.
- CNN: The architecture was optimized by adjusting the number of layers, filter sizes, and dropout rates.

$$Loss\ Function = -\sum \quad y\ log\ log\ (\hat{y}) \tag{5}$$

$$Optimizer = Adam(learning\ rate) \tag{6}$$

Evaluation Metrics
Accuracy: Measures the proportion of correctly classified instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

Precision, Recall, and F1-Score: Evaluate the model's ability to correctly identify each class.

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{9}$$

Confusion Matrix: Provides a detailed breakdown of true positives, true negatives, false positives, and false negatives for each class.

$$Confusion\ Matrix = [TP_0\ FP_0\ TP_1\ FP_1\ \ FN_0\ TN_0\ FN_1\ TN_1\ \ TP_2\ FP_2\ TP_3\ FP_3\ \ FN_2\ TN_2\ FN_3\ TN_3\ ] \tag{10}$$

Below is the flowchart of working of model:

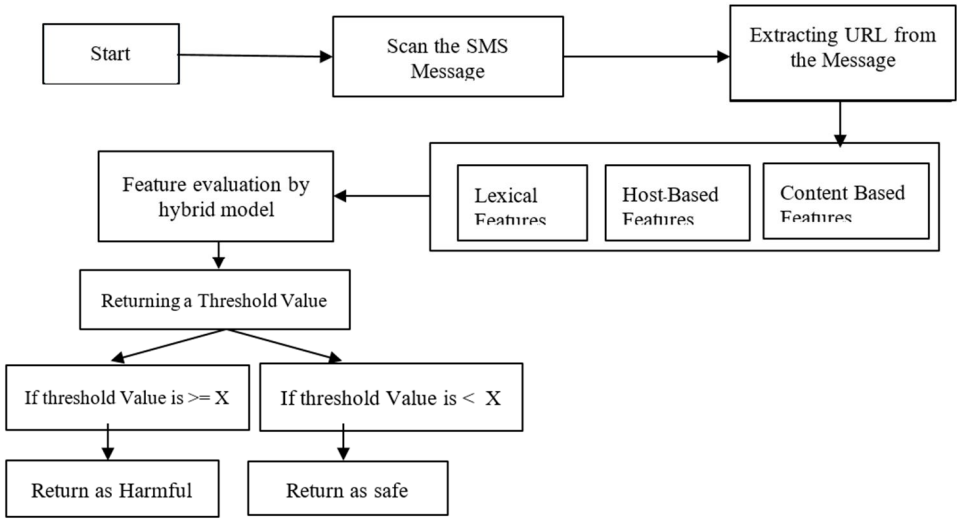


Figure 1. Flowchart of working model

IV. Result and Discussion

In this section, we present the results of our combined Random Forest (RF) and Convolutional Neural Network (CNN) model for detecting malicious URLs. The performance metrics are discussed in detail, and comparisons with existing models are provided to illustrate the efficacy of our approach.

A. Performance Metrics

The performance of the hybrid RF-CNN model was evaluated using a comprehensive set of metrics, including accuracy, precision, recall, F1 score, and Receiver Operating Characteristic (ROC) curves. These metrics provide a holistic view of the model's capability to correctly identify different types of malicious URLs, such as phishing, defacement, malware, and benign URLs.

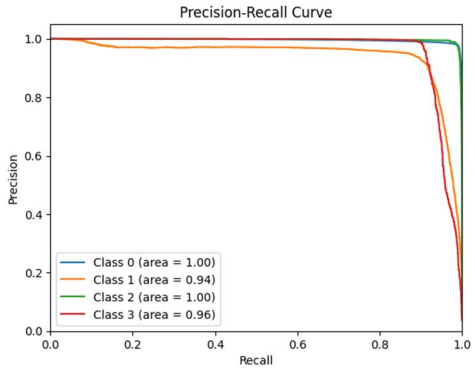
Accuracy is a fundamental metric that measures the proportion of correctly classified instances out of the total instances. Our model achieved an accuracy of 97.06%, indicating a high level of overall correctness in its predictions.

Precision measures the proportion of true positive predictions among all positive predictions, providing insight into the model's ability to avoid false positives. The precision for each class is shown below:

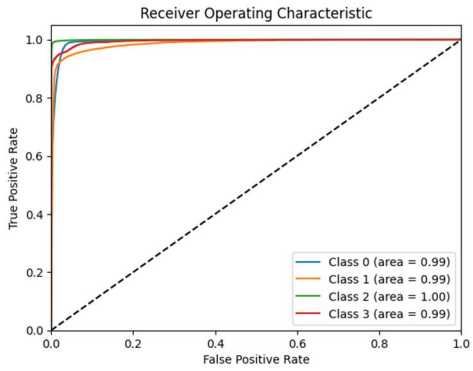
Recall (also known as sensitivity) measures the proportion of true positive predictions among all actual positives, indicating the model's ability to capture all relevant instances. The recall for each class is as follows:

Table I

Class	Recall
Benign	0.99
Phishing	0.88
Defacement	0.98
Malware	0.91



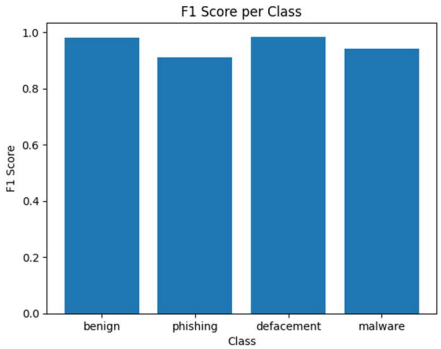
(a)

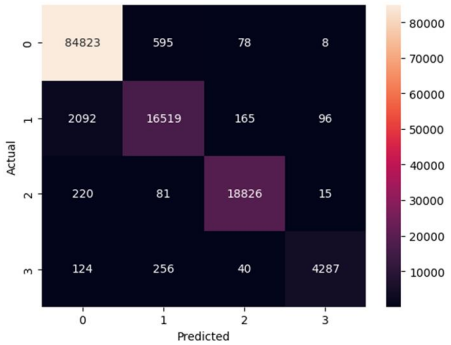


(b)

Figure 1(a): Precision recall and Figure 1 (b) : ROC Curves for Each Class with Corresponding AUC Values

F1 Score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. T





(a) (b)

Figure 2 (a): F1 score of each classes, and Figure (b): Confusion matrix

A confusion matrix provides a detailed breakdown of the model's performance across all classes, highlighting the true positives, false positives, false negatives, and true negatives for each class. Figure 1 illustrates the confusion matrix for our model.

ROC Curve and AUC: The ROC curve plots the true positive rate against the false positive rate for each class, while the Area Under the Curve (AUC) provides a single measure of the model's discriminative ability. Higher AUC values indicate better performance. Figure 2 shows the ROC curves for all classes.

V. CONCLUSION

This study introduced a hybrid model that combines Random Forest (RF) and Convolutional Neural Network (CNN) techniques to detect malicious URLs. The model achieved an accuracy of 97.06%, demonstrating its capability to effectively classify benign, phishing, defacement, and malware URLs. The high precision and F1 scores for benign, defacement, and malware categories emphasize the model's reliability in identifying threats with minimal errors. However, the recall for phishing URLs (0.88) highlights a limitation, suggesting a need for improvements in addressing the deceptive nature of phishing attempts. The hybrid RF-CNN model holds significant promise for practical use in tasks such as URL analysis and threat detection. With further enhancements, it can offer robust real-time protection against emerging cyber threats.

To overcome existing limitations, future work will aim to enhance the recall for phishing URLs by incorporating additional data sources, including real-time network traffic and metadata. This broader approach can help capture more diverse phishing patterns. Ensuring the dataset is regularly updated is also essential for the model to stay adaptive to new types of attacks. Collaborating with cybersecurity organizations to integrate real-time threat intelligence will further strengthen the model’s reliability. Additionally, exploring advanced ensemble methods like stacking or blending with other deep learning architectures could improve performance, especially for complex URL categories like phishing. Integrating explainable AI techniques could also enhance transparency, providing better insights into the model's decisions and improving its ability to address false positives and negatives.

REFERENCES

[1] A. S. Rafsanjani, N. B. Kamaruddin, M. Behjati, S. Aslam, A. Sarfaraz, and A. Amphawan, "Enhancing Malicious URL Detection: A Novel Framework Leveraging Priority Coefficient and Feature Evaluation," IEEE Access, vol. 12, pp. 85001-85026, 2024. doi: 10.1109/ACCESS.2024.3412331.

[2] Aljabri, M., & Salah, K. (2022). Detecting malicious URLs using machine learning techniques: Review and research directions. Journal of Network and Computer Applications, 10, 121396. https://doi.org/10.1016/j.jnca.2022.121396

[3] Roopalakshmi, R., Shukla, A., Karthikeyan, J., & Banerjee, K. (2024). Robust framework for malevolent URL detection using hybrid supervised learning. Procedia Computer Science, 230, 241-247. https://doi.org/10.1016/j.procs.2023.12.079

[4] Patil, D. R., & Patil, J. B. (2018). Malicious URLs detection using decision tree classifers and majority voting technique. Cybernetics and Information Technologies, 18(1), 11–29.

[5] Yahya, F., Isaac W., Mahibol, R., Kim Ying, C., Bin Anai, M., Frankie, A., Sidney, Ling Nin Wei, E., & Guntur Utomo, R. (2021). Detection of phising websites using machine learning Wireless Networks approaches. In 2021 International conference on data science and its applications (ICoDSA).

[6] Hong, J., Kim, T., Liu, J., Park, N., Kim, SW. (2020). Phishing URL Detection with Lexical Features and Blacklisted Domains. In: Jajodia, S., Cybenko, G., Subrahmanian, V., Swarup, V., Wang, C., Wellman, M. (eds) Adaptive Autonomous Secure Cyber Systems. Springer, Cham. https://doi.org/10.1007/978-3-030-33432-1_12