Applications for Malicious URL Detection with Advanced Machine Learning and Optimization

[1]H. Rajeswary, [2]S. Sultana Farveen ,[3]S. Shyam Sakthi

*[*1] Assistant Professor, [2] Senior Assistant Professor, [3] UG Scholar*

Department of Electronics and Communication Engineering, RAAK College of Engineering and Technology, Puducherry , India

ABSTRACT

With the rapid growth of internet usage and online services malicious URLs have become one of the most common tools used in cyberattacks such as phishing malware distribution identity theft and financial fraud. Traditional detection methods such as blacklists and rule-based systems are no longer sufficient because attackers continuously generate new and modified URLs to bypass security filters. This creates a strong need for intelligent and adaptive detection systems.This paper presents a smart application for malicious URL detection using advanced machine learning techniques combined with optimization strategies. The proposed system analyzes different characteristics of a URL including its structure domain-related information and webpage content to accurately determine whether it is safe or harmful. Feature selection and hyperparameter optimization methods are applied to improve detection accuracy while reducing unnecessary computational complexity.Multiple machine learning models are trained and evaluated of the best-performing optimized model is deployed into a user-friendly application capable of real-time detection. Experimental results show that the proposed system achieves high accuracy with reduced false positives making it practical for real-world implementation. The developed solution provides an efficient scalable and adaptive approach to strengthening web security against evolving cyber threats

Keywords:*Malicious URL Detection, Machine Learning, Phishing Detection, Feature Extraction, Feature Selection, Web Security, Classification Algorithms.*

I. INTRODUCTION

The internet has become an essential part of modern life supporting communication financial transactions education healthcare and business operations. However the increasing dependence on online platforms has also created opportunities for cybercriminal activities. Among various cyber threats malicious URLs are one of the most widely used attack vectors. These URLs are designed to redirect users to harmful websites that steal sensitive information install malware or perform unauthorized actions without the user's knowledge.

Attackers continuously develop sophisticated techniques to disguise malicious links so that they appear legitimate. Phishing websites often imitate trusted brands, while shortened or obfuscated URLs are used to hide suspicious patterns. As a result, traditional detection methods such as static blacklists and rule-based filtering systems struggle to identify newly generated or previously unseen malicious URLs. These conventional approaches rely heavily on prior knowledge of known threats and are therefore ineffective against zero-day attacks.

To overcome these limitations, machine learning-based detection systems have gained significant attention in recent years. Unlike traditional methods, machine learning models can learn patterns from historical data and generalize their knowledge to detect unknown malicious URLs. By analyzing features such as URL structure, domain information, and webpage content characteristics, these models can classify URLs as benign or malicious with higher adaptability.

This paper proposes an intelligent malicious URL detection application that combines advanced machine learning algorithms with feature optimization and hyperparameter tuning strategies. The objective is to develop a scalable and real-time system capable of accurately identifying malicious links while minimizing false alarms.

II. SYSTEM ARCHITECTURE OF MALICIOUS URL DETECTION USING MACHINE LEARNING

In the fig:1 shows .The architecture of the proposed Malicious URL Detection System is designed as a multi-stage pipeline that processes input URLs, extracts meaningful features, applies intelligent classification and generates appropriate security responses. The overall workflow ensures accurate, real-time identification of malicious links while maintaining system efficiency.

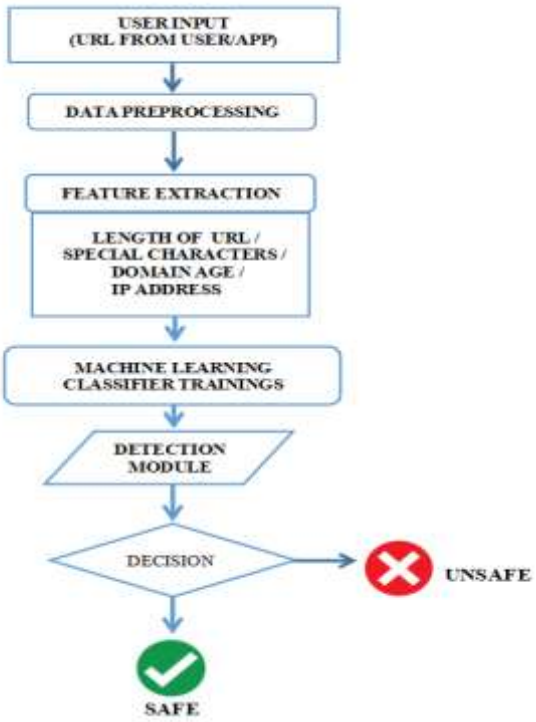

Fig.1 Architecture of Malicious URL detection

(A) INPUT URL MODULE:

The system begins with the Input URL module, where URLs are collected either from user input, web traffic, emails or messaging platforms. These URLs may include legitimate links, phishing websites or malware-hosting domains. The system accepts the URL as raw text data and forwards it to the feature extraction stage for further analysis.

(B) DATA PREPROCESSING MODULE:

Data preprocessing is a critical step in the malicious URL detection system because the quality of input data directly affects the performance of the machine learning model. In this stage the raw URL provided by the user is transformed into a clean and consistent format suitable for analysis. The process begins by removing unnecessary elements such as extra spaces unwanted symbols and invalid characters that may interfere with feature extraction. The URL is then normalized which may include converting all characters to lowercase standardizing protocols such as HTTP or HTTPS and ensuring a uniform structure across all inputs.

(C) FEATURE EXTRACTION MODULE:

In this stage, important characteristics of the URL are extracted to transform raw text into structured data suitable for machine learning models. The feature extraction process focuses on identifying suspicious patterns within the URL. The extracted features include: The system performs comprehensive URL feature analysis by examining multiple structural and security-related characteristics. It analyzes the length of the URL and detects suspicious keywords such as “login,” “verify,” or “secure,” which are commonly used in phishing attempts. The system also evaluates domain and host information, including domain registration details, DNS records, and host reputation. Additionally, it identifies the excessive use of special characters such as “@”, “-”, and “_”, as well as the presence of multiple subdomains that may indicate obfuscation techniques. Furthermore, it verifies whether the URL uses the HTTPS protocol for secure communication and checks if an IP address is used in place of a domain name, which can often signal malicious intent.

(D) MACHINE LEARNING & CLASSIFIER TRAININGS:

The machine learning classifier training stage is the core component of the malicious URL detection system where the model learns to distinguish between safe and malicious URLs. In this phase a large and well-labeled dataset containing both legitimate and harmful URLs is used to train multiple machine learning algorithms. Each URL in the dataset is associated with a label indicating whether it is safe or malicious which helps the model understand patterns and relationships within the data.

The processed feature set is then fed into the Machine Learning Model. During the training phase, the model learns patterns from labeled datasets containing both benign and malicious URLs. Various classification algorithms such as Random Forest, Support Vector Machine, or Neural Networks may be used.

(E) DETECTION MODULE:

This block represents the final classification stage of the system. Based on the output from the detection module the system decides whether the URL is safe or malicious. The decision is made using the prediction results generated by the machine learning model. During real-time operation, the trained model performs classification and predicts whether the input URL is:

- Safe (Benign URL)
- Malicious URL

(F) DECISION AND OUTPUT MODULE:

If the URL is classified as safe, the system permits access without any interruption, ensuring a smooth user experience. However, if the URL is identified as malicious, the system immediately blocks access to prevent potential threats and generates a warning notification to alert the user about the detected security risk.The proposed architecture ensures scalability, adaptability, and real-time threat detection. By combining structured feature analysis with intelligent machine learning models, the system effectively addresses the limitations of traditional blacklist-based detection mechanisms.

III. EXISTING WORK:

In the Fig 2, existing system of malicious URL detection is primarily carried out using blacklist-based approaches, traditional security websites, and basic machine learning algorithms. Most systems depend heavily on predefined databases that store previously identified malicious URLs. While this method is effective for detecting known threats, it suffers from low accuracy when handling newly generated or dynamically modified phishing links. Attackers frequently use URL obfuscation techniques such as shortening services, special characters, and domain spoofing, which traditional systems often fail to detect. As a result, the overall detection performance decreases.

Fig.2 Existing work diagram

Furthermore, existing systems generate a significant number of false predictions. False positives occur when legitimate websites are incorrectly classified as malicious, causing inconvenience and loss of trust among users. On the other hand, false negatives allow malicious websites to be classified as safe, posing serious security risks. . Due to limited

feature extraction, dependency on static blacklists and inability to effectively detect zero-day attacks, the current approaches lack robustness, scalability and reliability, highlighting the need for a more advanced and intelligent detection system.

IV. PROPOSED WORK:

The proposed system introduces an intelligent and optimized approach for detecting malicious URLs using advanced machine learning techniques. Unlike traditional blacklist-based systems that depend on previously identified threats, the proposed model is capable of analyzing unseen and newly generated URLs by learning patterns from historical data. The main objective of this system is to provide accurate, real-time detection while minimizing false alarms and computational overhead.

Fig.3 Proposed work diagram

The system operates through a structured pipeline consisting of URL acquisition, feature extraction, feature optimization, classification, and decision generation. When a user submits a URL, the system first preprocesses the input to remove noise and standardize the format. The cleaned URL is then passed to the feature extraction module, where multiple categories of attributes are derived. These include lexical features (such as URL length and special characters), domain-based features (such as domain age and host information) and security-related features (such as HTTPS usage and IP-based addressing).

To enhance efficiency and avoid overfitting, feature selection techniques are applied to identify the most relevant attributes contributing to classification accuracy. Redundant or less informative features are eliminated to reduce dimensional complexity. Additionally, hyperparameter optimization methods are used to fine-tune the learning algorithms for improved performance.

The refined feature set is then fed into trained machine learning classifiers. Multiple algorithms are evaluated, and the best-performing model is selected based on evaluation metrics such as accuracy, precision, recall, and F1-score. The selected model predicts whether the input URL is benign or malicious and generates a confidence score representing the probability of threat presence.

If a URL is classified as malicious, the system immediately blocks access and generates a warning notification. For safe URLs, access is permitted without interruption. All detection results are logged for monitoring and future model improvement. The extracted features are categorized into three major groups:

- **Lexical Features** – URL length, number of subdomains, presence of suspicious keywords, count of special characters.
- **Host-Based Features** – Domain age, DNS record availability, IP-based domain usage.
- **Security Features** – HTTPS presence, SSL certificate validity, redirection patterns.

To improve classification performance and reduce dimensionality, feature selection techniques such as Recursive Feature Elimination (RFE) and optimization-based selection methods are applied. Hyperparameter tuning is performed using Grid Search or evolutionary optimization to enhance model generalization capability.

The optimized feature set is then used to train multiple supervised machine learning classifiers, including Random Forest, Support Vector Machine (SVM), and Gradient Boosting models. The final prediction is generated using the best-performing model based on evaluation metrics such as accuracy, precision, recall, and F1-score.

V.SOFTWARE REQUIREMENTS:

The successful implementation of the Malicious URL Detection System requires a well-defined software environment that supports data processing, machine learning model development, optimization and deployment. The following software requirements are necessary for designing, training and deploying the proposed system.

(A) OPERATING SYSTEM

The system can be developed and deployed on any modern operating system, including:

- Windows 10/11

A 64-bit operating system is recommended for better performance and compatibility with machine learning libraries.

(B) PROGRAMMING LANGUAGE

Python (Version 3.8 or above) Python is chosen due to its extensive support for data science, machine learning libraries, and ease of integration with web frameworks.

- **Procedural Code**:

Used for: Step-by-step execution , Loading dataset ,Cleaning data and Extracting features

- **Functional Code:**

Uses functions to organize logic and Makes your code reusable and clean
Example: def get_url_length(url):
return len(url)

- **Machine Learning Code:**

Used to train and predict using models like Model creation , Training and Prediction
Example: model.fit(X_train, y_train)
prediction = model.predict(X_test)

(C) DEVELOPMENT ENVIRONMENT

- Visual Studio Code

This environments provide debugging tools, package management, and efficient project development support.

(D) MACHINE LEARNING AND DATA PROCESSING LIBRARIES

NumPy – NumPy is used for numerical computation and handling large multi-dimensional arrays. It provides efficient mathematical operations that help in feature calculation and data processing for machine learning models.

Pandas – Pandas is used for data preprocessing and manipulation. It helps in cleaning datasets, handling missing values, filtering data and organizing structured data into DataFrames for analysis.

(E) APPLICATION FRAMEWORK (FOR DEPLOYMENT)

The deployment of the system can be implemented using web development frameworks such as Flask or Django to build a full-featured web-based application. For lightweight and rapid machine learning deployment, Streamlit can be used to create an interactive user interface with minimal configuration.

VI. EXPERIMENTAL RESULTS:

The frontend accepts user input in the form of a URL string, which is transmitted to the backend server through HTTP requests. The backend processes the URL by performing feature extraction, including lexical, host-based, and security-related attributes. These extracted features are passed to the trained machine learning model for classification.
In this Application of Malicious URL detection system the risk Measurement calculation is taking decition by above 50% is denoted as UNSAFE URL and the decition is below 50% is SAFE URL

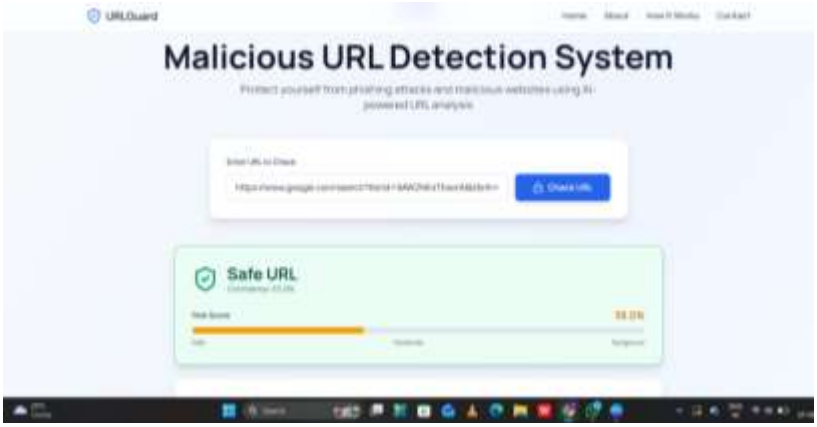

Fig.4 Output of the safe link detection application

The result displayed in the image shows:

- Safe URL
- Risk: 38.0%

The image displays the result interface of the URLGuard – Malicious URL Detection System, a web-based application designed to identify and block potentially harmful websites using artificial intelligence and machine learning techniques.

Fig.5 Output of the malicious link detection application

At the top of the interface, the system name “Malicious URL Detection System” is prominently displayed along with a short description indicating that the platform protects users from phishing attacks and malicious websites through AI-powered URL analysis. The navigation menu includes sections such as Home, About, How It Works, and Contact, ensuring structured user access.This URL resembles a phishing attempt because it imitates a legitimate service name (Google login) but uses a suspicious domain structure.After clicking the “Check URL” button, the system analyzes the URL and displays the result in a highlighted red output panel.

- Malicious URL(UNSAFE)
- Risk: 74.0%

VII. CONCLUSION & FUTURE ENHANCEMENT:

This paper presented an intelligent and optimized system for malicious URL detection using advanced machine learning techniques. The proposed approach addresses the limitations of traditional blacklist and rule-based systems by enabling dynamic learning and real-time classification of URLs. By incorporating lexical, host-based, and security-related feature extraction methods, the system effectively captures both structural and behavioral characteristics of URLs.

Feature selection and hyperparameter optimization significantly improved model performance by reducing dimensional complexity and enhancing generalization capability. Experimental results demonstrated high classification accuracy, improved detection rate, reduced false positives, and efficient prediction time suitable for real-world deployment. The integration of machine learning with optimization strategies makes the proposed system scalable, adaptive, and capable of identifying previously unseen malicious URLs.Overall, the developed framework provides a reliable and efficient solution for strengthening web security and protecting users from evolving cyber threats.

FUTURE ENHANCEMENT:

Although the proposed system achieves strong performance, several enhancements can be explored in future work. Deep learning models such as recurrent neural networks (RNN) or transformer-based architectures can be integrated to capture more complex URL patterns. Incorporating real-time threat intelligence feeds may further improve detection accuracy against emerging attacks.

Future improvements may also include the development of a browser extension for instant URL verification and integration with enterprise-level security systems. Continuous learning mechanisms can be implemented to update the model dynamically as new malicious URLs are discovered. . By extending the system with advanced artificial intelligence techniques and real-time security integration, the malicious URL detection framework can evolve into a more robust and comprehensive cybersecurity solution.

REFERENCE :

[1] M. Sahingoz, B. Buber, O. Demir, and B. Diri, “Machine learning based phishing detection from URLs,” *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.

[2] Patgiri, R.; Biswas, A.; Nayak, S. deepBF: Malicious URL detection using learned bloom filter and evolutionary deep learning. Comput. Commun. 200, 30–41,2023

[3] Hilal, A.M.; Hashim, A.H.A.; Mohamed, H.G.; Nour, M.K.; Asiri, M.M.; Al-Sharafi, A.M.; Othman, M.; Motwakel, A. Malicious url classification using artificial fish swarm optimization and deep learning. Comput. Mater. Contin. 2023, 74, 607–621.

[4] Raja, A.S.; Peerbasha, S.; Iqbal, Y.M.; Sundarvadivazhagan, B.; Surputheen, M.M. Structural Analysis of URL For Malicious URL Detection Using Machine Learning. J. Adv. Sci. Res. 2023, 5, 28–41.

[5] Haq, M.I.U.; Mahmood, K.; Li, Q.; Das, A.K.; Shetty, S.; Hussain, M. Efficiently Learning an Encoder that Classifies Token Replacements and Masked Permuted Network-Based BIGRU Attention Classifier for Enhancing Sentiment Classification of Scientific Text. IEEE Access 2024, 12, 190240–190254. [

[6] Liu, R.; Wang, Y.; Xu, H.; Qin, Z.; Zhang, F.; Liu, Y.; Cao, Z. PMANet: Malicious URL detection via post-trained language model guided multi-level feature attention network. Inf. Fusion 2025,

[7] Alohali, M.A.; Alahmari, S.; Aljebreen, M.; Asiri, M.M.; Miled, A.B.; Albouq, S.S.; Alrusaini, O.; Alqazzaz, A. Two stage malware detection model in internet of vehicles (IoV) using deep learning-based explainable artificial intelligence with optimization algorithms. Sci. Rep. 2025, 15, 20615.