Phishing Website Detection Using ML

Dr. Sunil Wankhade[1], Hritik Kanse[2], Aditya Mohile[3], Sanika Padme[4], Riya Sawant
Dept. of Information Technology, MCT's Rajiv Gandhi institute of technology

***Abstract*—Phishing websites are a significant security threat. Numerous cyberattacks jeopardize the confidentiality, integrity, and availability of both company and consumer data, with phishing often being the initial step in these attacks. Optical Character Recognition (OCR) has emerged as a viable solution for real-time detection of phishing links on mobile platforms.[1] Researchers have dedicated decades to developing innovative methods for the automatic detection of phishing sites. Although advanced solutions can yield improved outcomes, they often require extensive manual feature engineering and struggle to identify new phishing tactics. Consequently, there remains a pressing need for strategies that can automatically detect phishing websites and swiftly address zero- day phishing attempts. The webpage linked in the URL contains a wealth of information that can help assess the maliciousness of the web server. Machine Learning has proven to be an effective approach for phishing detection, overcoming the limitations of previous methods. We performed a comprehensive literature review and proposed a novel technique for identifying phishing websites through feature extraction and a machine learning algorithm. This research aims to utilize the collected dataset to train machine learning models and deep neural networks to predict phishing websites**.

***Index Terms*—Phishing Website, Fake Website, Spam, Hacker, Verification, Action, Document, Urgent, Message, Password**

I. INTRODUCTION

In today's world, phishing has emerged as a significant issue for security researchers due to the ease of creating fake websites that mimic legitimate ones. While experts can often spot these fraudulent sites, many users struggle to recognize them, leaving some individuals vulnerable to phishing attacks. The main objective of these attackers is to steal bank account credentials. Hackers usually operate by sending emails often claiming that your university network password will expire in 24 hours and urging you to update it. If you click on link provided, you will be taken to a page hosted on a hackers server, where they can capture all of your online information. Phishing is one of the most dangerous criminal activities in in cyberspace. People utilize the internet to access banking and governmental services, and phishing assaults have significantly increased in recent years. Phishers have turned this into profitable business. They employ various tactics to target vulnerable users, including messaging, VOIP, spoofed links, and fake websites. Creating counterfeit websites is relatively easy, and these sites often mimic genuine ones in both

Layout and content. The primary goal of these sites is to collect sensitive information from users such as account numbers, login IDs, and passwords for debit and credit cards. Additionally, attackers may pose as security personnel and ask users to answer security questions under the guise of enhancing security measures. When users respond they can easily fall victim to phishing attacks. Preventing these attacks involves identifying fraudulent websites and raising user awareness about how to spot them. Machine learning algorithms have emerged as a powerful tool for detecting phishing sites. This study explores various methods for identifying phishing websites. Machine learning strategy has proven to be more effective than other methods.

II. LITERATURE SURVEY

A comprehensive review of existing literature reveals a myriad of approaches and techniques employed in the detection and mitigation of spam messages across various online platforms. Lightweight URL-based detection techniques have demonstrated the potential to significantly reduce resource overhead while maintaining accuracy [2]. Studies in this domain have explored content-based analysis, machine learning algorithms, and pattern recognition techniques to identify spam messages effectively. Furthermore, research has emphasized the importance of feature engineering, data preprocessing, and model evaluation in the development of robust spam detection systems. By synthesizing insights from prior research, this

study seeks to build upon existing knowledge and contribute to the advancement of spam detection methodologies tailored specifically for social media environments.

III. PROPOSED SYSTEM

The proposed Phishing Website Detection is a crucial aspect of cybersecurity, and an effective approach usually combines machine learning, feature extraction, and heuristic methods. Here a high-level methodology for identifying phishing websites while ensuring originality:

1. Problem Definition: Clearly outline the challenge of detecting phishing websites. Focus on distinguishing between legitimate and phishing websites based on various attributes (URLs, website content, certificates, etc.).
2. Data Collection: Phishing Dataset- Gather data from reliable sources like PhishTank, OpenPhish or similar platforms. Legitimate Dataset: Obtain Legitimate website data from reputable sources, such as Alexa Top Sites or Web Crawlers.
3. Feature Engineering: Identify key features from websites to tell phishing sites apart from legitimate ones. Common feature categories include: URL- based Features: Length of the URL, presence of suspicious characters (eg.@,-,_), use of shortened URLs, etc. Presence of dubious TLDs (top-level domains). Domain - Based Features: Domain age, WHOIS information, DNS record, etc. Utilize public databases to check for blacklisted domains. Content-Based Features: HTML and Text analysis, Keyword analysis, presence of forms requesting sensitive information. Detection of mismatched logos, unusual JavaScript behavior, external links, and content similarity. SSL/TLS Certificate Features: Certificate validity, certificate authorities, and whether the site employs HTTPs. Behavioral Features: Observe how the website responds to user interactions. Does it generate multiple pop ups? Redirect users? Initiate suspicious downloads?
4. Preprocessing: Address missing values (eg, websites lacking SSL certificates or invalid WHOIS data) Normalise or standardize numerical features. Encode categorical feature (like domain registrar or certificate issuer) if employing machine learning algorithms.

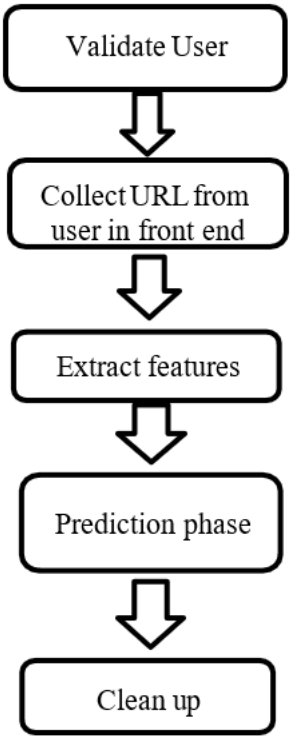

Fig.: Flow Diagram for Website Detection.

Module-wise description is given as follows:

1. Validate Users: The phishing website detection system is designed to help users verify whether a website is legitimate or a phishing attempt. The user engages with the system to determine whether a given website is phishing or legitimate.

2. Collect URL from user in front end: Once logged in, the user is prompted to enter the URL of the website they want to check. The system ensures the entered URL is in a valid format before proceeding.

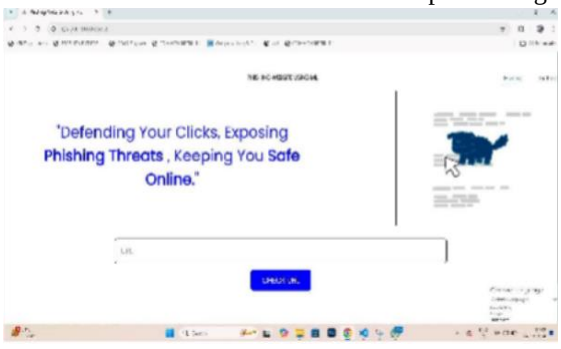

Fig: URL Uploading Page

3. Feature Extraction: The system analyses the submitted URL to extract various features that help differentiate phishing websites from legitimate ones.

These features may include: URL-Based Features: Length of the URL, presence of special characters. Domain-Based Features: Domain age, WHOIS data, use of HTTPs.

Network-Based-Features: IP address reputation, DNS record, SSL Certificate validity.

Fig: Result for secured Website Detection.

4. Prediction Phase: The Random Forest Model evaluates the extracted features and determines the likelihood of the website being a phishing site. The model returns one of two possible outcomes:

- Prediction output 1: Website is secure.
- Prediction output 2: Website is not secure.

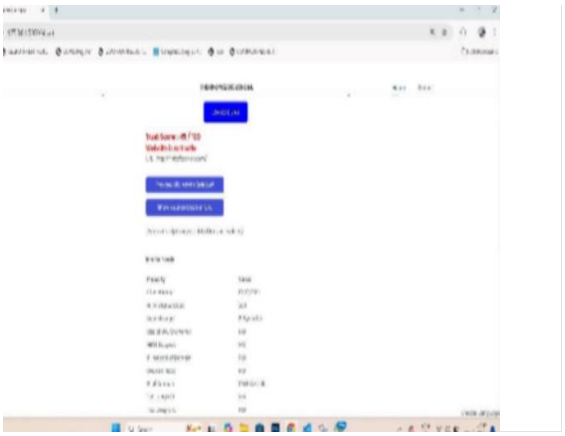

Fig: Result for not secured website detection

Clean up: The User can analyze the result and decide whether to avoid the website or proceed. They may choose to enter a not her URL for log out of the system.

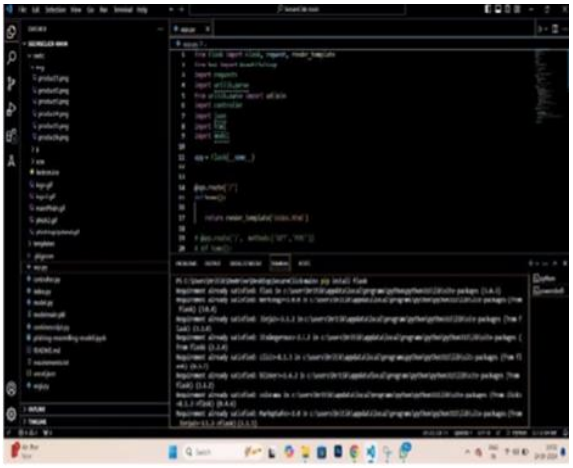

Fig: Flask Installation

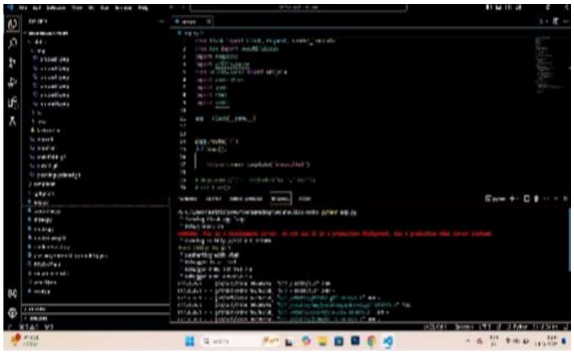

Fig: Code Run Command

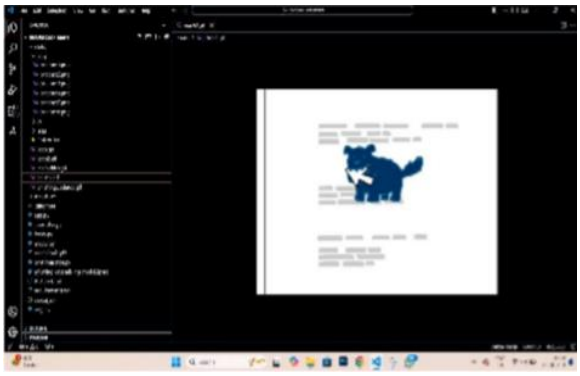

Fig: GIF 1

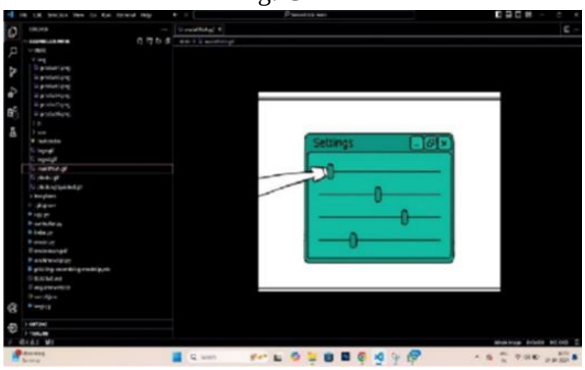

Fig: GIF 2

IV. PHISHING WEBSITE DETECTION USING MACHINE LEARNING:

In today's fast-paced digital world, the rising threat of phishing attacks calls for new and effective ways to detect and prevent them. AI meta-learners like Extra-Trees algorithms have shown remarkable precision in detecting phishing attempts by analyzing patterns in website attributes.[3] This paper introduces a fresh approach that leverages the power of Artificial Intelligence (AI) and Machine Learning (ML) to tackle phishing attempts head-on. By using a combination of techniques, this research employs algorithms like XGBoost, LightGBM, Naïve Bayes, and CatBoost, along with a Graph Neural Network (GNN), to carefully analyze URL structures, content patterns, and user behavior. It zeroes in on important features such as URL length, the presence of dots, slashes, numbers, and special characters to thoroughly train the model. With real-time monitoring, the system can continuously adapt to new phishing tactics, boosting its ability to protect users and organizations from the constantly changing landscape of cyber threats. This research offers a detailed look at various machine learning methods aimed at strengthening online security against the widespread issue of phishing.

One of the salient features of this research is the use of Graph Neural Networks (GNNs) that allow examining complex relationships between domains, URLs, and network topology. Contrasting from conventional machine learning models that depend on numeric or categorical features, GNNs allow for a structural perception of how phishing sites are interconnected, thereby gaining better insights into link topology and domain dynamics. Machine learning models embedded within cognitive security architectures are increasingly being used to counteract phishing attacks effectively.[4] This approach improves the system's capacity to identify never-before-seen phishing sites using their connectivity and interaction patterns instead of static indicators.

In addition, the study focuses on real-time monitoring mechanisms that provide constant adaptation to new phishing strategies. SPWalk model leverages property-oriented feature learning to enhance the detection capabilities of phishing classifiers.[5] Most conventional phishing detection methods have difficulties keeping pace with the dynamic evolution of cyber-attacks. To counter this, the system dynamically adjusts its feature set, using continuous learning methods to improve model accuracy with time. Through the use of real-time threat intelligence and the constant retraining of the model using newly discovered phishing examples, the system is resistant to adversarial attacks and zero-day phishing threats.

The efficacy of this method is established through large-scale experimentation on benchmark phishing datasets, such as those from PhishTank, OpenPhish, and the UCI Machine Learning Repository. Performance metrics like accuracy, precision, recall, F1-score, and ROC-AUC are employed to measure the strength of the proposed models. he integration of MongoDB, Express, Angular, and Node.js within the MEAN stack offers a streamlined approach to building responsive and scalable inventory management solutions.[8]

This study makes a substantial contribution to the development of phishing detection methods through the integration of various AI-based approaches to build an intelligent, adaptive phishing detection system. The results highlight the significance of using feature-based learning in conjunction with graph-based analysis to enhance detection accuracy and adaptability. Businesses are increasingly leveraging ChatGPT for strategic planning and enhanced decision-making support through AI-driven insights.[7] Moreover, the approach offers a scalable solution for organizations, providing real-time protection against dynamic cyber threats while minimizing dependency on legacy rule-based detection systems.

With the strength of machine learning, ensemble models, and graph-based methods, this research provides the building blocks for future advancements automated cybersecurity tools. The constant adaptation of phishing methods requires continuous AI-driven defense mechanisms research, protecting online users and organizations from the continually expanding collection of cyberattacks.

Tools & Technologies Used**:**

- Hardware:

Processor: A minimum of 2.5 GHz per core RAM: 4GB or higher

Hard Disk**:** A minimum of 80 GB

- Software:

Visual Studio Code

Front end technology:

HTML
JavaScript CSS
➢ Python Libraries: WHOis
Matplotlib Pandas NumPy Random Forest

V. RESULT AND DISCUSSION

Phishing websites are a significant security threat. Numerous cyberattacks jeopardize the confidentiality, integrity, and availability of both company and consumer data, with phishing often being the initial step in these attacks Researchers have dedicated decades to developing innovative methods for the automatic detection of phishing sites. Small and medium-sized enterprises (SMEs) benefit from ERP systems that centralize sales tracking and workforce operations to improve overall productivity.[9] Effective Customer Relationship Management (CRM) systems are essential for aligning marketing strategies with customer needs, ultimately boosting business outcomes.[10]

Proposed Solution*:*
To address the Phishing Website Detection problem more effectively, we propose the following solution:

1. Data Collection*:*
Start by gathering a dataset that includes both phishing and legitimate websites. Public datasets like PhishTank or UCI's phishing website dataset are commonly used. Ensure the data encompasses features such as URL length, domain age, SSL certificate status, and keywords found in the URL.

2. Feature Engineering:
Extract important features from each website, including:
- URL-based features (e.g., presence of "https", number of special characters, or URL length)
- Domain-based features (e.g., age, domain registration details)
- Content-based features (e.g., presence of login forms, suspicious scripts)

3. Preprocessing:
Clean and preprocess the data by normalizing numerical values, addressing missing values, and converting categorical features into numerical formats if needed.

3. Model Selection:
Choose a suitable machine learning model for classification. Some popular options include: Random Forest: An ensemble model that utilizes multiple decision trees to enhance accuracy.
Support Vector Machine (SVM): Effective for binary classification tasks.

Logistic Regression: A straightforward model ideal for binary classification.
Neural Networks: Suitable for identifying more complex patterns, though they may require additional data and computational resources.

4. Training:
Make training and testing sets out of the dataset. The model is trained using the training data using the features extracted from the websites.

5. Evaluation:
After training, use criteria like accuracy, precision, recall, and F1-score to assess the model's performance on the testing instance. These measures are going to show how effectively the model can differentiate between phishing and legitimate websites.

6. Deployment:
Once the model demonstrates satisfactory performance, it can be integrated into a web browser extension or security system. This system will classify new websites in real-time, alerting users to potential phishing threats.

7. Continuous Learning:
Regularly update the model by incorporating new data as phishing tactics change, ensuring it remains effective.

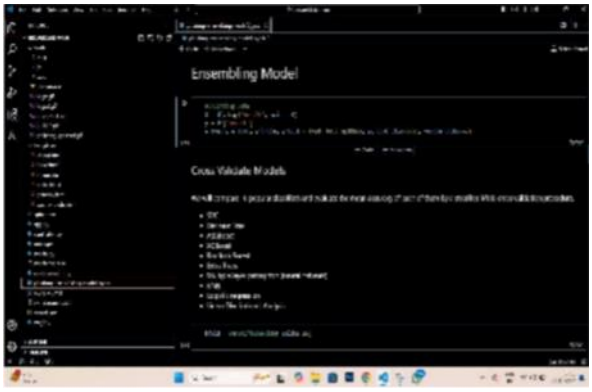

Fig: Ensembling the model

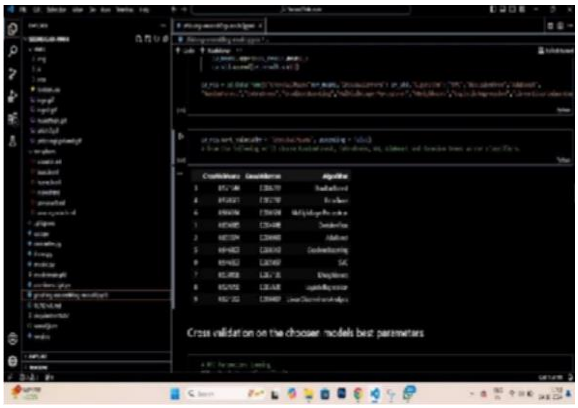

Fig: Algorithms and their values

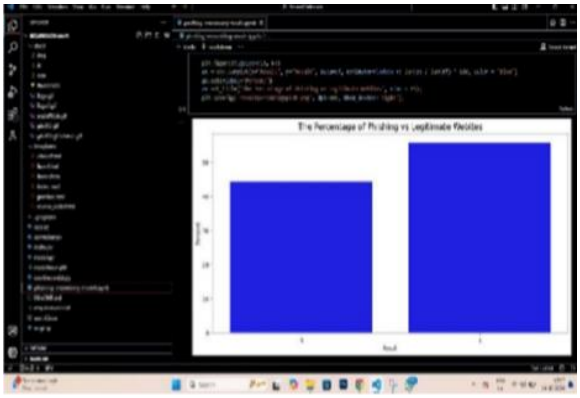

Fig: - 4.8 The percentages of the class values

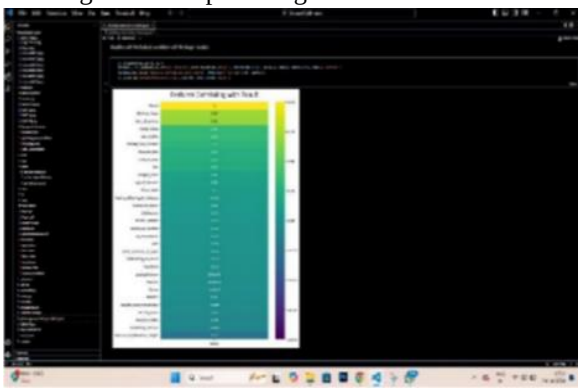

Fig: Variables with the highest correlation

VI. CONCLUSION

In conclusion, we have emphasized the serious threat that phishing poses to online security and the necessity of identifying such attacks. We explored various traditional methods for detecting phishing including blacklist and heuristic evaluation techniques, as well as their drawbacks. We evaluated three machine learning algorithms using the 'Phishing Websites Dataset' from the UCI Machine Learning Repository and assessed their performance.

By utilizing various web scraping and feature extraction techniques, the system identifies patterns that are often linked to phishing attacks, such as dubious URLs, unusual external link behavior, and unsafe content. While the system is operational, it mainly depends on established techniques and recognized patterns, which might not always catch new and more sophisticated phishing methods.

REFERENCE

[1] Wang, Y, Liu, Y, Wu, T, & Duncan, I. (2021). Cost–Effective OCR Implementation to prevent Phishing on Mobile Platforms [Journal-article]. University of St Andrews.

[2] Butnaru, A, Mylonas, A, & Petropolis, N. (2021). Towards Lightweight URL-Based Phishing Detection. Future Internet,13,154. https://doi.org/10.3390/fil3060154

[3] Alsariera, Y.A., Adeyemo,V.E., Balogun, A.O., &Alazzawai, A.K. (2020).AI Meta-Learners and Extra-Trees Algorithm the detection of Phishing Websites.IEEE Access,8,142532-142542.https://doi.org/10.1109/access.2020.3013699

[4] Garces,I.O, Cazares, M.F., &Andrade,R.O(2019). Detection of Phishing Attacks with Machine Learning Techniques in Cognitive Security Architecture. Hu. https://doi.org/10.1109/csci49370.2019.00071

[5] Lui,X.,Fu,J.,Keyboard Laboratory of Aerospace Information Security and Trusted Computing of Ministry of Education, School of Cyber Science and Engineering,Wuhan University,Wuhan 430072, China. (2020).SPWalk:Similar Property Oriented Feature Learning for Phishing Detection [Journalarticle]. https://doi.org/10.1109/ACCESS.2020.2992 381

[6] Madhavan,

[7] Jusman, I.A., Almaududi Ausat, A.M. and Sumarna, A. (2023) 'Application of chatgpt in Business Management and Strategic Decision making', Jurnal Minfo Polgan, 12(2), pp. 1688–1697. doi:10.33395/jmp. v12i2.12956.

[8] Choudhury, T., & Nayak, R. (2021). "Web-based Inventory Management Systems using MEAN Stack Technologies." This paper focuses on leveraging Node.js and MongoDB for dynamic inventory systems.

[9] Al-Mashari, M. (2019). "Integrated ERP Systems for SMEs: A Case of Employee Management and Sales Tracking," Journal of Systems and Software.

[10] Kumar, V., & Reinartz, W. (2018). Customer Relationship Management: Concept, Strategy, and Tools. This work provides a deep dive into CRM systems and their impact on business performance.