DOI: 10.22413/ijastems/2026/v12/i5/5

Phishguard: IoT Integrated NLP for Predictive Phishing URL Detection

P.Bhaskar, J. Pavithra, B. Pavani, K. Varshitha, G. Yashaswini
Department of Artificial Intelligence and Data Science,
Aditya College of Engineering,
Madanapalli, India.

Abstract— Phishing attacks are the major cybersecurity threat that deceive users into sharing sensitive information through the fake websites and malicious URLs. The project, PhishGuard: IoT integrated NLP for predictive phishing URL detection, it proposes an intelligent system to detect phishing links and protect users from fraud activities. This system uses Natural Language Processing and Machine Learning techniques to analyze URL structures and text patterns to identify suspicious characteristics. The extracted features are then processed by a predictive model that classifies URLs as legitimate or phishing in real time. IoT hardware integration enables continuous monitoring of network activity and supports automated data transmission for the analysis. The PhishGuard provides instant alerts when the malicious links are detected, helping users avoid phishing attacks and improving overall cybersecurity. The proposed system offers an efficient, scalable, and proactive solution for secure internet usage.

Index Terms— Phishing Detection, IoT, NLP, Machine Learning, URL Analysis, Cybersecurit

I. INTRODUCTION

In the current cyber world, where online interactions are a common practice, cyberattacks like phishing pose a threat to the security and privacy of users. A massive phishing attack can compromise the security of millions of people, compromising personal information, spreading ransomware, and gaining access to the most critical parts of a company's network. Phishing attackers have come up with a variety of methods to target different technologies, trends, sectors, and users of the world.

If the user tries to access a website through the Google Chrome browser, the browser should verify whether the given URL is safe or malicious. This can be achieved by sending the URL to a central server maintained by Google to verify the URL. This approach has several problems associated with it including the dependency on the central server. To solve this problem, the browser can store a local blacklist of malicious URLs to verify the given URL on the device itself. The problem with storing the local blacklist is that it requires a huge amount of storage space.

To solve the above problem, a Bloom Filter is used. A Bloom Filter is a space-efficient data structure that can test whether an object is a member of a set or not. It can solve the membership query problem in constant time, i.e., O(1).This means that Bloom Filters do not provide false negatives but may provide false positives. If we store 10^8 URLs using a Bloom Filter with a 1% false positive probability, we only need 114 MB storage space, which is significantly less than storing the URLs directly. Because of its high space efficiency and fast query time, Bloom Filters are used in different applications such as web proxy caching, IP traceback, virus scanning, weak password detection, safe browsing, etc.

Despite their efficiency, Bloom Filters still have some disadvantages. Furthermore, once elements are inserted into Bloom Filters, they cannot be easily removed, and as more elements are inserted, the false positive rate increases gradually. In addition, updating a blacklist using Bloom Filters also involves rebuilding a Bloom Filter. To overcome these disadvantages, a new data structure called a "Learned Bloom Filter" was proposed, which integrates a machine learning model and a Bloom Filter. In this structure, a machine learning model predicts whether a URL is malicious, and a small Bloom Filter is used as a backup in uncertain situations. It reduces storage space significantly, saving up to 60% of the space compared to Bloom Filters.

In order to address this concern, a framework called Defensive Learned Bloom Filter (DLBF) was proposed. It is a framework that intends to mitigate the effect of poisoning attacks and improve the resilience of the learned Bloom Filter in various applications, such as malicious URL detection, in terms of security and efficiency. Experimental studies using different real-world URL data sets showed that DLBF performs better than both Bloom Filters and existing variants of learned Bloom Filters in normal and adversarial scenarios. As a result, DLBF is a more secure and efficient solution for various applications, such as safe browsing systems, in terms of security and efficiency.

The rapid rise of internet access, along with the use of smart devices connected to the internet, has led to a rise in the potential for cyberattacks, especially phishing attacks. Phishing URLs are harmful URLs that attempt to access sensitive information, such as passwords, financial information, or personal information, from users. Traditional web browsers, such as Google Chrome, also provide protection against phishing attacks by verifying the URLs of the websites users have visited against a list of malicious URLs provided by Google and other such organizations

To solve the above issues, various methods like the Bloom Filter have been used to provide efficient storage in the malicious URL detection systems. Though the Bloom Filters provide efficient storage as well as fast lookup time for the malicious URLs, various issues like false positives, inability to identify newly created phishing URLs, as well

DOI: 10.22413/ijastems/2026/v12/i5/5

as difficulties in updating or deleting the URLs, have been associated with the Bloom Filters. Additionally, the phishing URLs being created today are of advanced types, generating new URLs that cannot easily be detected.

Recent advancements in the field of Natural Language Processing offer a chance to explore the textual patterns within the URLs to predict the malicious nature of the URLs. Using machine learning techniques combined with NLP, it would be possible to identify the patterns that are normally used in the URLs for phishing. This can be done even if the URLs have not been previously identified as malicious in the existing blacklists. Using such intelligent techniques would be helpful to improve the security of the IoT environment, as most IoT devices operate with limited resources.

Thus, the underlying idea behind the system "PHISHGUARD" project is to design an efficient and intelligent system that can be effectively detect phishing URLs using predictive analysis based on NLP, making it suitable to use in the IoT domain. The idea is to design a system that can efficiently overcome the limitations of traditional blacklisting techniques by using the efficient data structures along with machine learning algorithms to achieve faster, efficient and accurate phishing URL detection and ensuring security and safety for users providing a reliable security system for modern web surfing and IoT applications.

II. LITERATURE SURVEY

A. *Phishing URL Detection using Machine Learning Techniques.*

The research focuses on detecting phishing URLs using machine learning techniques. Phishing attacks commonly occur through the malicious links that trick users into revealing sensitive information such as passwords and financial details. The study analyzes different characteristics of URLs to identify suspicious patterns. Feature extraction is performed on the datasets to obtain necessary attributes such as URL length, domain structure and path segments. These features helps distinguishing between legitimate and phishing websites. The researchers apply various machine learning algorithms including decision Tree, Random forest and other classification models. The dataset is trained and tested and evaluated the performance of models. The results shows that ML algorithms can effectively classify URLs as safe or malicious. This improves detection accuracy by learning patterns from previous data. The research highlights the importance of automated detection systems to prevent cyber threats. It also demonstrates how feature-based analysis can enhance the reliability of phishing detection. The model can be integrated into security tools such as browsers or email filters. Overall, the study provides a strong foundation for developing intelligent phishing detection systems using machine learning methods.

B. *Detection of Email Spam using NLP based random Forest approach*

The paper presents an intelligent system for detecting phishing and the spam emails using NLP techniques. Phishing emails often can contain misleading content designed to trick users into the clicking fraud links or sharing confidential data. The research focuses on analyzing the text content of emails to identify the suspicious patterns. The NLP techniques such as tokenization, keyword extraction and text preprocessing are applied to analyze the email messages. The methods helps in understanding the structure and meaning of email content. After processing text data, the extracted features are used for classification The random forest algorithm is applied to classify emails as legitimate or phishing. Randon forest is chosen because it provides high accuracy and handles the large datasets effectively. The model can learn from the training data and predicts the nature of incoming emails. Context-based analysis also helps to improve detection performance by identifying hidden patterns in email messages. The study shows that combining NLP techniques with ML algorithms significantly improves phishing detection accuracy. The proposed approach can be implemented in email security systems to prevent phishing attacks. Overall the research demonstrates the effectiveness of NLP-based approaches for detecting emails.

C. *Heuristic Phishing Detection and URL Checking Methodology Based on Scraping and Web Crawling*

The research proposes a heuristic-based approach for detecting phishing websites. Instead of relying completely on machine learning models, the study uses the rule-based techniques to identify suspicious URLs. The system analyzes the structure of URLs and checks for unusual patterns that commonly appear in phishing websites. Important factors such as domain domain names, path segments and suspicious characters are examined carefully. The proposed system also uses web scraping techniques to collect webpage content from URLs. Web crawling is used to retrieve additional information from linked webpages. These techniques help analyze the actual content and structure of websites. Several heuristic rules are defined to detect the phishing indicators. For example, long URL strings, abnormal domain names, and unusual webpage structure are treated as the warning signs. The system evaluates URLs based on these predefined rules. If multiple suspicious characteristics are detected, the URL is classified as a phishing link. This approach does not require large training datasets and can work effectively with limited data. It is also easier to understand and interpret compared to complex machine learning models. The study highlights the usefulness of heuristic analysis in identifying malicious websites and improving cybersecurity systems.

D. *Phish Detect: A BiLSTM-based Phishing URL detection Framework using Fast Text embeddings.*

The paper introduces a deep learning framework called as the Phish Detect for detecting phishing URLs. The main objective of the research is to improve phishing detection accuracy using advanced neural network techniques. The system processes URLs by breaking them into smaller

DOI: 10.22413/ijastems/2026/v12/i5/5

components using tokenization. These tokens represent different parts of the URL such as domain names, paths and parameters. The study uses fast text embeddings to convert these tokens into numerical vector representations. Fast Text helps capture semantic relation between different tokens in the URL. After embedding, the data is fed into a Bidirectional Long Short-Term Memory neural network. BiLSTM is capable of learning sequential patterns from both forward and backward directions. This helps the model understand complex relations within the URL structure. The deep learning model learns patterns that distinguish phishing URLs from legitimate Ones compared to traditional machine learning methods, deep learning models can capture deeper contextual information. The system achieves improved detection accuracy and better performance in identifying phishing websites. The proposed approach is suitable for real-time phishing detection systems. Overall, the research demonstrates how deep learning and embedding techniques can significantly enhance cybersecurity solutions.

E. *An Efficient Machine Learning Approach for Phishing Website Detection*

The research paper focuses on developing an efficient system for detecting the phishing websites using machine learning techniques. Phishing attacks are one of the most common cybersecurity threats where attackers create fake websites to steal sensitive user information such as login credentials, banking details and personal data. The main objective of the study is to design a detection framework that can accurately distinguish phishing websites from legitimate websites. The proposed system analyzes several important features of URLs and webpages to identify the suspicious patterns. These features include domain information, URL length, special characters and webpage characteristics. Machine learning algorithms are used to classify websites based on these extracted features. The model is trained using a dataset containing both phishing and legitimate website samples. After training, the system evaluate new URLs to determine whether they are malicious or safe. The study compares the performance of different machine learning models to find the most accurate one for the phishing detection. Experimental results show that the proposed approach achieves high accuracy in identifying phishing websites. The system also reduces false positives and improves reliability in cybersecurity applications. The research highlights the importance of automated phishing detection systems to protect users from online fraud.

III. PROBLEM METHOD

To overcome the limitations of existing approaches, the proposed system introduces an intelligent phishing detection framework using machine learning and Natural Language processing techniques. The proposed system performs URL classification by analyzing the structure and patterns of URLs. A Recurrent Neural Network model is trained to recognize common characteristics of phishing URLs, such as suspicious keywords, unusual domain structures and misleading subdomains. In addition to URL analysis, the system also performs email classification. The system parses email content to identify suspicious indicators such as misleading hyperlinks, malicious attachments and the spoofed sender information. By analyzing both URL features and email content, the system can detect phishing attempts more accurately. The system integrates the results from both classifiers to determine the probability of phishing activity. If the likelihood exceeds a certain threshold, than the system flags the content as suspicious or malicious. The architecture also includes IoT-enabled monitoring. Smart devices such as routers or network nodes continuously capture incoming URLs and send them to a cloud-based NLP engine for analysis.

The NLP model categorizes the analyzed URLs into three classes:

- Phishing
- Defacement
- Malware
- Benign

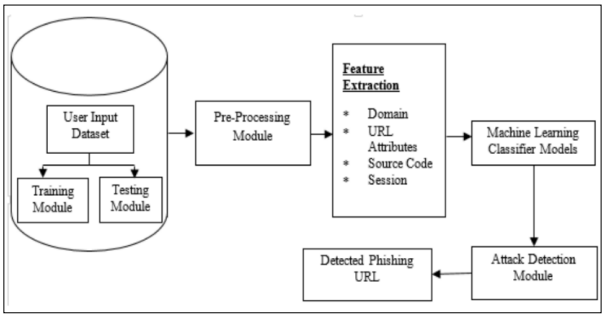
This multi-layered approach significantly improves phishing detection accuracy and enables proactive security protection.

IV. PROPOSED ALGORITHM

Deep learning, a specialized branch of Artificial Intelligence, has emerged as a powerful approach for solving complex problems involving large-scale and unstructured data. In the context of this project, deep learning is utilized to detect and classify malicious URLs into categories such as phishing, malware, defacement, and benign.

Traditional machine learning methods rely heavily on manual feature extraction, where domain experts must define characteristics of URLs such as length, special characters, and domain information. However, these approaches often fail to capture hidden and evolving patterns used by attackers. To overcome these limitations, deep learning techniques-especially Recurrent Neural Networks are employed in this system.

Recurrent Neural Networks are particularly effective



for sequential data analysis, making them suitable for processing URLs as sequence of characters.

Figure 1: Process of Deep Learning

DOI: 10.22413/ijastems/2026/v12/i5/5

The proposed system follows a structured pipeline consisting of data collection, preprocessing training, and prediction phases. Initially a large dataset of labelled URLs is collected, where each URL is categorized into predefined classes. This dataset is then pre-processed by cleaning, tokenizing, and converting text into numerical representations suitable for the model input.

During the training phase, the deep learning model learns patterns associated with each category. The training process involves adjusting weights using optimization techniques to minimize prediction error. Once trained, the model can generalize well to unseen data.

In the prediction phase, when the user inputs a URL, the system can process it through the trained model, which outputs the probability of the URL belonging to each category. Based on the highest probability, the system classifies the URL and displays the result.

V. BLOCK DIAGRAM

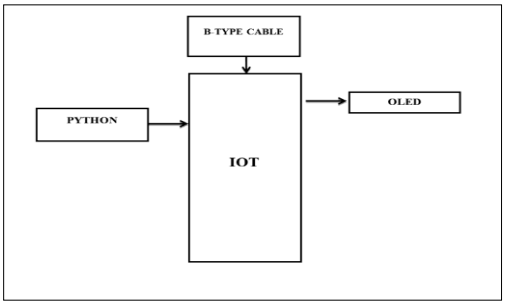


Figure 2: Block Diagram

The block diagram represents the overall architecture and working flow of the proposed PhishGuard system. It illustrates how different components interact to perform phishing URL detection and display the results through the IoT-based hardware interface.

In this system, python is used as primary programming environment to implement the phishing detection algorithm using Natural Language processing techniques. The python program analyzes the input URL and processes it to determine whether the link is legitimate or malicious.

The processed information is then communicated to the IoT module through a B-Type cable, which acts as the interface between the software system and the hardware device. The IoT module receives the processed data and controls the output operations.

Finally, the detection result is displayed on an OLED display, allowing users to easily view whether the URL is safe or phishing link. This integration of software analysis and IoT hardware enables real-time monitoring and efficient phishing detection.

Python Module

Python is used as the main programming environment for implementing the phishing detection system. It processes the input URL and analyzes it using Natural Language Processing (NLP) and machine learning techniques. The algorithm extracts important features from the URL and predicts whether the link is legitimate or phishing.

B-Type Cable

The B-Type cable acts as a communication interface between the computer system and the IoT hardware module. It transfers the processed data from the Python program to the IoT device. This cable ensures stable and reliable data transmission between the software and hardware components.

IoT Module

The IoT module works as the central processing unit of the hardware system. It receives the analyzed data from the Python program and performs the required actions. The module helps in integrating the software-based phishing detection system with the physical hardware components.

OLED Display

The OLED display is used to present the output of the phishing detection system. After the IoT module receives the prediction result, the information is displayed on the OLED screen. It shows whether the entered URL is safe or a phishing link, providing a simple and clear interface for users.

VI. RESULT AND DISCUSSION

The performance of the proposed PhishGuard system was evaluated using different machine learning and deep learning models such as decision Tree classifier, GRU, LSTM. These models were trained and tested using phishing URL and text datasets to determine their effectiveness in identifying malicious links and phishing content.

During the evaluation process, various performance metrics such as accuracy, precision, recall, and confusion matrix were analyzed to measure the effectiveness of the models. The results show that the system is capable of accurately distinguishing between legitimate and phishing URLs.

Among the implemented models, GRU and DTC demonstrated better performance in handling sequential data and identifying hidden patterns in URLs and text. The Decision Tree classifier also provided reliable classification results with faster processing time The experimental results indicate that the proposed system can effectively detect phishing attacks with high accuracy.

The analysis confirms that integrating machine learning, deep learning, and NLP techniques improves the overall performance of phishing detection systems. The

DOI: 10.22413/ijastems/2026/v12/i5/5

developed system successfully predicts phishing URLs and text content, thereby helping users avoid malicious websites and the online fraud.

The database screens confirm that all user inputs, along with their corresponding prediction results, are being stored properly for future reference and analysis. This ensures data persistence and improves system reliability. Furthermore, the integration of the IoT module is validated through the OLED display, which shows real-time results corresponding to the predictions generated by the system.

Overall, the seamless interaction between input modules, processing models, output visualization, and hardware integration proves that the system is well-structured and efficiently implemented. The results across all screens are consistent and validate the effectiveness of the proposed phishing detection system in rel-time scenarios.

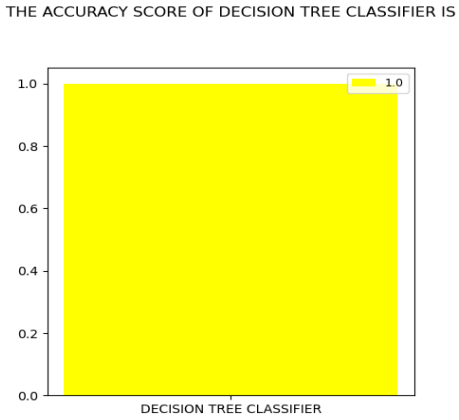


Figure 3: Accuracy of Decision Tree classifier

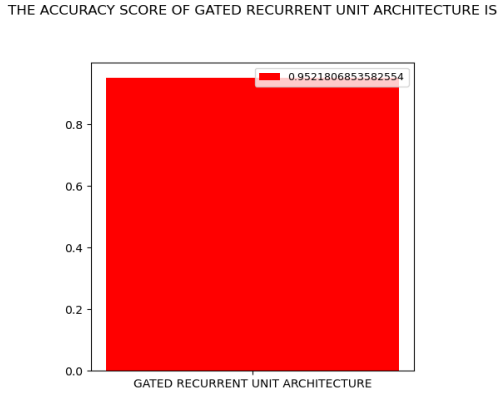


Figure 4 : Accuracy of Gated Recurrent Unit

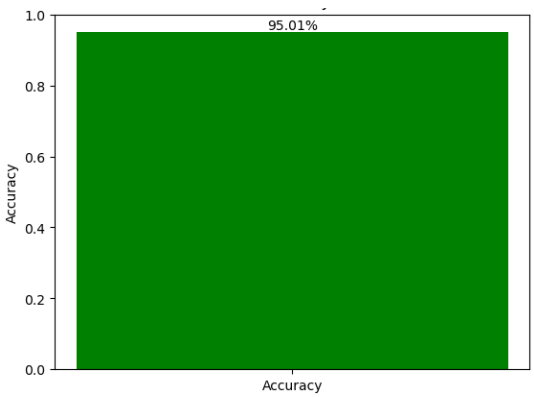


Figure 5: Accuracy of Long short Term Memory

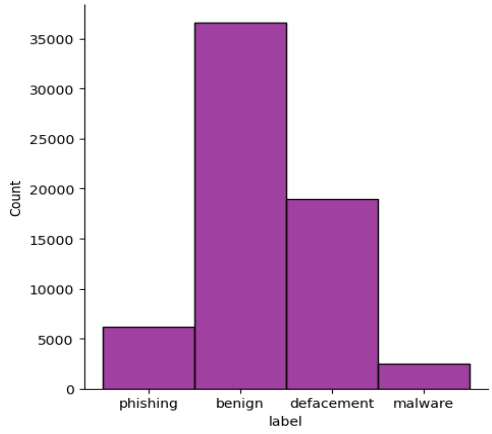


Figure 6: Distribution plot

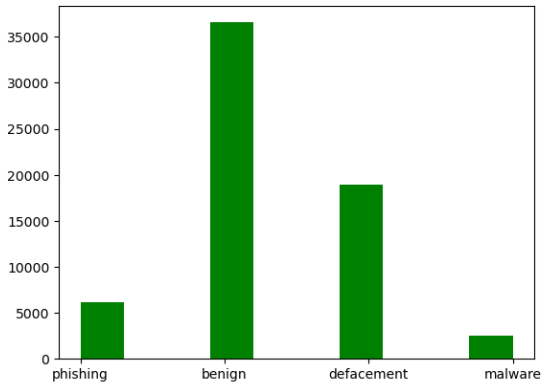


Figure 7: Histogram

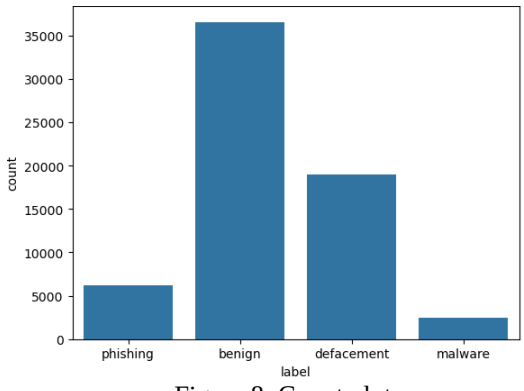


Figure 8: Count plot

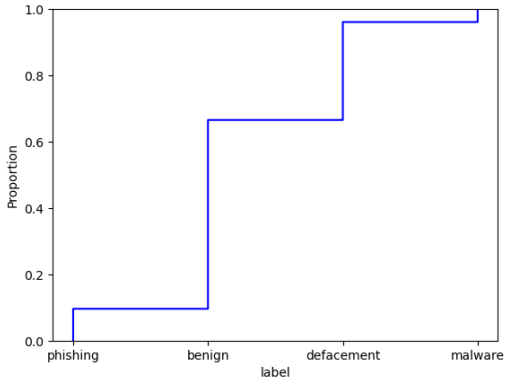


Figure 9: ECDF plot

DOI: 10.22413/ijastems/2026/v12/i5/5

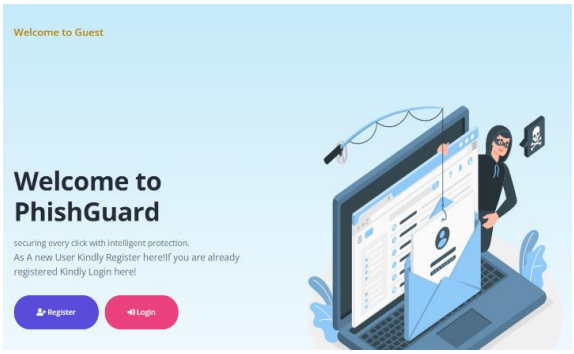


Figure 10: PhishGuard Website

VII. CONCLUSION

ThePhishGuard: IoT integrated NLP for Predictive Phishing URL detection system was developed to identify and prevent phishing attacks by analyzing suspicious URLs and text content. With the increasing number of cyber threats and phishing activities, it is important to provide an efficient solution that can help users to detect malicious links and messages.

The proposed system was designed to identify Phishing URLs and suspicious text messages using Machine learning and deep learning techniques. The system integrates different technologies including NLP and ML algorithms, and IoT-based system monitoring to provide an efficient phishing detection mechanism.

In the project, multiple algorithms such as DTC, GRU, LSTM were implemented and evaluated for phishing detection. These algorithms help in analyzing the structure and patterns present in URLs and text messages. The Decision tree classifier provides fast and effective classification based on extracted features, while GRU and LSTM are deep learning models capable of learning complex sequential patterns in textual data.

The experimental results obtained from the system demonstrates that proposed approach can effectively detect phishing URLs and suspicious text content. The models used in the system are capable of learning patterns from phishing datasets and providing reliable classification results.

Therefore, the PhishGuard system provides a reliable and efficient solution for phishing detection using advanced technologies such as machine learning, deep learning, and NLP. The system contributes to improving online security by helping users to identify and avoid malicious websites and phishing messages. With further improvements and real-time data integration, the system can be extended into a more advanced cybersecurity solution capable of protecting users from evolving phishing threats.

ACKNOWLEDGEMENT

It is our privilege and pleasure to express our profound sense of respect, gratitude and indebtedness to Dr. S. RAMALINGA REDDY, Director for guiding and providing facilities for the successful completion of our project work.

It is our privilege and pleasure to express our profound sense of respect, gratitude and indebtedness to Dr. RAYUDU PEYYALA, principal for guiding and providing facilities for the successful completion of our project work.

We sincerely thank to Dr. R. MD. SHAFI, Head of the Department, Department of Artificial Intelligence and Data Science, for his valuable support and constant encouragement given to us during the entire journey of our project work.

We express our deep sense of gratitude to Mr. P. Bhaskar, project Guide for supporting and encouraging at each stage of our project work and guided us to do our best.

REFERENCE

[1] S. Alnemari and M. Alshammari, “Detecting Phishing Domains Using Machine Learning,” Applied Sciences, vol. 13, no. 8, p. 4649, Apr. 2023, doi: 10.3390/app13084649.

[2] Lanjewar, Madhusudan G., et al. “Small Size CNN-Based COVID-19 Disease Prediction System Using CT Scan Images on PaaS Cloud.” Multimedia Tools and Applications, vol. 83, no. 21, Jan. 2024, pp. 60655–87.

[3] P. Pandey and N. Mishra, “Phish-Sight: a new approach for phishing detection using dominant colors on web pages and machine learning,” Int J Inf Secur, vol. 22, no. 4, pp. 881–891, Aug. 2023, doi: 10.1007/s10207-023-00672-4.

[4] Montasari, R., Carroll, F., Macdonald, S., Jahankhani, H., Hosseinian-Far, A., & Daneshkhah, A. (2021). Application of artificial intelligence and machine learning in producing actionable cyber threat intelligence. Digital Forensic Investigation of Internet of Things (IoT) Devices, 47-64.

[5] Pan, Y., & Zhang, L. (2021). Roles of artificial intelligence in construction engineering and management: A critical review and future trends. Automation in Construction, 122, 103517. https://doi.org/10.1016/j.autcon.2020.103517

[6] Subasi and E. Kremic, “Comparison of adaboost with multiboosting for phishing website detection,” Procedia Computer Science, vol. 168, pp. 272–278, 2020. doi:10.1016/j.procs.2020.02.251

[7] Devi, Mampi, et al. “MCM-$$V_b$$F: Dance Hand Gestures Recognition with Vision Based Features.” Discover Internet of Things, vol. 4, no. 1, Oct. 2024, p. 18.

[8] F. Çolhak, M. İ. Ecevit, B. E. Uçar, R. Creutzburg, and H. Dağ, “Phishing Website Detection through Multi-Model Analysis of HTML Content,” arXiv (Cornell University), Jan. 2024, doi: 10.48550/arxiv.2401.04820.

[9] M. W. Shaukat, R. Amin, M. M. A. Muslam, A. H. Alshehri, and J. Xie, “A Hybrid Approach for Alluring Ads Phishing Attack Detection Using Machine Learning,” Sensors, vol. 23, no. 19, p. 8070, Sep. 2023, doi: 10.3390/s23198070.

[10] D. Minh Linh, H. D. Hung, H. Minh Chau, Q. Sy Vu, and T.-N. Tran, “Real-time phishing detection using deep learning methods by extensions,” International Journal of Electrical and Computer Engineering (IJECE), vol. 14, no. 3, p. 3021, Jun. 2024, doi: 10.11591/ijece.v14i3.pp3021-3035.

[11] L. M. Abdulrahman, S. H. Ahmed, Z. N. Rashid, Y. S. Jghef, T. M. Ghazi, and U. H. Jader, “Web Phishing Detection Using Web Crawling, Cloud Infrastructure and Deep Learning Framework,” Journal of Applied Science

and Technology Trends, vol. 4, no. 01, pp. 54–71, Mar. 2023, doi: 10.38094/jastt401144.

[12] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems," Heliyon, vol. 5, no. 6, p. e01802, Jun. 2019, doi: 10.1016/j.heliyon.2019.e01802.

[13] Abdel-Rahman, M. (2023). Advanced cybersecurity measures in IT service operations and their crucial role in safeguarding enterprise data in a connected world. Eigenpub Review of Science and Technology, 7(1), 138-158.

[14] Panchbhai, Kamini G., et al. "Non-Destructive Beer Gravity Measurement Using Spectroscopy and Machine Learning with mRMR-Based Wavelength Selection." Journal of Food Science and Technology, Oct. 2025. DOI.org (Crossref), https://doi.org/10.1007/s13197-025-06482-x.

[15] Ahmed, I., Jeon, G., & Piccialli, F. (2022). From artificial intelligence to explainable artificial intelligence in industry 4.0: a survey on what, how, and where. IEEE Transactions on Industrial Informatics, 18(8), 5031-5042.

[16] Azaria, A., Richardson, A., Kraus, S., & Subrahmanian, V.S. (2014). Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data. IEEE Transactions on Computational Social Systems, 1(2), 135-155.

[17] Busuioc, M. (2021). Accountable artificial intelligence: Holding algorithms to account. Public Administration Review, 81(5), 825-836.