

AI-Powered Browser Threat Guardian

S J Srikanth Dhanunjay
Department of Computer
Science and Engineering
Vardhaman College Of
Engineering
Shamshabad, India
sjsrikanthdhanunjay@gmail.com

Balguri Rishitha
Department of Computer
Science and Engineering
Vardhaman College Of
Engineering
Shamshabad, India
rishithareddybalguri@gmail.com

Sabavath Arjun
Department of Computer
Science and Engineering
Vardhaman College Of
Engineering
Shamshabad, India
arjunonline63@gmail.com

Nagashetty B. Kolar
Assistant Professor
Computer Science and
Engineering Vardhaman College
of Engineering,
Shamshabad, India
kolarnb@gmail.com

Abstract - As the use of web applications continues to grow rapidly, users are becoming more vulnerable to various cyber threats such as phishing websites, harmful scripts, unsafe downloads, and deepfake content. Many existing browser security solutions depend on cloud-based analysis, which can lead to concerns about user privacy and data protection. To overcome these limitations, this project presents an AI-based browser extension that performs real-time threat detection directly on the user's device. The system examines URLs, webpage structures, and multimedia elements using lightweight machine learning models. Since all processing is carried out locally, no browsing information is shared with external servers. The extension highlights suspicious components on webpages and provides immediate warnings to users about possible risks. This method offers an efficient and privacy-focused approach to improving browser security against modern cyber threats.

Keywords - privacy-focused security, local AI-based analysis, web threat identification, browser protection, safeguarding user data.

I. INTRODUCTION

In the modern digital age, web browsers have become a primary tool for accessing a wide range of online services, including banking, shopping, education, entertainment, and social media platforms. With the rapid increase in internet usage worldwide, users now depend heavily on browsers for their everyday activities. However, this growing reliance has also increased exposure to various cyber threats such as phishing websites, harmful scripts, unsafe downloads, and deepfake content [7]. These types of attacks can deceive users into sharing sensitive information, which may result in financial losses, identity theft, and significant privacy risks [8]. As cyber threats continue to evolve, ensuring secure and reliable browsing has become more important than ever.

Conventional browser security methods mainly depend on rule-based techniques and blacklist-based detection, where website URLs are checked against databases of known malicious links [10]. Although these methods are effective in identifying previously reported phishing sites, they often fail to detect newly created phishing pages that frequently change their structure and domain patterns [9]. Moreover, many current security tools rely on cloud-based processing, where user browsing data is transmitted to external servers for analysis [15]. This practice raises serious concerns about user privacy and the possible misuse of sensitive information.



Fig. 1: Browser security using AI.

To overcome these challenges, this project introduces an AI-based browser threat detection system that uses multiple lightweight machine learning models integrated into a browser extension [17]. The system examines various elements of a webpage, including URL features, HTML and DOM structures, and multimedia content. The outputs from these individual modules are then combined using a decision-level fusion approach to produce a final classification of the webpage as safe or malicious [12].

The proposed system is designed to function efficiently on the client side, where most of the detection processes are handled using CPU-based machine learning models, while deep learning components are optimized for lightweight performance. Advanced neural network models such as MobileNet and EfficientNet are specifically developed for environments with limited resources, such as mobile devices and web browsers [14]. By carrying out all analysis within the browser instead of using cloud-based services, the system enhances user privacy while still ensuring effective detection of phishing websites and manipulated media content [2][3].

II. Literature Survey

Web Phishing Detection

Phishing attacks are among the most common cybersecurity threats on the internet, and researchers have proposed various machine learning techniques to detect such attacks by analyzing URL patterns, webpage structures, and domain-related features. Initially, detection methods mainly relied on blacklist-based systems, where URLs were compared with databases of known malicious websites [10]. Although effective for identifying previously reported threats, these methods are not capable of detecting newly created phishing sites that can easily bypass such lists. To address this limitation, later research introduced machine learning models such as Logistic Regression, Support Vector Machines (SVM), and Random Forest [7]. These approaches analyze features extracted from URLs and webpage content to learn patterns associated with malicious behavior, resulting in improved accuracy and better adaptability compared to traditional rule-based methods.

Machine Learning for Web Threat Detection

Recent research has focused on applying advanced machine learning techniques, including XGBoost and other ensemble-based models, to improve the detection of malicious web pages. These approaches evaluate various features such as URL length, use of special characters, unusual domain patterns, and suspicious HTML or JavaScript activities. Studies have shown that tree-based models like Random Forest are highly effective for phishing detection, as they handle structured data efficiently and provide consistent predictions [12]. In a similar way, XGBoost has gained popularity in cybersecurity tasks due to its strong performance, scalability, and ability to handle large datasets effectively [11].

Deepfake Detection Techniques

With the rapid progress of deep learning, deepfake images and videos have become a major concern in the areas of online security and misinformation [5]. To address this issue, researchers have developed several deep learning models based on Convolutional Neural Networks (CNNs) to identify manipulated facial content. Among these, the XceptionNet architecture is widely used as a benchmark model due to its strong performance in image classification

tasks [2]. In addition, large datasets such as FaceForensics++ and Celeb-DF have been introduced to support the training and evaluation of deepfake detection models [4][5]. The DeepFake Detection Challenge (DFDC) dataset also provides extensive data that helps improve the robustness of models against different types of media manipulation [6].

Integrated Threat Detection Approaches

Recent studies highlight the importance of combining multiple detection methods to enhance overall threat detection performance. Hybrid approaches that simultaneously examine URLs, webpage structures, and multimedia content are capable of identifying a wider range of cyber threats more effectively. Decision fusion methods, such as Logistic Regression, are often used to combine the outputs from different detection models and generate a final prediction [12]. At the same time, lightweight deep learning models like MobileNet are gaining popularity due to their ability to deliver efficient performance in environments with limited computational resources, such as mobile devices and browser-based systems [14]. Based on these advancements, the proposed system integrates URL analysis, HTML/DOM analysis, and deepfake detection within a browser extension to achieve a comprehensive and privacy-focused web security solution.

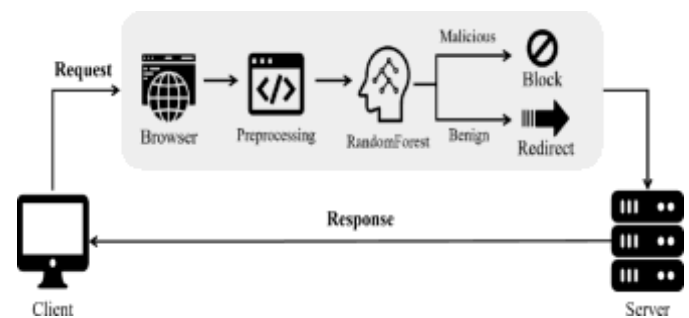


Fig. 2: Browser phishing detection using Random Forest to block malicious sites.

The review of existing studies indicates that various techniques have been proposed to detect phishing websites and deepfake content using both machine learning and deep learning approaches. However, traditional methods such as blacklist-based systems and cloud-driven solutions face challenges in identifying newly emerging threats and may also create concerns related to user privacy. To overcome these issues, the proposed system incorporates lightweight machine learning and deep learning models within a browser extension to enable efficient and privacy-focused threat detection. The system enhances real-time analysis by examining multiple aspects, including URL features, webpage structure, and multimedia content. It minimizes reliance on external servers, thereby improving data protection and user confidence [8]. Overall, this approach provides a more flexible and effective solution for handling modern web security challenges. Additionally, the use of lightweight models ensures faster processing with minimal resource usage.

III. PROPOSED MODEL

The proposed system, known as the **AI-Powered Browser Threat Guardian**, is developed as a browser extension aimed at enhancing web security by detecting phishing websites and manipulated media content. It combines multiple machine learning and deep learning models to examine various parts of a webpage. These include the URL, the structure of the webpage, and any multimedia elements present. By analyzing these components together, the system can identify potential threats more effectively and improve overall browsing safety. The first component of the system is the **URL Detection Model**, which is responsible for identifying suspicious or phishing links. It extracts several features from the URL, such as its length, the use of IP addresses, domain-related patterns, and the presence of special characters. These features are then processed using the XGBoost classifier, which is known for its efficiency in handling structured data [11]. Research has shown that models trained on URL-based features can successfully detect phishing websites [7].

The second component is the **HTML/DOM Feature Detection Model**, which focuses on analyzing the structure of the webpage. Phishing sites often include irregular HTML elements, hidden input forms, or unusual JavaScript behavior. The system extracts these structural features and processes them using a Random Forest classifier, a model well-suited for handling structured data [12]. This method helps in improving detection accuracy by identifying abnormal patterns within the webpage structure.



Fig.3: AI-powered browser extension detecting and warning users about a phishing website in real time.

The results from these modules are further integrated with a **Deepfake Detection Model**, which is responsible for identifying manipulated images or videos present on webpages. For this purpose, a lightweight convolutional neural network, **MobileNetV2**, is used, as it is well-suited for environments with limited computational resources [14]. The final classification is then produced using a **Logistic Regression-based fusion model**, which combines the outputs

from all individual modules to determine whether the webpage is safe or potentially harmful. This combined multi-model strategy enhances detection accuracy while ensuring efficient and lightweight operation.

IV. EXISTING METHODS

1. Blacklist-Based Phishing Detection

A widely used technique for identifying phishing websites is the blacklist-based method. In this approach, the browser checks the URL of a webpage against a database containing previously reported malicious links. Services like Google Safe Browsing maintain and update such lists to block access when a suspicious match is found [17]. While this method is simple and effective for detecting known threats, it has limitations. It cannot identify newly created phishing websites or domains that frequently change, which reduces its effectiveness in real-time threat detection [10].

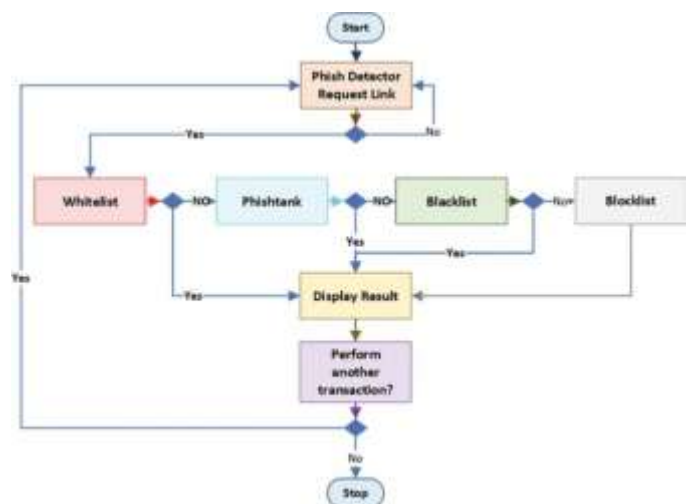


Fig. 4: Illustration showing how browsers compare website URLs with a blacklist database to block known phishing sites.

2. Rule-Based and Heuristic Detection

Another commonly used traditional approach in web security is rule-based detection, which relies on predefined rules and heuristic techniques to identify suspicious webpages. These rules examine factors such as unusual URL patterns, excessive use of special characters, hidden input forms, and abnormal webpage behavior. Although this method can detect certain types of threats, it largely depends on manually created rules. As attackers continuously adapt their techniques, they can easily modify phishing pages to avoid these fixed rules, making this approach less effective against advanced and evolving cyber threats.

3. Machine Learning-Based Phishing Detection

Recent advancements in research have introduced machine learning techniques to enhance the accuracy of phishing detection. In these approaches, algorithms analyze features derived from URLs, HTML structures, and overall webpage behavior to determine whether a site is legitimate or malicious.

Models such as Support Vector Machines, Decision Trees, and Random Forest classifiers are commonly used for this purpose [16]. These systems are capable of learning patterns from training data, allowing them to identify suspicious websites more effectively than traditional rule-based methods. However, some machine learning models may require large datasets and higher computational resources during the training process.

4. Deep Learning-Based Deepfake Detection

With the rapid development of artificial intelligence, deep learning methods have been widely used to detect manipulated images and videos, commonly referred to as deepfakes. Models based on Convolutional Neural Networks (CNNs), such as XceptionNet and EfficientNet, are frequently applied to identify visual irregularities in such content. These models examine facial features, texture details, and frame-level inconsistencies to recognize signs of manipulation. Although deep learning techniques offer high accuracy, many of these models are computationally intensive and are mainly designed for server-based systems, making them less suitable for lightweight browser-based applications.

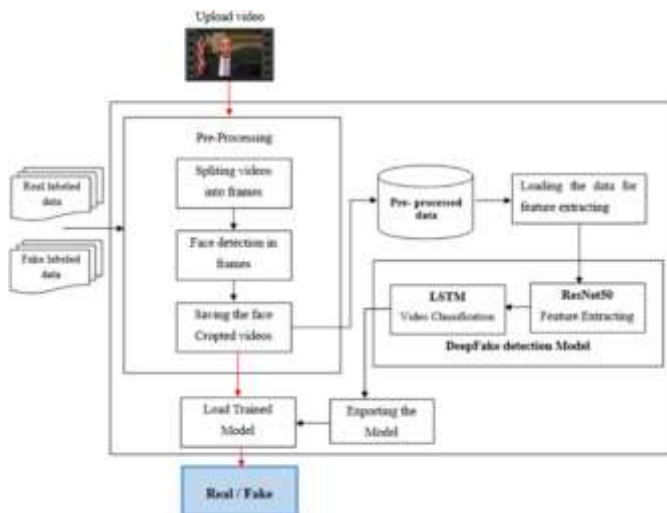


Fig. 5: Conceptual representation of deep learning models analyzing facial features to detect deepfake images or videos.

V. SYSTEM DESIGN

The overall design of the **AI-Powered Browser Threat Guardian** aims to provide secure and real-time threat detection directly within the browser environment. The system is developed as a browser extension that continuously monitors the webpages accessed by the user and analyzes multiple aspects of each page. These aspects include the URL, the structural elements of the webpage, and any multimedia content present. By evaluating these components, the system can detect potential phishing sites and manipulated media before the user interacts with them.

The system is built using a **multi-model architecture**, where each machine learning model is assigned a specific task. The URL detection module extracts important features from the webpage URL and uses the XGBoost model to identify

suspicious patterns. The HTML/DOM analysis module studies webpage elements and embedded scripts using a Random Forest classifier to detect abnormal or phishing-related structures. In addition, the system examines multimedia content through the MobileNetV2 deep learning model to identify any deepfake images or videos that may be present on the webpage.

The outputs from all these modules are then combined using a **fusion decision model based on Logistic Regression**, which generates the final classification of the webpage as either safe or malicious [14]. This approach ensures efficient threat detection by using lightweight models that are suitable for browser-based environments. Since all analysis is performed locally on the user's device, the system reduces reliance on cloud services and enhances user privacy while still providing real-time protection against modern cyber threats.

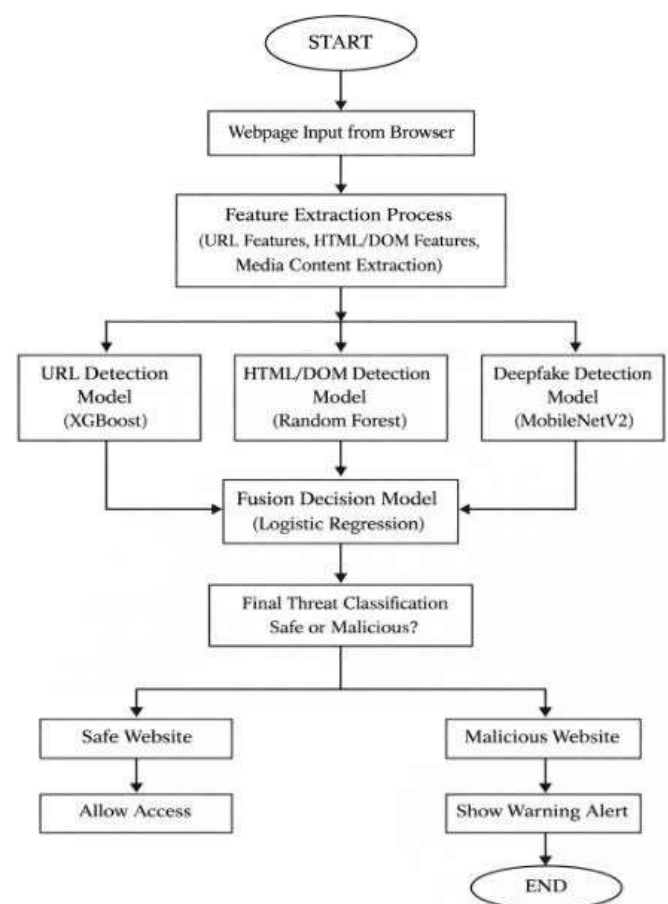


Fig. 6: Block Diagram

Algorithms Used in the Proposed System

1. XGBoost (Extreme Gradient Boosting)

Step 1: Start.

Step 2: Read the webpage URL from the browser.

Step 3: Extract URL features such as length, special characters, domain type, and presence of IP address.

Step 4: Convert the extracted features into a structured input format.

Step 5: Apply the XGBoost model on the extracted features.

Step 6: Analyze the output score to determine if the URL is safe or phishing.

Step 7: Store the result as URL risk score.

Step 8: Stop.

2. Random Forest Algorithm

Step 1: Start.

Step 2: Read the HTML content of the webpage.

Step 3: Extract DOM features such as forms, scripts, hidden elements, and suspicious tags.

Step 4: Convert these features into structured data.

Step 5: Apply the Random Forest model on the extracted features.

Step 6: Combine results from multiple decision trees to get prediction.

Step 7: Determine whether the webpage structure is safe or suspicious.

Step 8: Store the result as HTML risk score.

Step 9: Stop.

3. MobileNetV2 (Deep Learning Model)

Step 1: Start.

Step 2: Check if the webpage contains images or videos.

Step 3: If media is present, extract frames or images.

Step 4: Preprocess the images (resize, normalize).

Step 5: Input the processed images into the MobileNetV2 model.

Step 6: Analyze visual features to detect manipulation or deepfake patterns.

Step 7: Classify media as real or fake.

Step 8: Store the result as media risk score.

Step 9: Stop.

4. Logistic Regression (Fusion Model)

Step 1: Start.

Step 2: Read inputs from URL risk score, HTML risk score, and media risk score.

Step 3: Combine all scores into a single feature vector.

Step 4: Apply Logistic Regression model on combined inputs.

Step 5: Calculate final probability of threat.

Step 6: IF probability is high, THEN classify as malicious.

Step 7: ELSE classify as safe.

Step 8: Display result to user (warning or safe message).

Step 9: Stop.

VI. EXPERIMENTAL RESULT

The performance of the proposed **AI-Powered Browser Threat Guardian** was tested using datasets that included phishing URLs, webpage structures, and deepfake media samples. The system was able to successfully extract important features from URLs, HTML/DOM elements, and multimedia content for analysis. The XGBoost and Random Forest models effectively identified patterns commonly associated with phishing websites. This enabled the system to detect malicious webpages more accurately when compared to traditional rule-based methods. The models also demonstrated consistent performance across different types of webpages and varying input conditions. This highlights the robustness of the system in handling diverse and dynamic web environments.

For detecting manipulated media, the system utilized the lightweight **MobileNetV2** convolutional neural network. This model analyzed images and video frames to identify signs of manipulation or deepfake characteristics. The results showed that the model could reliably differentiate between genuine and altered media content. Due to its lightweight structure, the model ensured faster processing and required fewer computational resources, making it suitable for browser-based environments.



Fig. 6: Phishing Website Detection Alert

The final decision was generated by combining the outputs from all detection modules using a **Logistic Regression fusion model**. This approach improved the overall performance of the system by reducing both false positives and false negatives. The experimental findings indicate that the system is capable of performing real-time threat detection within the browser. In addition, the lightweight design allows the system to run efficiently while preserving user privacy, as all processing is carried out locally on the user's device.

VII. CONCLUSION

The proposed **AI-Powered Browser Threat Guardian** offers an effective solution for improving web security by detecting phishing websites and manipulated media content. With the rapid growth of online services, cyber threats such as phishing attacks and deepfake media have become major concerns for internet users. The system is designed to identify these threats in real time within the browser environment, allowing users to avoid interacting with harmful websites. By incorporating intelligent detection techniques, the system enhances overall browsing safety and provides a more secure and reliable user experience.

The system is based on a **multi-model architecture** that analyzes various components of a webpage, including the URL, HTML structure, and multimedia content. Each module focuses on detecting specific indicators commonly found in malicious webpages, which improves the overall reliability of the system. Machine learning models such as XGBoost and Random Forest are used to identify suspicious URL patterns and abnormal webpage structures by learning from data. This ability to recognize patterns enables the system to detect previously unseen threats, making it more adaptable and effective compared to traditional rule-based approaches.

To further strengthen detection capabilities, the system integrates a **MobileNetV2 deep learning model** to identify deepfake images and videos that may appear on webpages. As deepfake technology is increasingly used in scams and misinformation, detecting such content is essential for modern web security. The model analyzes visual inconsistencies to determine whether media content is genuine or manipulated. Its lightweight design ensures efficient performance even in resource-constrained environments, making it suitable for real-time implementation within browser extensions.

In the final stage, the results from all individual modules are combined using a Logistic Regression–based fusion model to determine whether a webpage is safe or potentially harmful. By considering multiple indicators together, this approach improves overall accuracy and reduces the chances of incorrect predictions. The system shows that lightweight machine learning models can be successfully integrated into browser extensions to provide real-time protection. It operates efficiently while maintaining user privacy, as all processing is carried out locally on the user's device. In the future, the system can be enhanced by incorporating adaptive learning to handle new types of threats, using user feedback to improve performance, and supporting multiple browsers. Additionally, integrating more advanced AI models can further strengthen its ability to detect evolving cyber threats.

REFERENCES

1. "DeepFake Detection for Human Face Images and Videos: A Survey," *IEEE Xplore*, 2022.
2. P. Yu, Z. Xia, J. Fei and Y. Lu, "A Survey on Deepfake Video Detection," *IET Biometrics*, vol. 10, pp. 607–624, 2021.
3. Y. Patel, S. Tanwar, R. Gupta, P. Bhattacharya, I. E. Davidson, R. Nyameko, S. Aluvala and V. Vimal, "Deepfake Generation and Detection: Case Study and Challenges," *IEEE Access*, vol. 11, pp. 143296–143299, 2023.
4. B.-X. Wang, J.-L. Chen and C.-L. Yu, "An AI-Powered Network Threat Detection System," *IEEE Access*, vol. 10, pp. 54029–54030, 2022.
5. "Deepfake Generation and Detection: Case Study and Challenges," Research Survey Document, 2023.
6. B.-X. Wang, J.-L. Chen and C.-L. Yu, "AI-Powered Network Threat Detection System," Research Document, 2022.
7. P. Yu, Z. Xia, J. Fei and Y. Lu, "A Survey on Deepfake Video Detection," Research Survey Document, 2021.
8. A. Alazab, A. Khraisat, M. Alazab and S. Singh, "Detection of Obfuscated Malicious JavaScript Code," *Future Internet*, vol. 14, no. 217, 2022.
9. A. Dewald, T. Holz and F. C. Freiling, "ADSandbox: Sandboxing JavaScript to Fight Malicious Websites," *Proceedings of the ACM Symposium on Applied Computing (SAC)*, pp. 1859–1860, 2010.
10. M. Cova, C. Kruegel and G. Vigna, "Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code," *Proceedings of the World Wide Web Conference (WWW)*, pp. 281–283, 2010.
11. A. Almomani, M. Alauthman, M. T. Shatnawi, M. Alweshah, A. Alrosan, W. Alomoush and B. B. Gupta, "Phishing Website Detection With Semantic Features Based on Machine Learning Classifiers: A Comparative Study," *International Journal on Semantic Web and Information Systems*, vol. 18, no. 1, 2022.
12. 14. A. Howard, M. Zhu, B. Chen, et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint arXiv:1704.04861, 2021.
13. M. Elsadig, A. O. Ibrahim, S. Basheer, M. A. Alohal, S. Alshunaifi, H. Alqahtani, N. Alharbi and W. Nagmeldin, "Intelligent Deep Machine Learning Cyber Phishing URL Detection Based on BERT Features Extraction," *Electronics*, vol. 11, 2022.
14. M. Gokulkannan, V. R. Reddy, S. V. Kumar and S. Abbas, "Phishing Detection System Through Hybrid Machine Learning Based on URL," *Journal of Electrical Systems*, vol. 20, 2024.
15. S. R. Abdul Samad, S. Balasubaramanian, A. S. Al-Kaabi, B. Sharma, S. Chowdhury, A. Mehbodniya, J. L. Webber and A. Bostani, "Analysis of the Performance Impact of Fine-Tuned Machine Learning Model for Phishing URL Detection," *Electronics*, vol. 12, 2023.
16. P. Maneriker, J. W. Stokes, E. Garcia Lazo, D. Carutasu, F. Tajaddodianfar and A. Gururajan, "URLTran: Improving Phishing URL Detection Using Transformers," Research Article, 2021.
17. R. Liu, Y. Lin, X. Yang, S. H. Ng and D. M. Divakaran, "Inferring Phishing Intention via Webpage Appearance and Dynamics: A Deep Vision Based Approach," *Proceedings of the 31st USENIX Security Symposium*, pp. 1633–1634, 2022.