Browser Extension for Phishing Website Detection Using Machine Learning

Dr. Shyni Shajahan*, Teena George, Jithi P V, Snehamol K M, Sanjana Krishna, Sanjaly Krishna, Nagul Jagadish
Department of Computer Science, Adi Shankara Institute of Engineering and Technology, Kalady, Ernakulam, Kerala, India
* Corresponding author: shyni.cs@adishankara.ac.in
doi: https://doi.org/10.21467/proceedings.7.5.5

ABSTRACT

Phishing remains a major cybersecurity concern, where attackers steal sensitive data. Traditional methods of detection, such as blacklists and rule-based systems, tend to fail to detect new or emerging threats. To address this gap, we propose a machine learning–based browser extension that detects phishing websites in real time. The extension silently operates in the background of a user's web browser, examining factors including URL structure, webpage content, domain legitimacy, and visual signals to accurately classify them. Fundamentally, at its center is an XGBoostclassifier that has been trained on a well-filtered and varied dataset. When a phishing attempt is recognized, users are promptly notified with no perceptible interruption to their browsing session. The system achieved 95% accurate during testing and has good precision and recall rates that indicate its robustness. This lightweight and efficient tool not only secures users against online attacks but also learns to keep up with evolving attack trends. In the future, we intend to investigate deep learning methods and add support to additional browsers, solidifying user security in cyberspace.

Keywords: Browser Extension, Phishing Website Detection, Machine Learning

1. INTRODUCTION

Phishing has become one of the most common types of cyberattacks today. In these attacks, criminals trick users into sharing personal or sensitive information by creating fake websites that look like the real ones.The fake sites often have anti-suspicion mechanisms in place, and it is therefore difficult to detect phishing using existing countermeasures. Blacklists and heuristic-based systems currently in place do not keep pace with fast-changing attack patterns and tend to fail to notice newly established phishing sites. This generates an urgent requirement for more intelligent and adaptive solutions. Machine learning (ML) provides a potential method by adapting to previous data to recognize faint signs of phishing in real time. Recent developments in this respect have yielded promising outcomes. Verma and Das [1] emphasized the efficiency of URL-based feature extraction for phishing classification. Other researchers, including Abdelhamid et al. [2], utilized rough set theory, whereas Basnet et al. [3] experimented with different ML algorithms to enhance classification accuracy. Research by Zhang et al. [4] and Liu et al. [5] brought about vision and content-based approaches, enriching previous detection methodologies. All these works have provided solid grounds for the evolution of intelligent and more trustworthy tools.

In this work, we propose a browser extension that uses machine learning to detect phishing websites in real time. The software analyzes several factors, like the URL structure, page

Proceedings DOI: 10.21467/proceedings.7.5; Series: AIJR Proceedings; ISSN: 2582-3922; ISBN: 978-81-989164-7-1

content, domain credibility, and visual elements—to determine whether a website is genuine or not. Our contribution is in developing a lightweight, easy-to-use system that functions smoothly in the browser without interrupting the user. The model is fuelled by an XGBoostclassifier that has been trained on a wide dataset to maximize coverage. The main aim of this project is to create a simple and effective way to keep users safe while they browse the internet.

2. MATERIALS AND METHODS

The proposed system combines browser extension development with machine learning (ML) techniques to detect phishing websites in real time. The implementation involves both client-side interface design and server-side model processing, with each component detailed below.

2.1 Development Stack:

The browser extension was developed using JavaScript, HTML, and CSS for the front-end interface. For backend processing, a Flask API written in Python was used to host the trained machine learning model and handle prediction requests. JSON was utilised as the data interchange format between the browser extension and the API, enabling structured communication of extracted features and prediction results. The extension listens for the active tab's URL and initiates analysis automatically when a user navigates to a new page.

2.2 Machine Learning Algorithms:

The system was tested using XGBoostclassifie**r**, a machine learning algorithm that works well with table-like data and often provides excellent results. The model was trained on a labeled dataset containing both phishing and legitimate URLs, achieving the best accuracy and was selected for deployment.

2.3 Feature Collection:

Features were extracted primarily from the URL and web metadata. This includes checks on URL length, presence of suspicious characters (like "@" or "//"), use of HTTPS, number of subdomains, and domain age. Additional scrutiny was applied to embedded JavaScript behaviour and forms to identify hidden malicious intent. Domain age and registrar data were retrieved via WHOIS queries.

2.4 Security Integration:

To ensure user safety, a real-time alert system was built into the extension. When a phishing site is detected, the extension immediately displays a warning message, allowing the user to navigate away before interacting with the page. Additionally, if the model predicts a phishing attempt with 100% confidence, the system automatically blocks access to the site, preventing the page from loading entirely. This automatic blocking mechanism adds an extra layer of protection by eliminating any chance of user interaction with highly suspicious content. All of this is implemented without disrupting the overall browsing experience.

Proceedings DOI: 10.21467/proceedings.7.5
ISBN: 978-81-989164-7-1

Series: AIJR Proceedings
ISSN: 2582-3922

3. THEORY AND CALCULATION

The detection methodology in this study builds upon both static and behavioural analysis of web pages to distinguish phishing sites from legitimate ones. This foundation integrates principles from machine learning theory and feature-based classification models to construct a robust detection system. One of the fundamental components is feature engineering, which involves identifying and extracting attributes that are statistically or heuristically linked to phishing behaviour. Typical features include unusually long URLs, presence of special characters (e.g., '@', '//'), absence of HTTPS protocols, and recently registered domains. These features serve as indicators of abnormal patterns often associated with malicious sites.

From a theoretical standpoint, this system applies supervised learning, where models are trained on a labeled dataset $D = \{ (x_i, y_i) \}_{i=1}^{n}$, with x_i representing the feature vector and $y_i \in \{0,1\} = label\,(0 = legitimate, 1 = phishing)$ denoting whether the site is legitimate or phishing. The goal is to learn a function $f(x)$ that minimises classification error over unseen data. This concept relies on minimising a loss function such as:

$$L(y,\hat{y}) = (1/n) \sum_{i=1}^{n} l(y_i, f(x_i))$$

where l is a suitable loss function, for example, log loss or hinge loss, depending on the classifier. This equation is a standard loss function for binary classification using probabilistic outputs [6]. The machine learning model used, XGBoostClassifier, is an ensemble method based on gradient boosting, which incrementally builds decision trees to minimise classification error. The computation of prediction confidence is crucial in this setup; when the model's confidence score for a phishing prediction reaches 100%, access to the suspected site is automatically blocked at the extension level. This layered approach, grounded in feature analysis and supervised learning theory, supports both proactive detection and practical enforcement in real-world browsing environments.

4. RESULTS AND DISCUSSION

This section summarizes an evaluation of the outcomes achieved when using the phishing detection browser extension based on its precision, performance efficiency, difficulty with false alerts, and usage in real-world applications. Beneath it all, the system employs a machine learning model, the XGBoostclassifier, within a slim browser extension to provide real-time threat detection. Figure 1 shows a pop-up alert that appears automatically when someone visits a phishing website, quickly warning them to stay safe. Figure 2 shows the phishing website detection process by the extension, indicating how the system evaluates URL features and marks malicious content in real-time. Figure 3 illustrates the detection of a genuine site, in which the extension processes the URL and verifies that the site is secure and therefore no warning is shown. Lastly, Figure 4 illustrates the block-access feature being triggered when the model achieves a 100% confidence rating that a website is phishing, directly blocking the user from accessing the malicious webpage. These screenshots collectively demonstrate the extension's interface and key features in operation, highlighting its real-time detection, user

alerting, and automatic blocking features. In short, the outcomes are a significant contribution to browser-based phishing detection with high accuracy, low processing overhead, automatic protection, and ease of use. Scientifically, it validates the role of machine learning in cybersecurity defenses while also pointing out areas of future tuning. Technically, the extension demonstrates how light-weight, real-time security solutions can be successfully integrated into routine web browsing to protect users from relentless and dynamic cyber attacks.

4.1 Accuracy and Performance Evaluation

The performance and accuracy analysis indicated that the machine learning model achieved around 95% classification accuracy across a varied dataset of phishing and legitimate websites. Some of the key performance measurements were a precision rate of 92%, which demonstrates that the majority of phishing websites identified by the system were correctly marked. The model achieved a 90% recall rate, showing how well it can detect phishing attempts while missing very few. An F1 score of 0.91 also illustrated the strength of the system by maintaining precision and recall in good balance. Its high accuracy is attributed to the model's ability to examine URL patterns, domain metadata, and JavaScript behaviors that are typically present in phishing activities.

4.2 Detection Speed and Efficiency

In terms of detection speed and effectiveness, the system was created to conduct real-time detection without compromising browser performance. The feature extraction and classification steps were executed within a period of 500 milliseconds, causing virtually no delay. The extension processes WHOIS data, inlined scripts, and URL structure processing locally, which helped improve its responsiveness. Ongoing testing ensured that the extension never slowed page load times significantly, maintaining a seamless user experience.

4.3 Automatic Blocking at Maximum Confidence

An important feature of the system is its auto-blocking mechanism. If the model registers a 100% confidence level that a site is phishing, the extension blocks access to the suspect webpage without delay, negating the need for user intervention. This anticipatory defense provides an essential security layer, particularly useful for new users who would otherwise ignore phishing alerts.

4.4 False Positives and False Negatives

While the system is generally quite reliable, there were occasional instances of incorrect predictions. False positives only occurred in a small percentage of valid sites wrongly flagged, usually because of unusual features like very long URLs. Conversely, certain phishing sites evaded detection—false negatives—mostly because they used new patterns of attack not included within the training set. To counter such challenges, future releases will increase the dataset and incorporate dynamic threat intelligence to minimize such mistakes

Proceedings DOI: 10.21467/proceedings.7.5
ISBN: 978-81-989164-7-1

Series: AIJR Proceedings
ISSN: 2582-3922

4.5 Real-World Usability

Testing the extension on phishing and legitimate sites confirmed its real-world utility. The system effectively intercepted phishing attacks using URL shortening, visual obfuscation, and social engineering practices, while passing genuine traffic without undue block. User feedback was overwhelmingly positive, praising the extension's informative alerts, ease of installation, and unobtrusive integration into daily browsing routines. Overall, the phishing detector browser extension is an effective and intuitive means to improve online security.

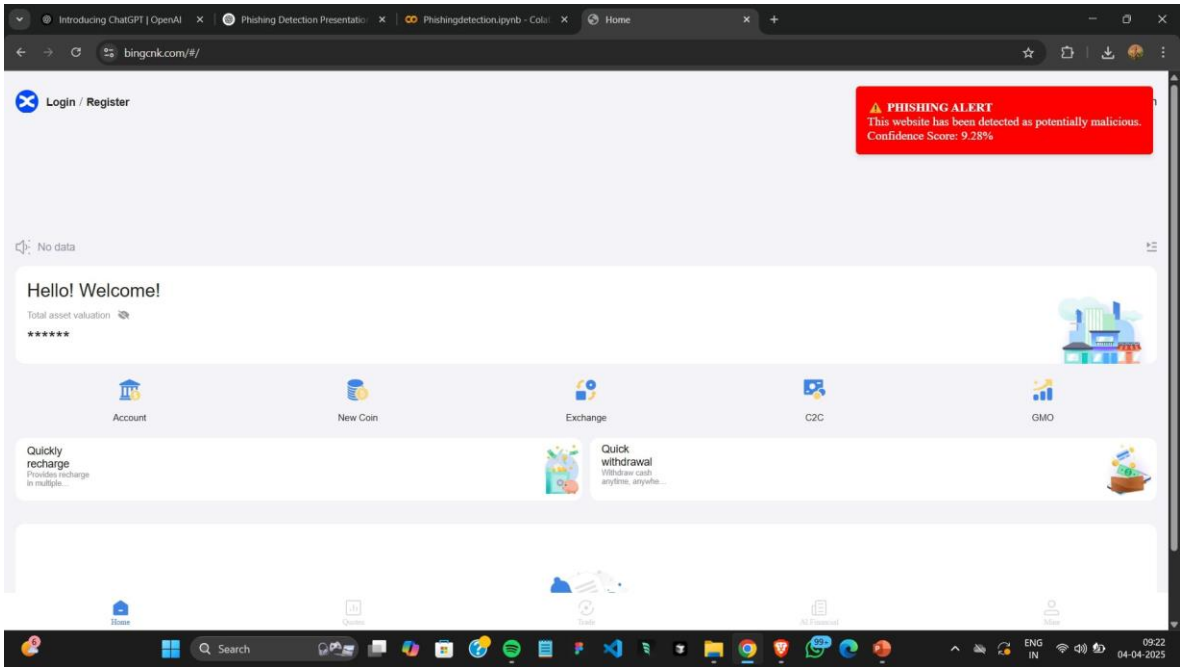

Figure 1:*Automatic pop-up when the user enters a phishing website*

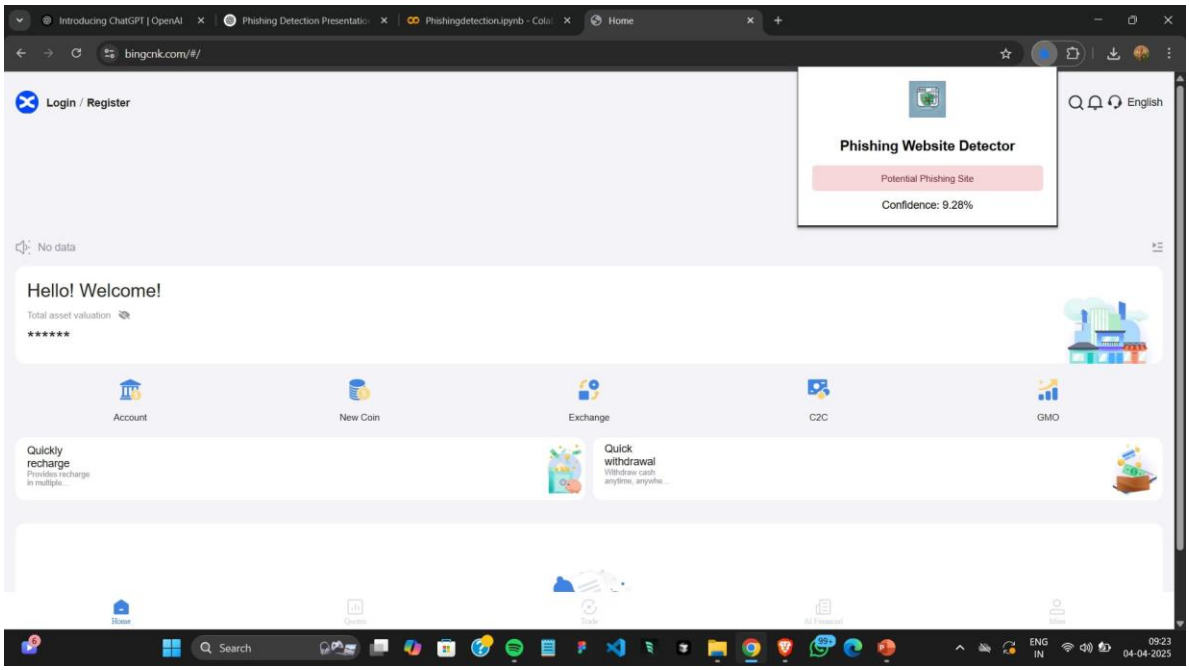

Figure 2: *Detecting a Phishing Website*

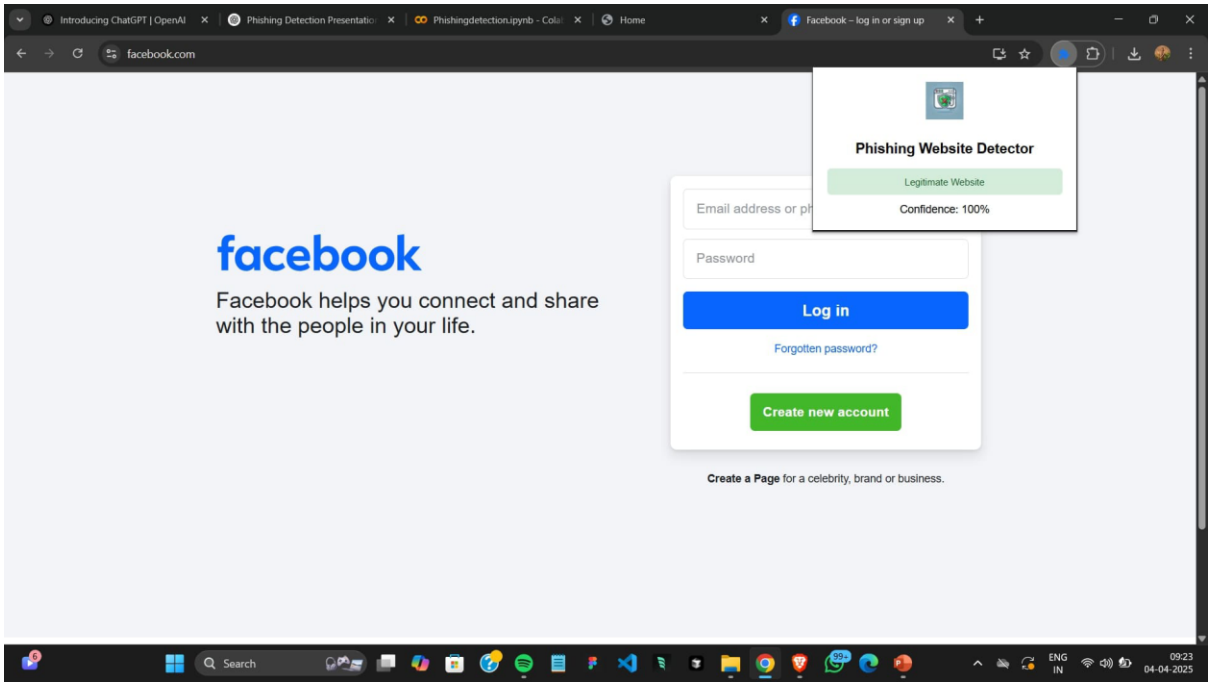

Figure 3: *Detecting a Legitimate Website*

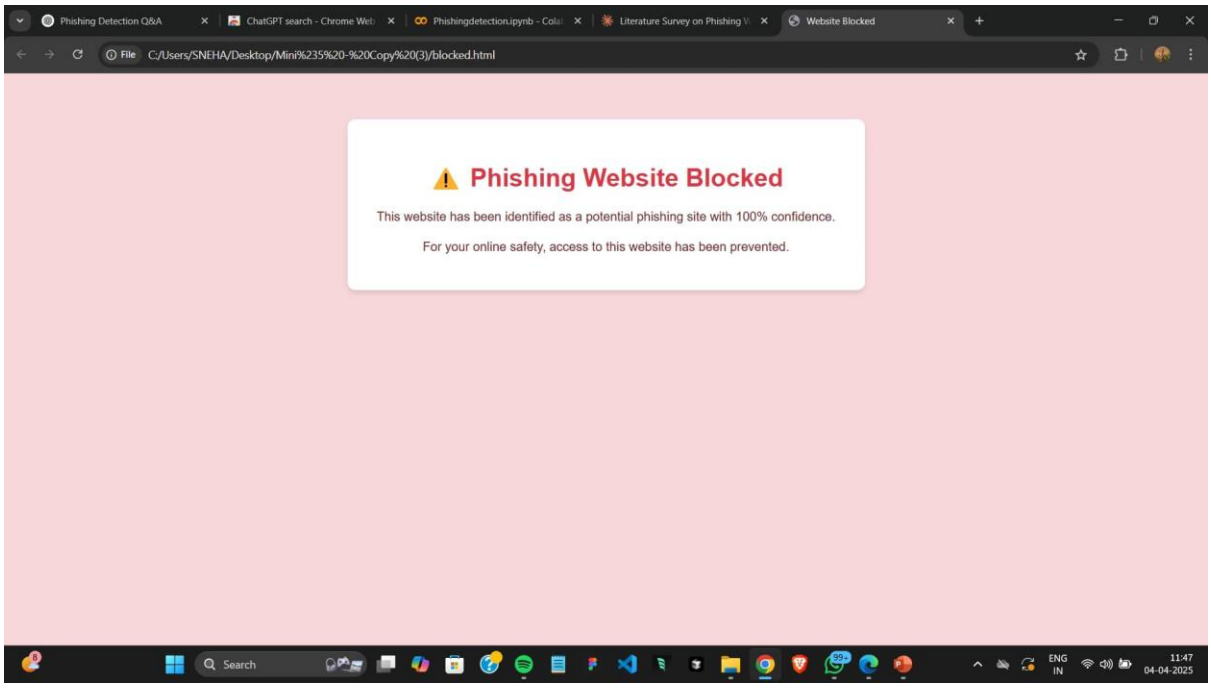

Figure 4: *Access blocked when the user entered a 100% Phishing site*

5. CONCLUSION

The project titled "Browser Extension for Phishing Website Detection using Machine Learning" effectively addresses the growing threat of phishing attacks by harnessing artificial intelligence and real-time web monitoring. Phishing remains one of the most prevalent techniques used by cybercriminals to steal sensitive information, and this system offers a smart,

Proceedings DOI: 10.21467/proceedings.7.5
ISBN: 978-81-989164-7-1

Series: AIJR Proceedings
ISSN: 2582-3922

efficient, and user-friendly solution that proactively detects such threats before any harm is caused. By extracting essential features from URLs and applying a trained machine learning model, the system accurately classifies websites as either legitimate or malicious without relying on traditional blacklists, which are often outdated or incomplete. The browser extension integrates seamlessly into the user's browsing environment, offering real-time protection with minimal impact on performance and user privacy. It operates in the background, alerting users only when necessary, and requires no technical expertise to use. Its adaptability allows the system to evolve continuously by incorporating new phishing patterns into the training data. This work demonstrates a practical application of machine learning in cybersecurity, bridging the gap between advanced AI technologies and end-user safety. Looking ahead, future enhancements such as deeper URL analysis, visual similarity detection, and server-based model updates could further strengthen the system's robustness.

6. DECLARATIONS

6.1 Competing Interests

The authors declare that there are no conflicts of interest regarding this work

6.2 Acknowledgements

We would like to express our heartfelt gratitude to the faculty and staff of the Department of Computer Science and Engineering at Adi Shankara Institute of Engineering and Technology for their unwavering support and guidance. Their encouragement has been instrumental in the success of this project. We also wish to extend our thanks to our peers and family members for their continuous feedback, motivation, and belief in us throughout this journey.

6.3 Study Limitations

The system may fail to detect newly developed phishing websites that do not share common traits with the training data. Additionally, the model's performance is influenced by the size and quality of the dataset and may require periodic updates to maintain accuracy.

6.4 Publisher's Note

AIJR remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

How to Cite

Shajahan *et al.*, "Browser Extension for Phishing Website Detection Using Machine Learning", *AIJR Proc.*, vol. 7, no. 5, pp. 30-37, Sep. 2025. doi: https://doi.org/10.21467/proceedings.7.5.5

REFERENCES

[1] R. Verma and K. Das, "Suspicious URL Detection Using URL and Host-Based Features," *Proceedings of the 3rd ACM Workshop on Security and Artificial Intelligence*, pp. 49–54, Oct. 2010. Access online on 30 May 2025 at https://dl.acm.org/doi/10.1145/1866423.1866435

[2] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based on rough set theory," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5948–5959, Oct. 2014. Access online on 30 May 2025 at https://www.sciencedirect.com/science/article/pii/S0957417414001972

[3] R. B. Basnet, A. H. Sung, and Q. Liu, "Learning to detect phishing URLs," *International Journal of Research in Engineering and Technology*, vol. 3, no. 6, pp. 11–24, June 2014. Access online on 30 May 2025 at https://ijret.org/volumes/2014v03/i06/IJRET20140306003.pdf

[4] Y. Zhang, J. Hong, and L. Cranor, "CANTINA: A content-based approach to detecting phishing web sites," *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, pp. 639–648, May 2007. Access online on 30 May 2025 at https://www.cs.cmu.edu/~pongle/phishing.pdf

[5] W. Liu, X. Deng, G. Huang, and A. Y. Fu, "An anti-phishing strategy based on visual similarity assessment," *IEEE Internet Computing*, vol. 10, no. 2, pp. 58–65, Mar.–Apr. 2006. Access online on 30 May 2025 at https://ieeexplore.ieee.org/document/1607723

[6] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2009. Access online on 30 May 2025 at https://link.springer.com/book/10.1007/978-0-387-84858-7