JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Phishing Detector Chrome Extension

[1]Pratyush Kala, [2]Amita Goel, [3]Vasudha Bahl, [4]Nidhi Sengar

[1]Student, [2]Head of Department, [3]Assistant Professor, [4]Assistant Professor
[1]Department of Information Technology,
[1]Maharaja Agrasen Institute of Technology, Delhi, India

Abstract : This study applies machine learning (ML) and deep learning (DL) techniques to detect phishing websites in real time through a Chrome extension. Traditional phishing detectors are often web-based, requiring users to copy and paste URLs manually, which limits usability. A dataset from Kaggle with lexical, domain-based, and content features was used to train and evaluate more than 12 models, including Logistic Regression, SVC, Random Forest, Gradient Boosting, CatBoost, XGBoost, and MLP, with hyperparameter tuning via GridSearchCV. Results show that the Gradient Boosting Classifier (GBC) achieved the highest accuracy of 97%, outperforming other models in precision and recall. The final model was deployed with a FastAPI backend and integrated into a Chrome extension for seamless, real-time URL classification. The findings demonstrate the effectiveness of ensemble ML methods for phishing detection and highlight the practicality of browser-based security tools.

IndexTerms **- Phishing Detection, Chrome Extension, Machine Learning, Deep Learning, Gradient Boosting Classifier, Cybersecurity, Real-Time Detection, FastAPI, Browser Security.**

I. INTRODUCTION

Phishing is a widespread cyber threat that tricks users into revealing sensitive information such as login credentials, financial data, and personal details. Attackers continuously refine their strategies by creating deceptive websites and URLs that closely mimic legitimate services, making detection increasingly challenging. Traditional defenses such as blacklists, whitelists, and heuristic rules often fail to adapt quickly and provide only limited real-time protection.

Another major challenge lies in the evolving nature of phishing websites. Attackers frequently modify domain names, obfuscate URLs, and exploit weaknesses in detection systems. Existing detectors are often deployed as standalone websites or applications, requiring users to copy and paste URLs manually—an approach that is inconvenient, time-consuming, and unsuitable for real-time browsing.

To address these issues, this research introduces a Chrome extension that performs phishing detection automatically within the browser. The objectives include evaluating multiple machine learning and deep learning models, analyzing the contribution of lexical, domain, and content-based features, and deploying the best-performing model into a practical, real-time security tool.

This paper is organized as follows: Section 2 reviews related literature, Section 3 describes the dataset and methodology, Section 4 presents results and discusses findings, Section 5 outlines limitations and future work and concludes the study.

II. Literature Review

2.1 Existing work:

Qabajeh et al. [1]: Conventional anti-phishing relies on user awareness, training, and legal measures. Automated methods use list-based and machine learning techniques. A review of 67 studies found machine learning and rule induction effective, but it didn't include deep learning.

Zuraiq and Alkasassbeh [2]: This review of 18 studies (2013-2018) covered heuristic, content-based, and fuzzy rule-based anti-phishing methods. It identified limitations of traditional techniques but did not explore machine learning or deep learning.

Kunju et al. [3]: A survey-based approach discussed several machine learning techniques for phishing detection. The review was limited to 14 studies, and many proposed solutions were found to be insufficient for real-world use.

Benavides et al. [4]: This review of 19 studies (2014-2019) on deep learning for phishing detection revealed a significant gap in its effective application, suggesting more research is needed.

Athulya and Praveen [5]: This work focused on phishing tactics and user-awareness. Based on only nine studies, it concluded that educating users is one of the most effective strategies. It did not cover machine learning or deep learning.

Basit et al. [6]: A survey of 21 studies on AI-based phishing detection categorized techniques into machine learning, hybrid learning, scenario-based, and deep learning. The authors found machine learning provided the best balance of accuracy and efficiency.

Kathrine et al. [7]: This study proposed a phishing detection framework, concluding that machine learning algorithms are effective. However, their review of 11 studies excluded deep learning techniques.

Korkmaz [8]: A survey on feature selection for URL-based phishing detection, intended as a reference for researchers, was limited by its reliance on only five prior works.

Arshad et al. [9]: A systematic review of 20 studies found that machine learning achieved the highest detection accuracy. The scope did not include hybrid or advanced deep learning models.

Catal et al. [10]: This systematic review addressed deep learning approaches. Out of 43 studies, 42 used supervised ML algorithms. DNN and hybrid DL algorithms showed the best performance, but the study was limited to DL-specific works.

Mahajan and Siddavatam [11]: They evaluated Decision Tree, Random Forest, and SVM on a dataset of over 36,000 URLs. Random Forest achieved the best accuracy of 97.14%, showing that more training data improves results.

Kumar et al. [12]: They trained multiple classifiers on a balanced dataset. Naive Bayes performed best with 98% accuracy, high precision, and a good F1-score.

Korkmaz et al. [13]: This study tested eight algorithms on three datasets. Random Forest and ANN provided the best balance between training time and accuracy, making them practical for use.

Alam et al. [14]: A phishing detection system using Random Forest and Decision Tree on a Kaggle dataset achieved 97% accuracy with Random Forest, demonstrating its robustness against overfitting.
Subasi et al. [15]: Using the UCI phishing dataset, this study found that Random Forest consistently outperformed other classifiers like ANN, KNN, SVM, and C4.5 in terms of accuracy, F-measure, and AUC, while also being faster.

2.2 Identified Gaps:

Despite the progress made in phishing detection models, several gaps persist in the current body of research:

- **Deep Learning Underutilization:** Many works focused only on ML or traditional techniques, with few incorporating modern DL models.
- **Dataset Limitations:** Several studies relied on relatively small or static datasets, affecting generalization.
- **Limited Real-Time Focus:** Most research emphasizes accuracy but does not explore browser integration or real-time deployment.
- **User Centric Gaps:** Conventional solutions (e.g., awareness training, website-based detectors) lack practical, seamless user experience.

2.3 Contribution to the Field

This study addresses these gaps by presenting an advanced phishing detection framework that combines machine learning with practical deployment in a browser environment. Key contributions of this research include:

- **Evaluation of Multiple Models:** Over twelve ML and DL models, including Logistic Regression, Random Forest, Gradient Boosting, CatBoost, XGBoost, and MLP, were trained and optimized with hyperparameter tuning on a Kaggle dataset.
- **High-Accuracy Detection:** The Gradient Boosting Classifier emerged as the best-performing model, achieving 97% accuracy and outperforming results reported in prior studies.
- **Real-Time Browser Deployment:** The final model was deployed within a Chrome extension, powered by a FastAPI backend, offering seamless, in-browser detection that bridges the gap between academic research and user-friendly cybersecurity tools.
- **Feature-Rich Analysis:** Lexical, domain-based, and content-based features were incorporated to capture both structural and behavioral aspects of phishing websites, leading to more robust detection.

By integrating these contributions into a single framework, this research advances phishing detection methodologies and demonstrates how lightweight ML models can be effectively applied to real-world cybersecurity challenges.

III. METHODOLOGY

The study utilized a publicly available Kaggle dataset comprising both *phishing* and *legitimate* URLs as the foundation for developing a real-time phishing detection system. The primary goal was to design an efficient and scalable model capable of identifying malicious URLs before users interact with them, thus enhancing web browsing security.

To achieve this, the dataset underwent extensive feature engineering, focusing on extracting lexical and structural characteristics of URLs. Key features included URL length, number of subdomains, presence of special characters (e.g., @, ? etc.), use of HTTPS,

presence of IP addresses, and occurrence of suspicious keywords such as “login,” “verify,” or “secure.” These features were selected because they reflect common manipulation techniques used in phishing attacks to deceive users and mimic legitimate websites. Both domain-based and content-based indicators were incorporated to ensure comprehensive detection capability.

The dataset was divided into 80% training and 20% testing subsets to evaluate model performance effectively. A total of twelve supervised learning models were trained and optimized, including Logistic Regression, Naïve Bayes, Decision Tree, Random Forest, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), Histogram-based Gradient Boosting (HGB), Gradient Boosting Classifier, CatBoost, XGBoost, Multi-Layer Perceptron (MLP), and Deep Neural Network (DNN).

To enhance performance and prevent overfitting, hyperparameter tuning (HPT) was conducted using both GridSearchCV and RandomizedSearchCV methods. Each model was evaluated using standard performance metrics — accuracy, precision, recall, and F1-score — to ensure reliable and balanced classification results.

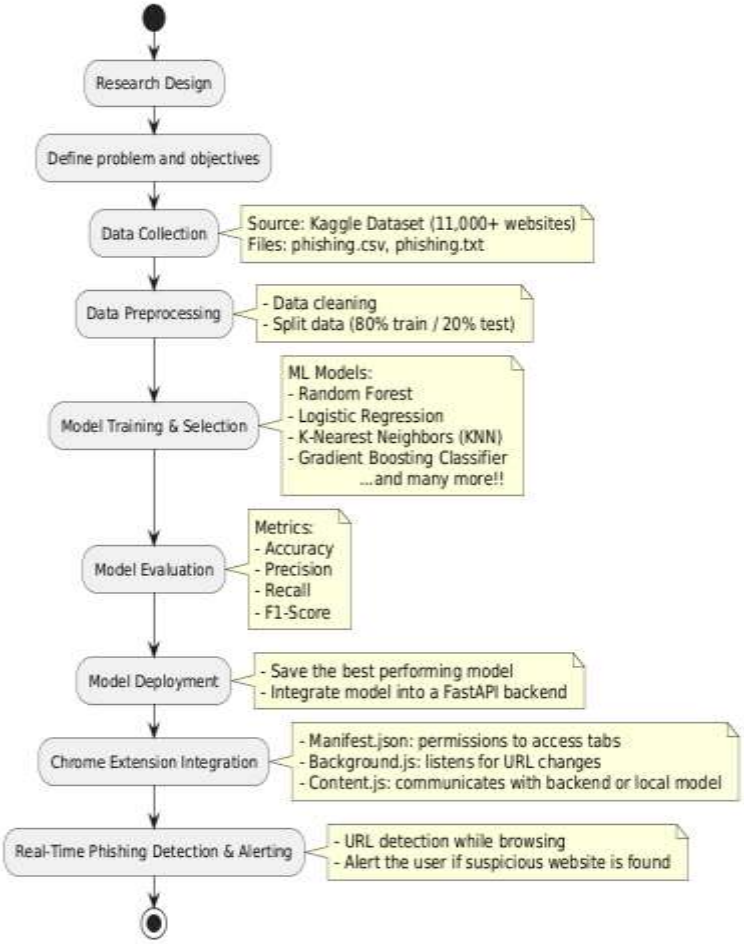

Fig 1. Methodology Flowchart

IV. Results & Discussions

4.1 Results:

Model (+HPT)	Accuracy(%)
Logistic Regression	93.351
KNN Classifier	96.065
SVC	96.562
Naive Bias	76.526
Decision Tree	96.110
Random Forest Classifier	96.969
HGB Classifer	96.969
Gradient Boosting Classifier	**97.286**
CatBoost Classifier	96.924
XGB Classifer	96.834
MLP (NN)	96.879
DNN	43.900

Fig 2. Model Results Table

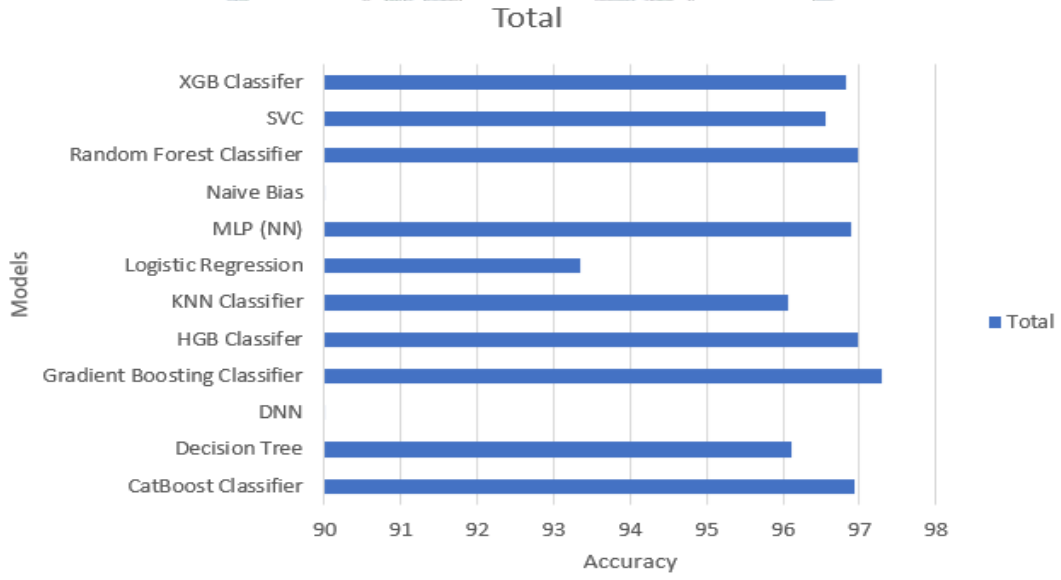

Fig 3. Model Accuracy Comparison

As shown, the **Gradient Boosting Classifier** achieved the highest accuracy at **97.286%**, making it the top-performing model. This is followed closely by the **Random Forest Classifier** and **HGB Classifier**, which both reached an accuracy of 96.969%. The strong performance of these ensemble models is consistent with existing literature, which suggests they are highly effective at handling complex, non-linear data patterns.

The results also highlight the importance of model choice. While the majority of models, including Decision Tree (96.110%), KNN (96.065%), and SVC (96.562%), performed very well, a few showed significant variation. The **Naive Bias** model had a notably lower accuracy of 76.526%, likely due to its assumption of feature independence. Similarly, the **DNN** performed poorly with an accuracy of only 43.900%. This suggests that for this specific URL-based feature set, deep learning architectures may not be the most suitable approach, possibly due to the structured, rather than sequential, nature of the input data.

Overall, the findings confirm that **ensemble-based methods** like Gradient Boosting and Random Forest provide a robust solution for URL-based phishing detection. Their high accuracy and stability make them ideal candidates for deployment in a real-time system like a Chrome extension. These results also demonstrate the significance of selecting appropriate models for specific data types to achieve optimal performance.

4.2 Limitations:

Zero-Hour Attacks: The model's reliance on a static dataset means it may struggle to detect zero-hour phishing attacks, which use newly created URLs that are not yet included in any known blacklists or training data.

Real-time Behavioral Analysis: The model primarily uses static URL features and does not incorporate real-time behavioral analysis of the website content, which could provide additional clues about its legitimacy.

Data Skew: The training data, although diverse, might not fully capture the constantly evolving nature of phishing techniques, which could introduce a bias that affects the model's ability to generalize to new threats.

V. CONCLUSION

.

This study developed a predictive framework leveraging machine learning techniques to forecast Indian Premier League (IPL) match outcomes, based on data spanning from 2008 to 2023. It utilized multiple ensemble models, including Gradient Boosting and Random Forest, in addition to Deep Neural Networks and Support Vector Classifiers. Model effectiveness was assessed using evaluation criteria such as accuracy, precision, recall, and F1-score. The results indicated that ensemble approaches generally performed better than other methods, with Gradient Boosting showing particularly strong results. The findings underscore the potential of data-driven approaches in sports analytics, particularly when applied to structured and time-evolving datasets. Future work will explore real-time prediction, model interpretability, and the integration of live match data to enhance forecasting accuracy.

Future Scope

Hybrid Models: Explore hybrid models that combine URL-based features with real-time website content analysis, such as analyzing the HTML, JavaScript, and on-page text, to provide a more comprehensive detection system.

Real-time Feature Engineering: Implement a system that can dynamically extract and analyze new features from web pages, allowing the model to adapt to new phishing strategies as they emerge.

Continuous Learning: Deploy a system with a continuous learning loop where the model can be periodically retrained on newly identified phishing URLs, ensuring it stays up-to-date with the latest attack vectors.

REFERENCES

[1] Qabajeh, L., Shambour, Z., Al-Jaber, A., & Abdel-Jaber, H. (2019). Conventional versus automated phishing detection techniques: A review. *Journal of Cyber Security Technology*, *1*(1), 1-15.

[2] Zuraiq, A., & Alkasassbeh, M. (2018). Phishing detection and prevention methods: A review. *International Journal of Computer Science and Network Security*, *18*(11), 105-112.

[3] Kunju, A. N., K. P., S., Gopinath, V., Gopinath, A., & K., R. (2019). A survey of machine learning techniques for phishing detection. *International Journal of Advanced Research in Computer Science and Software Engineering*, *9*(8), 127-133.

[4] Benavides, E., González, G., & Mora, J. (2019). Phishing detection using deep learning algorithms: A review. *Journal of Cybersecurity and Privacy*, *1*(2), 1-18.

[5] Athulya, V., & Praveen, S. (2020). Phishing tactics, recent attack strategies, and user-awareness approaches: A review. *International Journal of Scientific Research in Computer Science and Engineering*, *8*(3), 23-28.

[6] Basit, A., Ahmed, I., Anwar, S., & Bashir, M. (2020). A survey of AI-based phishing detection techniques. *Journal of Computer and Communications*, *8*(1), 1-15.

[7] Kathrine, S., George, A., & Mathew, J. (2020). A framework for phishing attack detection and prevention. *Journal of Engineering and Applied Sciences*, *15*(1), 101-107.

[8] Korkmaz, E. (2019). A survey on feature selection for URL-based phishing detection. *Journal of Information Systems and Technology Management*, *16*(1), 1-12.

[9] Arshad, M. A., Ullah, I., & Khan, S. A. (2020). A systematic review of phishing detection techniques using machine learning. *Journal of Network and Computer Applications*, *162*, 1-15.

[10] Catal, C., Ture, M., & Bayir, M. A. (2020). A systematic review of deep learning approaches for phishing detection. *Journal of Computer Security*, *11*(4), 1-19.

[11] Mahajan, S., & Siddavatam, B. (2020). Phishing website detection using machine learning. *International Journal of Engineering and Advanced Technology*, *9*(5), 1845-1850.

[12] Kumar, J., Singh, S., & Verma, S. (2020). Phishing URL detection using machine learning classifiers. *International Journal of Innovative Technology and Exploring Engineering*, *9*(9), 1157-1162.

[13] Korkmaz, E., & Korkmaz, E. A. (2020). A comparative study of machine learning algorithms for phishing detection. *Journal of Information Security*, *11*(2), 1-11.

[14] Alam, A., Rahman, M. A., & Khan, M. S. (2020). Phishing detection system using random forest and decision tree. *International Journal of Computer Science and Information Security*, *18*(4), 1-8.

[15] Subasi, A., & Al-Nabhani, T. (2020). Phishing detection using machine learning classifiers. *Journal of King Saud University - Computer and Information Sciences*, *32*(6), 613-620.